7 – Virtualisierung

Quellen:

http://www.informatik.hs-mannheim.de/~baun/CGC11/Skript/folien_cgc_vorlesung_13_SS2011.pdf
http://www.inf.fu-berlin.de/lehre/WS06/BS/folien/bs-9.2.pdf
http://en.wikipedia.org
http://www.cl.cam.ac.uk/research/srg/netos/xen/
http://www.itwissen.info/
http://www.elektronik-kompendium.de/

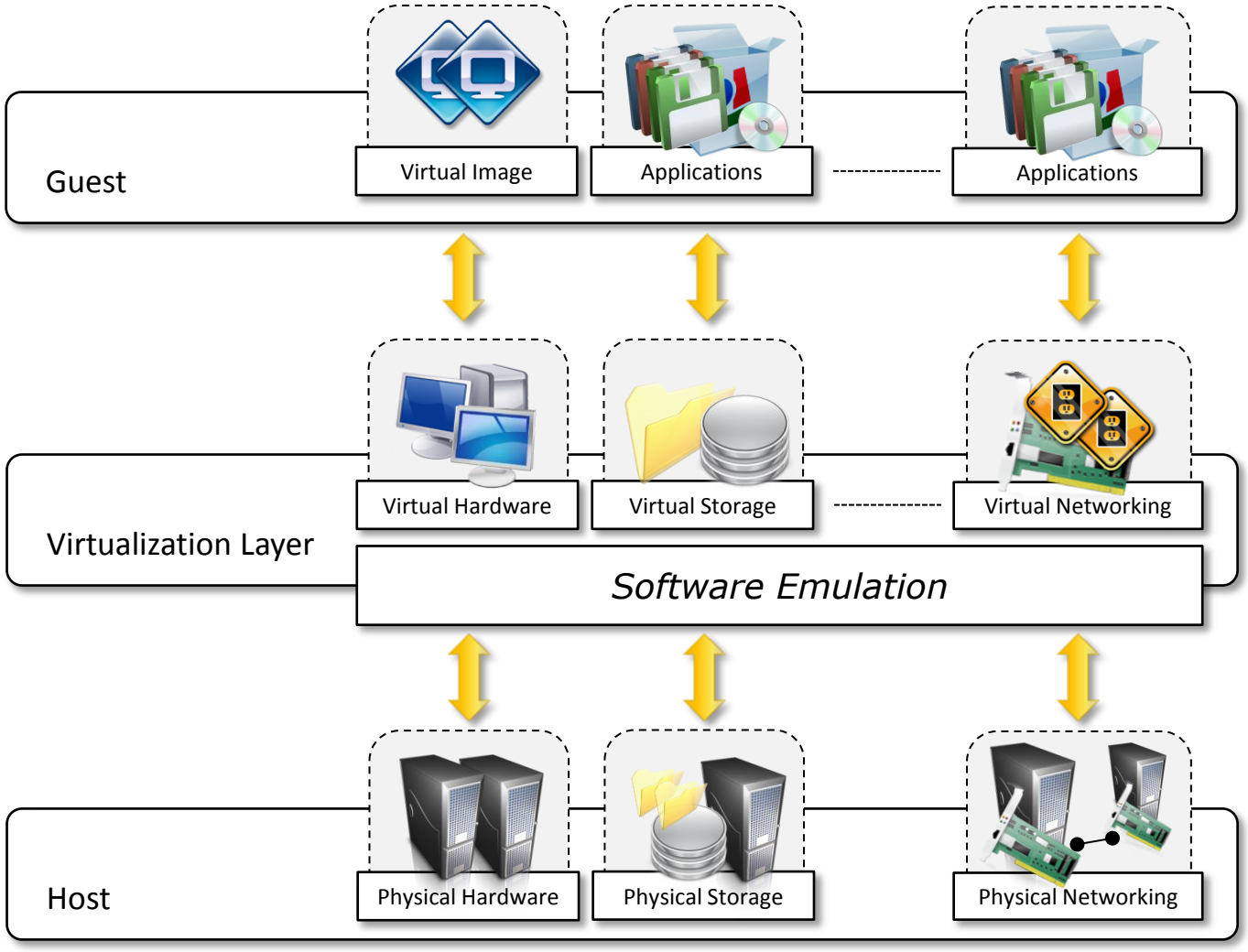
Definitionen

Virtualisierung bezeichnet in der Informatik die Erzeugung von virtuellen (d.h. nicht physikalischen) Dingen wie einer emulierten Hardware, eines Betriebssystems, Datenspeichers oder Netzwerkressource. Dies erlaubt es etwa, Ressourcen von Computern transparent zusammenzufassen oder aufzuteilen, oder ein Betriebssystem innerhalb eines anderen auszuführen. (Quelle: www.wikipedia.de)

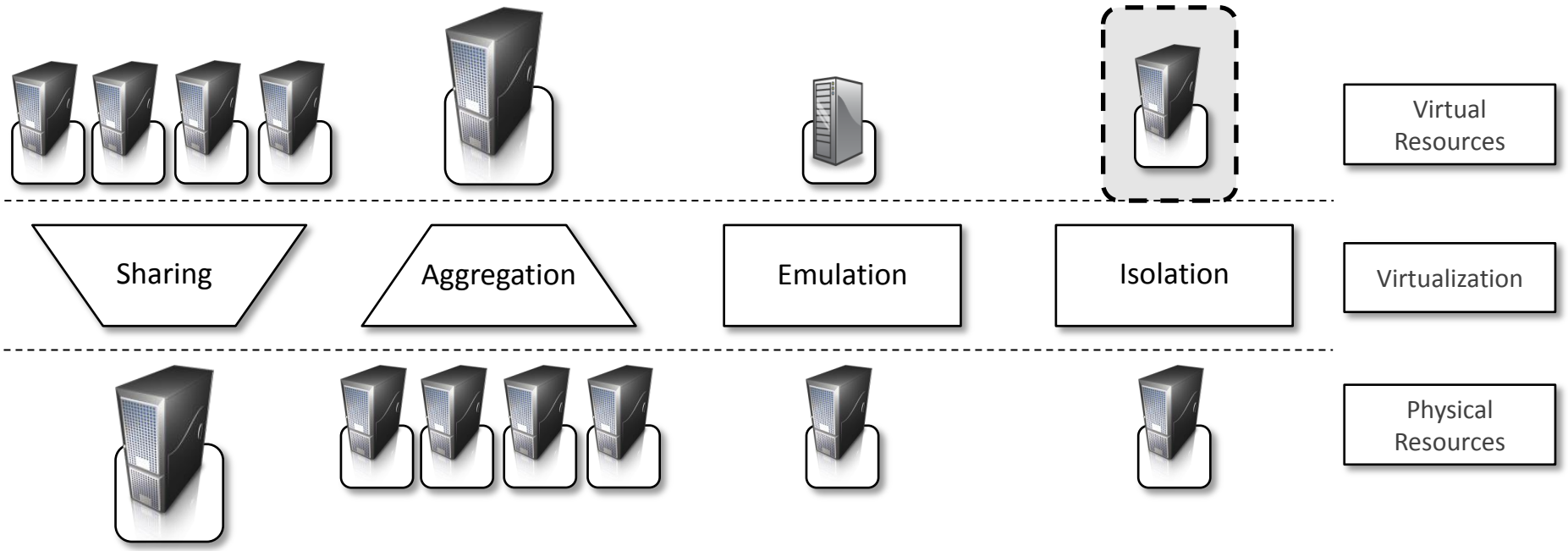
Virtualisierung ermöglicht die Abstraktion von Hardware, Software und Netzen. Im übergreifenden Sinn werden mit Virtualisierung Software- oder Hardware-Techniken bezeichnet, welche eine Abstraktionsschicht zwischen dem Benutzer (oder Applikationen oder Schnittstellen) einerseits und physischen Ressourcen wie z.B. Hardwarekomponenten eines Rechners andererseits, implementieren. (Quelle: www.itwissen.info)

Kurz: *Virtualisierung* bezeichnet Methoden, die es erlauben, Ressourcen eines Computers zusammenzufassen oder aufzuteilen.

Virtualisierung
Allgemeines

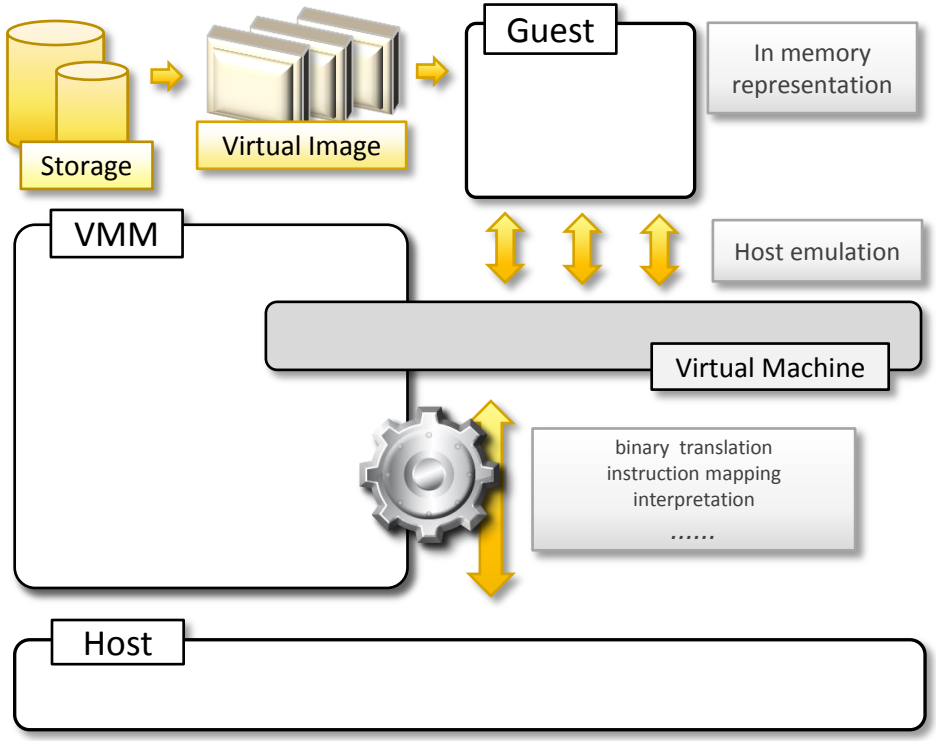


Virtualisierung
Allgemeines



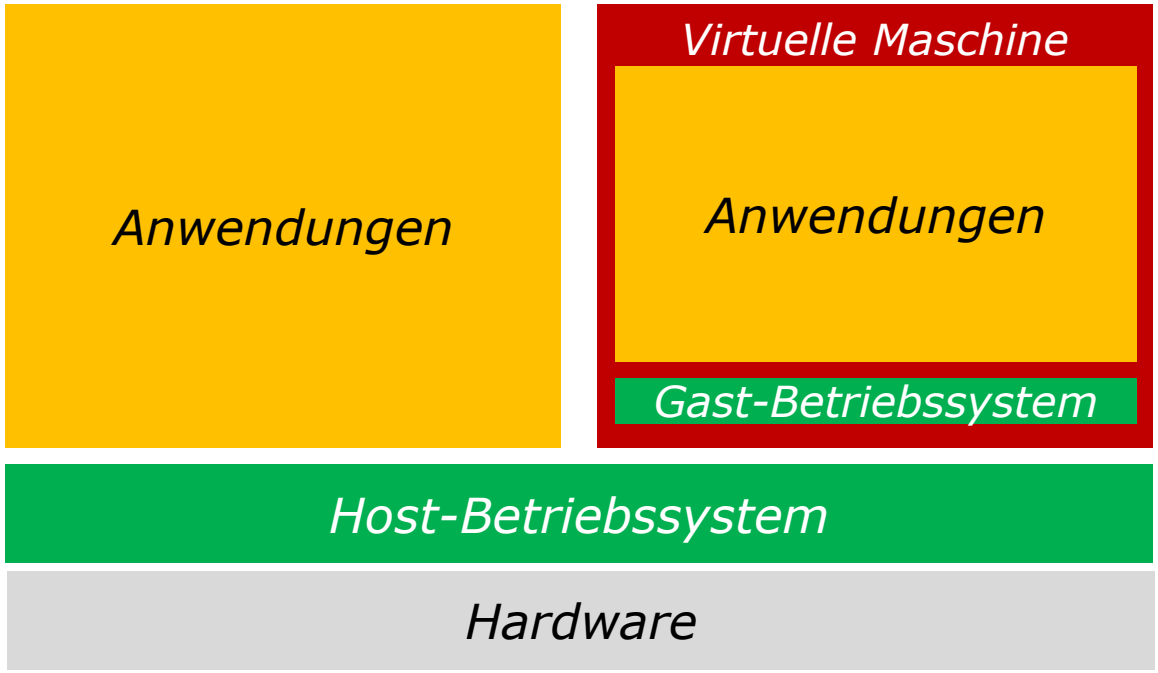
Eine *virtuelle* Maschine (VM) entsteht durch Virtualisierung einer *realen* Maschine

- Mehrere VMs mit unter Umständen auch verschiedenen Betriebssystemen sind auf derselben Hardware möglich



Eine *virtuelle* Maschine (VM) entsteht durch Virtualisierung einer *realen* Maschine

- Mehrere VMs mit unter Umständen auch verschiedenen Betriebssystemen sind auf derselben Hardware möglich



Virtualisierung
Virtuelle Betriebsumgebungen

Generelles Prinzip

- Es gibt eine Software, welche eine *virtuelle Maschine* erzeugt

Eine virtuelle Maschine kann einen vollwertigen „PC im PC" darstellen

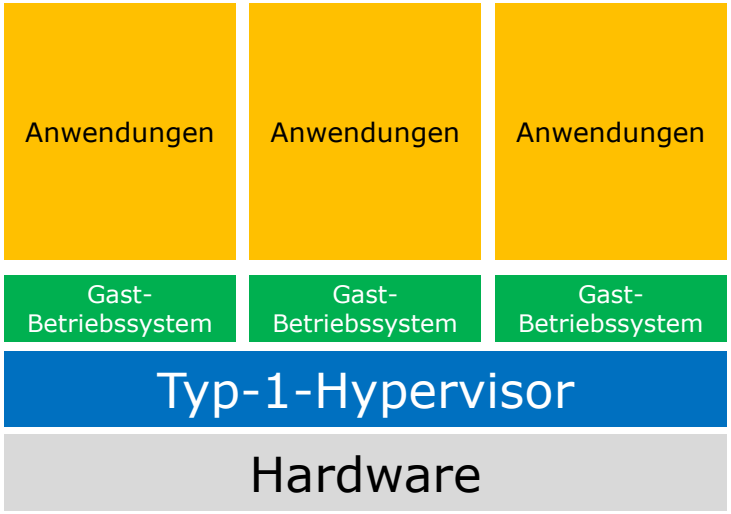
- Ein Betriebssystem kann installiert werden (Gast-BS)
- Dieses Gast-BS kann wie ein normales BS verwendet werden (Installation von Software, etc.)
- Ideal zum Testen von Software

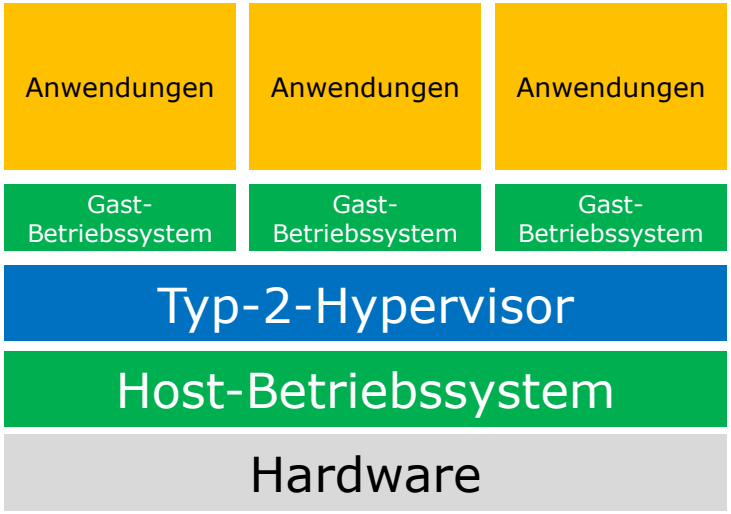
Eine virtuelle Maschine kann aber auch nur bestimmte Teile eines Systems nachbilden

- Einem Programm wird eine Umgebung zur Verfügung gestellt, die alles beinhaltet, was das Programm braucht
- z.B. JAVA Virtual Machine

Hypervisor oder Virtual Machine Monitor (VMM)

- Der *Hypervisor* ist ein Stück Soft- oder Hardware, das virtuelle Maschinen bereitstellt
- Er setzt die Zugriffe der VMs auf die Hardware entsprechend um
- Man unterscheidet Typ-1- und Typ-2-Hypervisor





Virtualisierungsmethoden und -plattformen

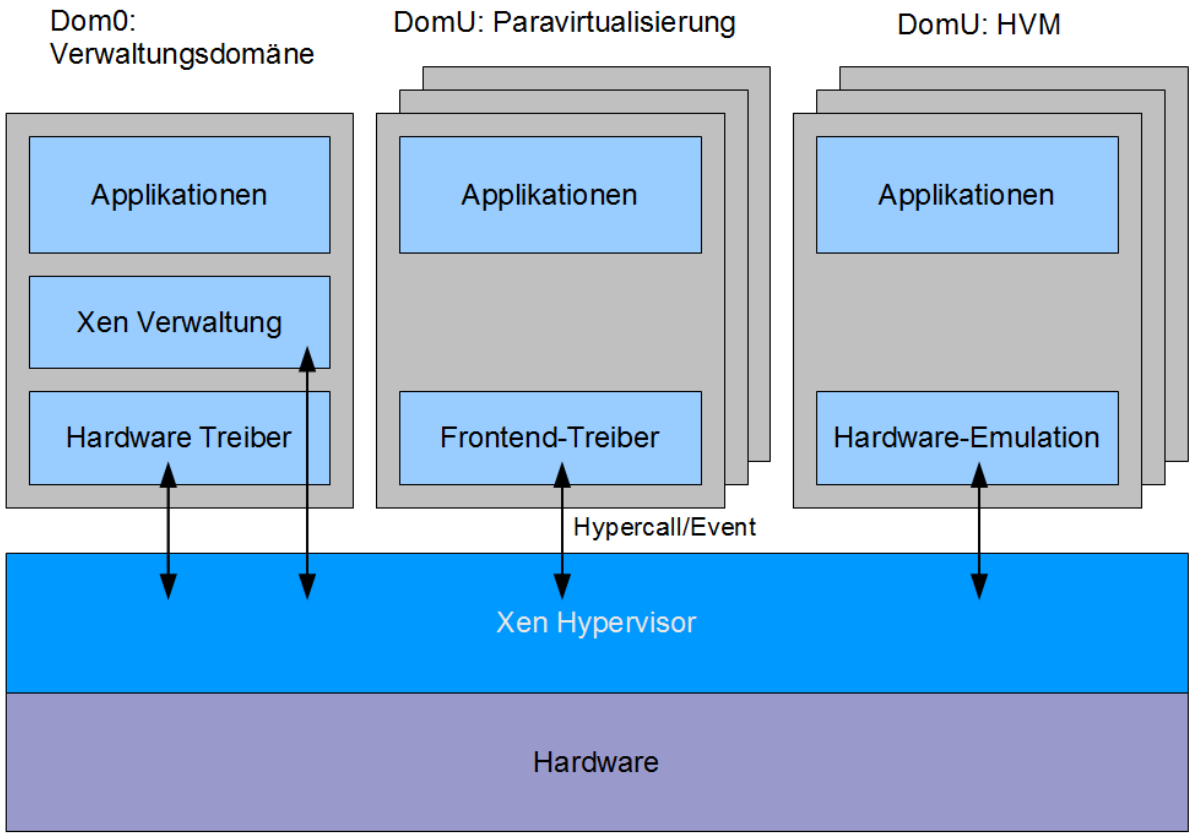
- Paravirtualisierung (z.B. Xen)
- Hardware-unterstützte Virtualisierung (z.B. KVM, VirtualBox, VMware)
- Betriebssystemvirtualisierung (z.B. OpenVZ)
- Hardware-Emulation (z.B. KVM, VirtualBox, VMware)

Paravirtualisierung

- Funktionalitäten des Gast-BS werden gezielt verändert (Kernel-Anpassungen)
- Gast-BS „weiß" somit, dass es sich in einer virtuellen Umgebung befindet
- Gast-BS kann sich direkt an den Hypervisor (Typ-1) wenden und benötigt keine Hardware-Emulation

→ Gute Performance
→ Gastsysteme nicht beliebig wählbar
→ Hoher Aufwand für Kernel-Entwickler

Paravirtualisierung - Xen



Hardware-unterstützte Virtualisierung

- Neue Prozessortechnologien, die CPUs besitzen einen Befehlssatz, der Virtualisierung direkt unterstützt (Intel-VT, AMD-V)
- Modifikation des Gast-BS soll vermieden werden und direkt durch Hardware gelöst werden
- Hypervisor soll durch hardwarebasierte Speicherverwaltung entlastet werden

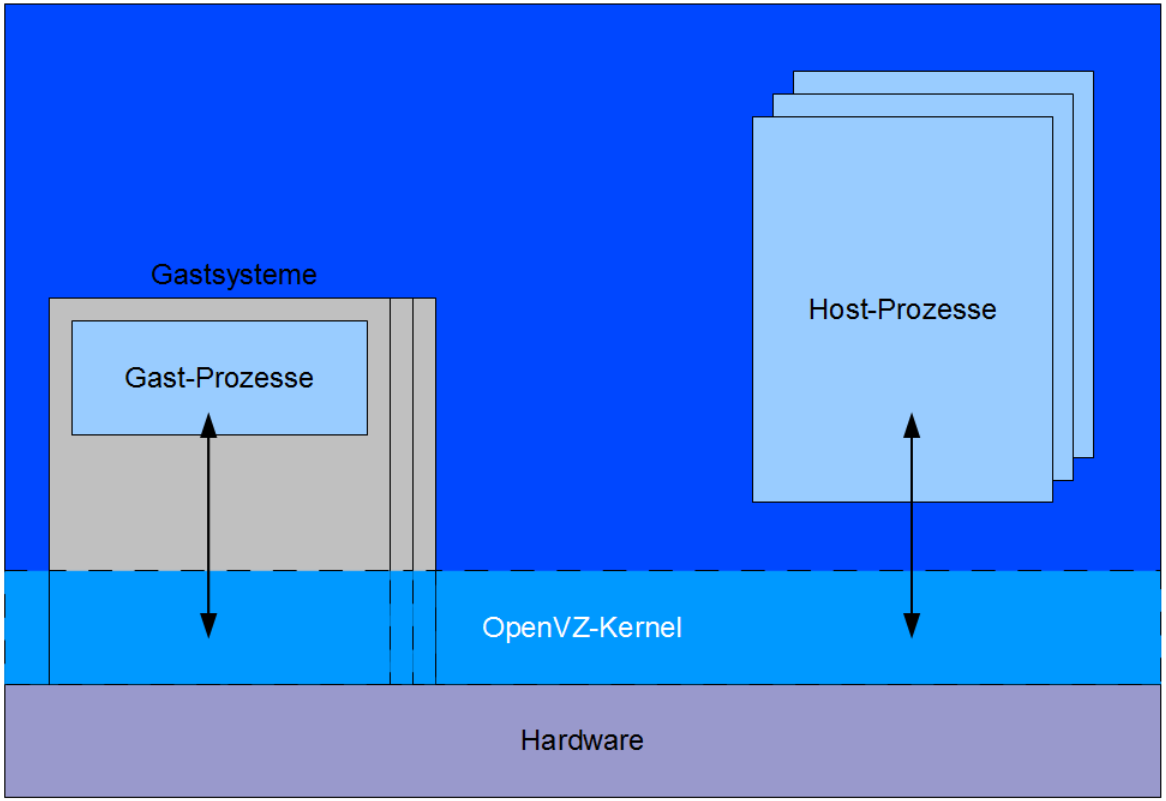
→ Gast-BS müssen nicht modifiziert werden
→ Gastsysteme frei wählbar
→ Intel und AMD haben keinen gemeinsamen Standard, dies kann zu Problemen bei der Migration von Gastsystemen führen
→ Virtualisierungsplattform muss diese Technologien unterstützen

Betriebssystemvirtualisierung

- Innerhalb des Host-BS werden isolierte Bereiche erzeugt, auch *Virtual Environment* (VE) oder *Container* genannt
- In einem VE ist kein eigenständiges Betriebssystem installiert, es verwendet die Kernel-Bibliotheken und Geräte-Treiber des Hostsystems
- Einige Individualdaten müssen für den Container definiert werden, z.B. Dateisystem, IP-Adresse, Hostname, Benutzer

→ Gute Performance durch geringen Virtualisierungsaufwand
→ Wenig Speicherbedarf, da kein komplettes BS installiert wird
→ Keine freie Wahl des Gast-BS (gebunden an das Hostsystem)

Betriebssystemvirtualisierung - OpenVZ



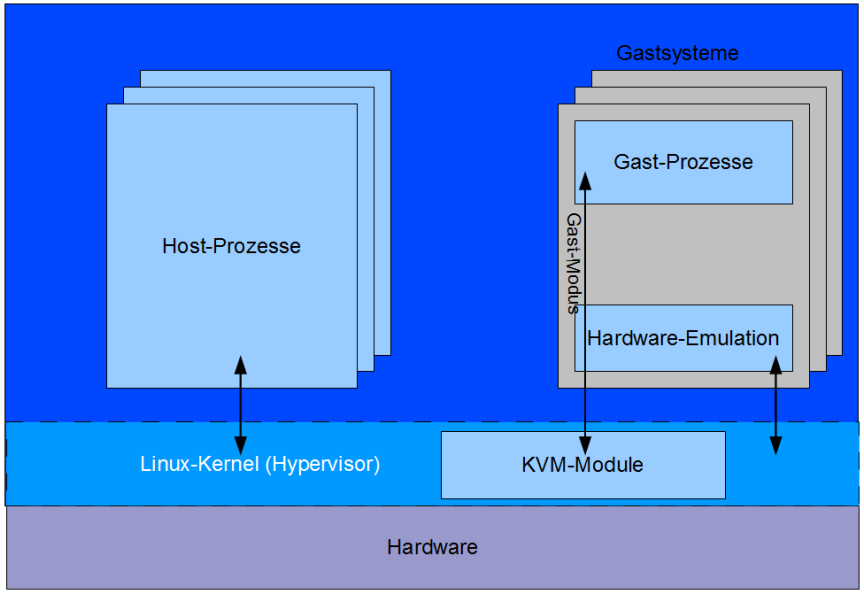
Hardware-Emulation

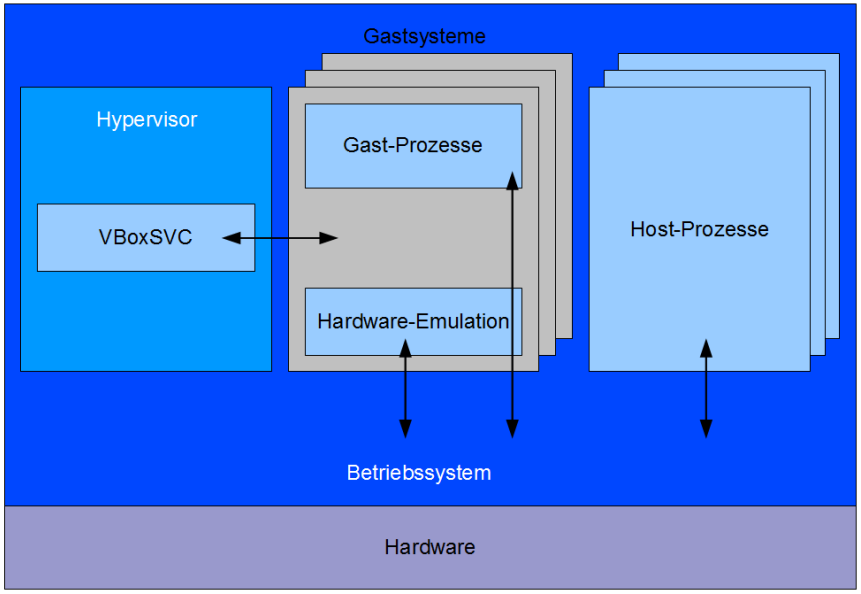
- Innerhalb einer VM wird die Standardhardware eines Rechners komplett oder teilweise simuliert
- Ein Emulator erzeugt entsprechende Softwareschnittstellen, die vom Gast-BS angesprochen werden können
- Der Emulator sorgt dafür, dass Befehle die an die simulierte Hardware gerichtet sind für die physische Hardware des Hostsystems umgewandelt werden

→ Flexible Wahl der Gast-BS

→ Performanceverlust durch hohen Virtualisierungsaufwand

Hardware-Emulation – KVM, VirtualBox





Zusammenfassung

	Paravirt.	HW-unterstützte Virt.	BS-Virt.	HW-Emulation
Performance	++	+	++	-
Gast-BS Flexibilität	-	+	-	+
Unmodifizierte Gastsysteme	--	+	-	+
Hardwareunabhängigkeit	+	-	+	++
Verbreitung	-	+	-	+

→ Gängige Desktop-Systeme wie VirtualBox oder VMware verwenden Hardware-unterstützte Virtualisierung in Kombination mit teilweiser Hardware-Emulation.

Virtualisierung
Was bringt Virtualisierung?

Vorteile

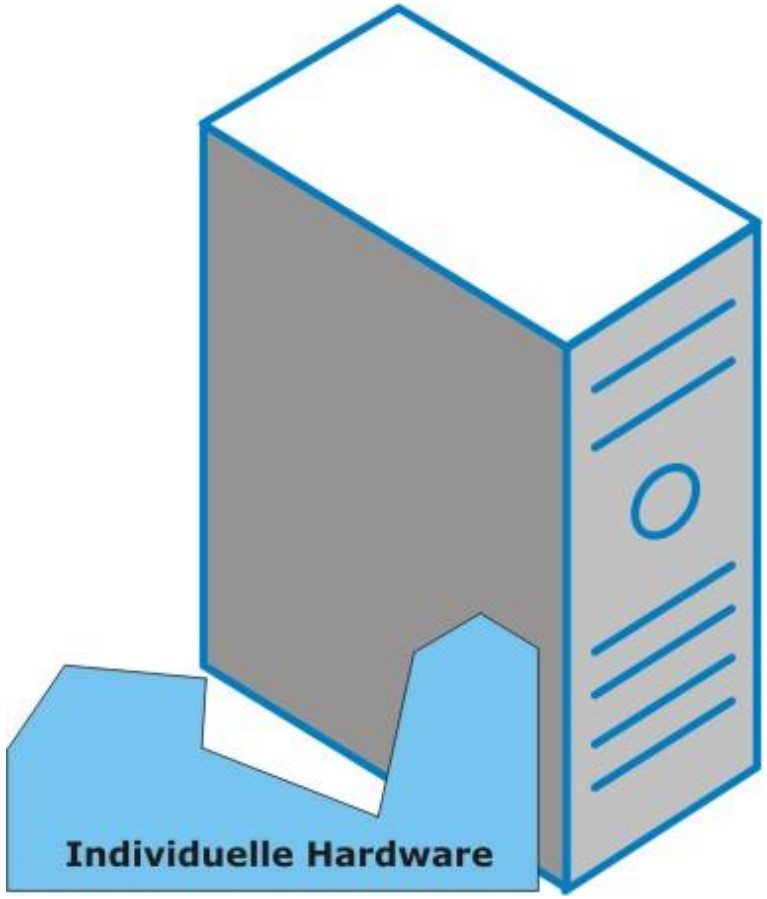
- Bessere Ausnutzung von Hardware
- Vereinfachte Administration von Hardware
- Vereinfachte Bereitstellung von Systemen
- Höhere Sicherheit durch zusätzliche Abstraktionsschicht
- Optimierung von Software-Tests
- Vereinfachte Abwärtskompatibilität
- Einfache Skalierung je nach Bedarf

Nachteile

- Leistungsverlust
- Ausfall von realer Hardware zieht Ausfall von mehreren virtuellen Systemen nach sich
- Overhead durch Verwaltung der virtuellen Maschinen
- Hardware, die leistungsfähig genug ist, ist teuer

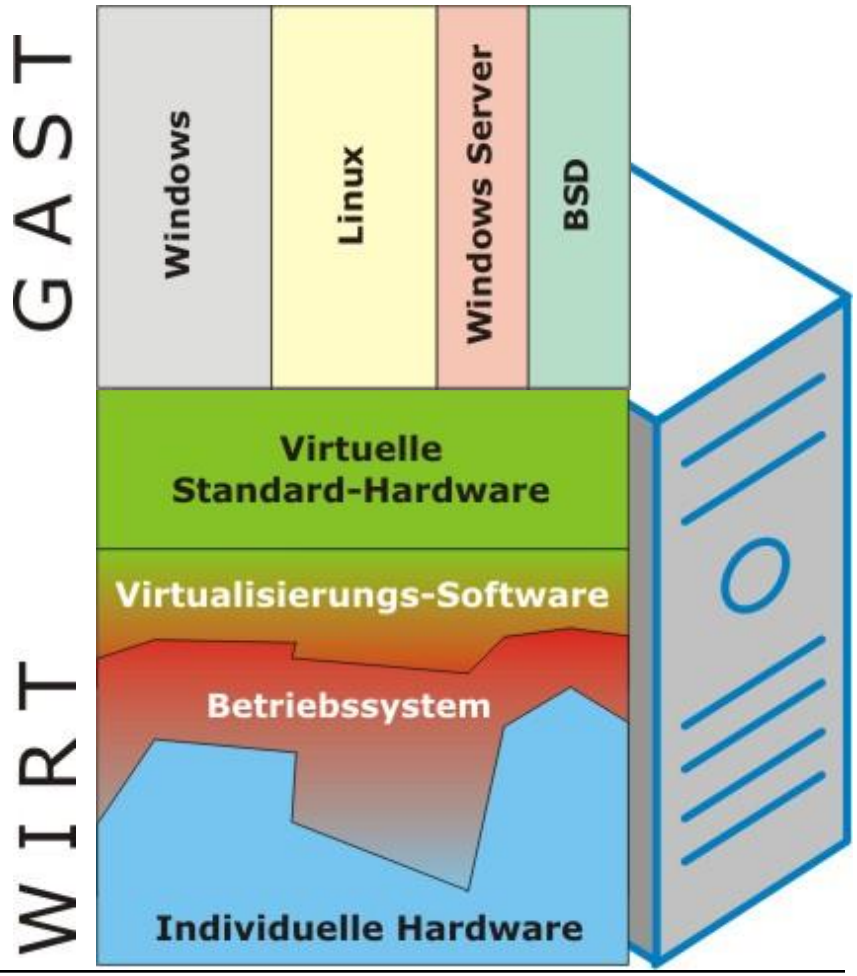
Virtualisierung
Ein flexibles Werkzeug

Physikalische Hardware ist immer individuell



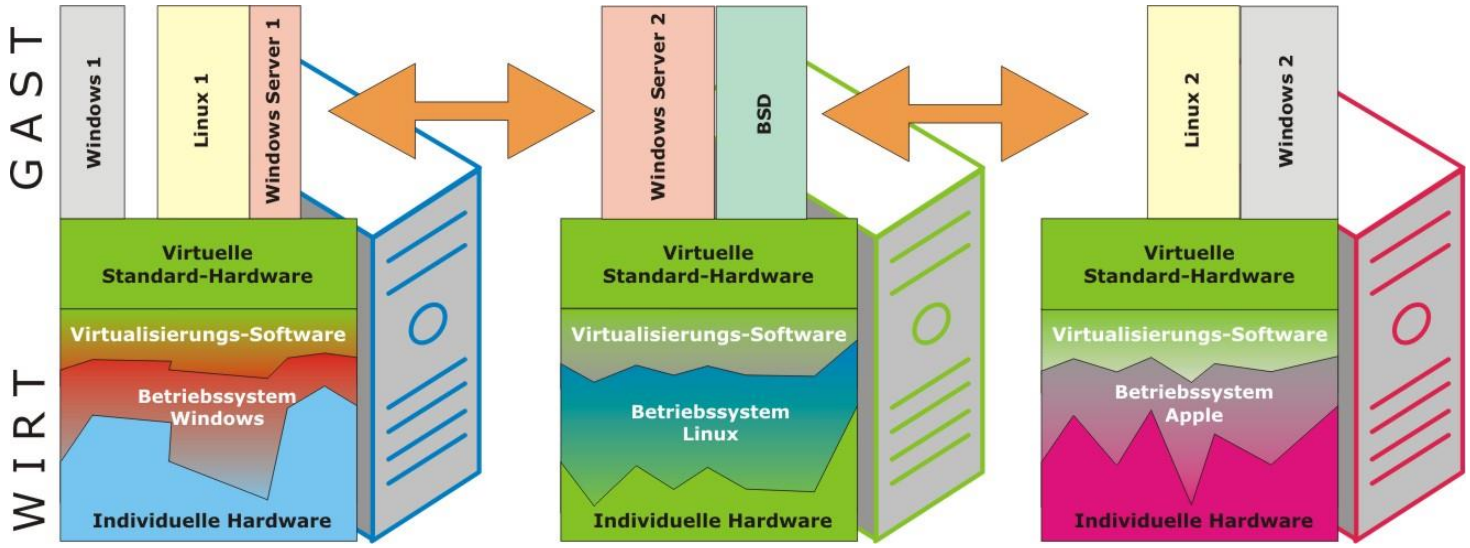
Virtualisierung
Ein flexibles Werkzeug

Physikalische Hardware ist immer individuell



Virtualisierung
Ein flexibles Werkzeug

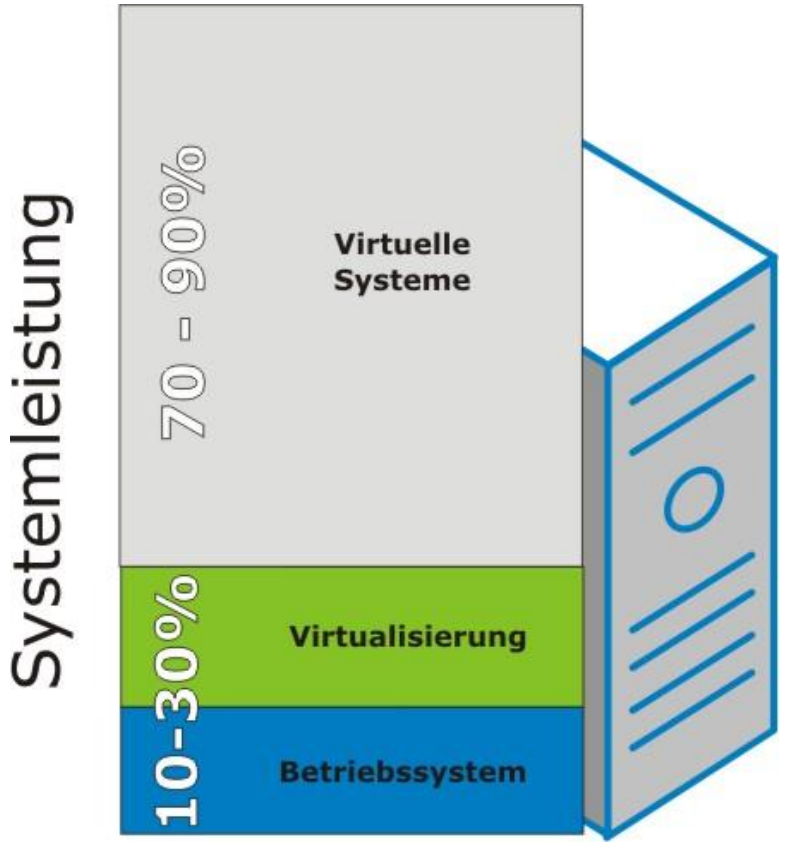
- ***Hohe Ausfallsicherheit***
  - *Virtuelle Maschine kann auf jeder Hardware betrieben werden*
  - *Schnelle Wiederinbetriebnahme*
- ***Einfache Wartung und Skalierung***
  - *Servertausch ohne aufwändige Neuinstallation*
  - *Hardwareanpassung ohne Downzeiten*
  - *Tests und Installationen ohne Risiko auf virtuellen Testsystemen*



Virtualisierung
Kostenlos?

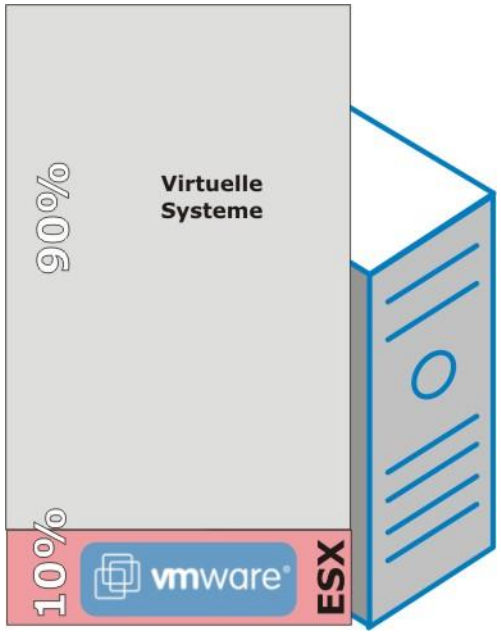
Leistungsverlust möglich:

- *Leistungsbedarf Betriebssystem und Virtualisierungsschicht bis zu 30%*
- *Stark abhängig von Typ und Anwendung des Gastsystem*
- *Verlust unterschiedlich je nach Komponentenauslastung*
  - > *Prozessor*
  - > *Arbeitsspeicher*
  - > *Festplatte*
  - > *Netzwerk*



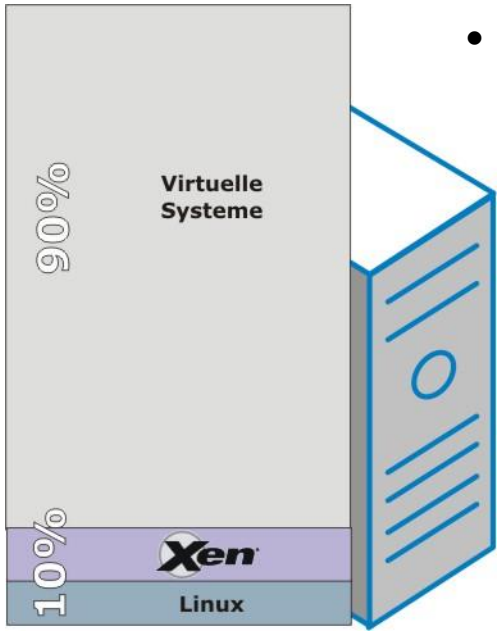
Virtualisierung
Kostenlos?

Optimierung: Speziallösungen mit nur ca. 10% Leistungseinbuße



Beispiel VMWare ESX: Betriebssystem und Virtualisierung in einem

- *Kommerziell*
- *High-End Lösung*
- *Nur zertifizierte Hardware*
- *Linux-Kern*



XEN: minimalistische Auslegung

- *Minimales Linux als Basis*
- *Stark mit dem Betriebssystem verwoben*
- *besonders hardwarenah*
- *Windows als Gast nur mit aktuellen Prozessoren*
  - *Intel Vanderpool*
  - *AMD Pacifica*

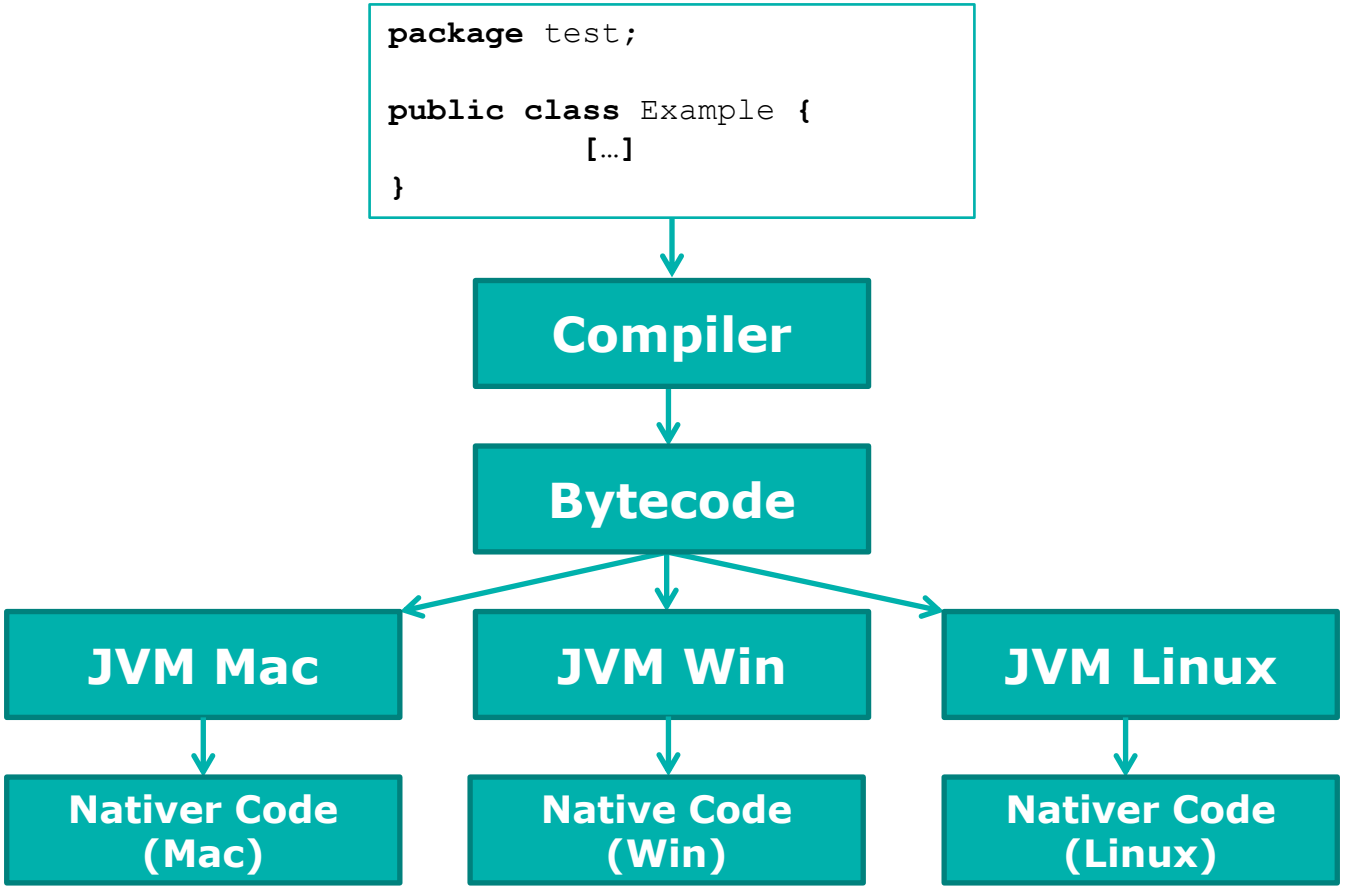
Weitere Virtualisierungsmöglichkeiten

- Anwendungsvirtualisierung (z.B. JVM)
- Partitionierung
- Virtual Hosting (Domaining)
- Virtuelle Netzwerke (VLAN, VPN)

Der Anwendung wird lokal eine virtuelle Umgebung bereitgestellt, die alle nötigen Komponenten enthält

- Die virtuelle Umgebung stellt tatsächlich vorhandene Ressourcen bereit, kapselt diese aber
- Die virtuelle Maschine befindet sich zwischen der Anwendung und dem Betriebssystem
- Vorteile
  - > Anwendungen können plattformunabhängig ausgeführt werden, sofern eine entsprechende virtuelle Maschine verfügbar ist
  - > Anwendungen werden zur Laufzeit überwacht, was z.B. Buffer Overflows verhindern kann
- Nachteile
  - > Die Ausführungsgeschwindigkeit der Anwendung ist geringer, als bei nativ kompilierten Programmen
- Beispiel: Java Virtual Machine (JVM)

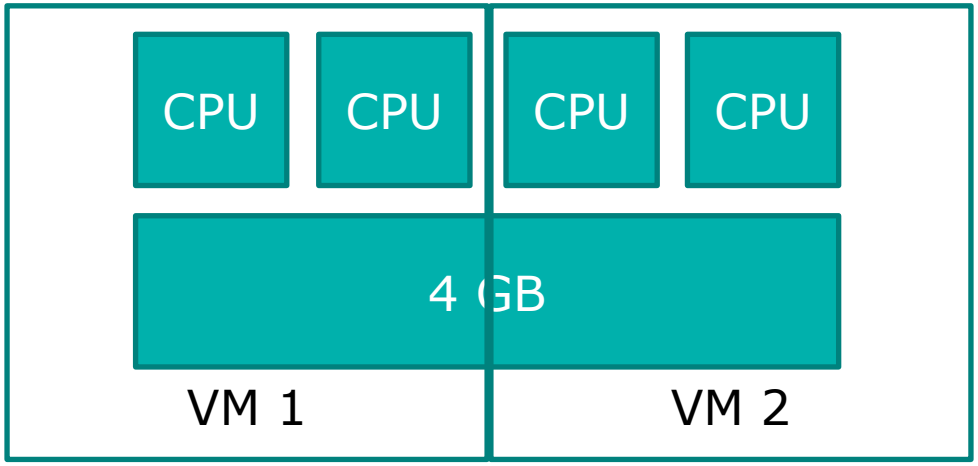
Virtualisierung
Anwendungsvirtualisierung



Bei der Partitionierung wird die Hardware eines Rechners in eigenständige Teilsysteme aufgeteilt

- Die einzelnen Komponenten werden über die Firmware des Rechners verwaltet und den VMs zugeteilt

- Einfaches Beispiel:
  - > Ein System mit 4 GB Ram und einer Quad-Core CPU wird in zwei Systeme geteilt



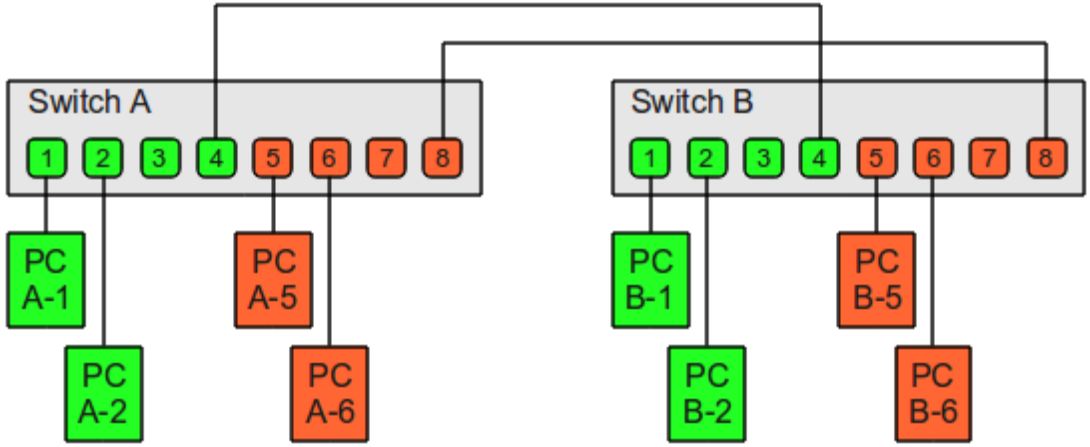
Virtualisierung
Virtual Hosting (Domaining)

***Virtual Hosting* ist das gleichzeitige Betreiben mehrerer Domains oder IP-Adressen auf einem Server.**

Man unterscheidet:

- IP-basiertes Virtual Hosting
  - > Ein Host mit mehreren Netzwerkschnittstellen
  - > Für jede Netzwerkschnittstelle können mehrere IP-Adressen konfiguriert werden
  - > Auf dem Host laufende Server/Dienste können sich an eine oder mehrere dieser IP-Adressen binden
  - > Für den Client ist nicht erkennbar, dass alle Dienste auf dem selben Host laufen
- Namensbasiertes Virtual Hosting
  - > Bietet die Möglichkeit den selben Dienst unter verschiedenen Domains anzubieten
  - > Unterscheidung wird auf Anwendungsebene (http, smtp,…) getroffen

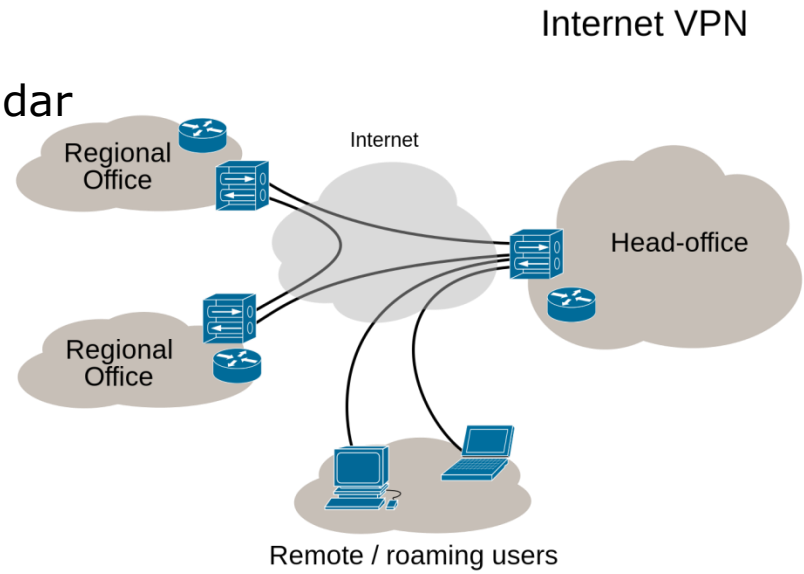
- VLAN (Virtual Local Area Network)
  - > Logisches Teilnetz innerhalb eines physischen Netzwerks
  - > Kann sich über einen oder mehrere Switches ausdehnen
  - > Trennung geschieht durch VLAN-fähige Switches (hardwarebasiert), die eine Weiterleitung der Datenpakete in andere Teile des physischen Netzwerks verhindern



Port-basiertes VLAN Quelle: http://www.thomas-krenn.com/de/wiki/VLAN

- VPN (Virtual Private Network)
  - > Per VPN lassen sich zwei private Netzwerke oder Hosts über ein öffentliches Netzwerk (Internet) hinweg verbinden
  - > Der Datenaustausch erscheint dabei so, als wären beide Netzwerke/Host direkt miteinander verbunden (point-to-point)
  - > Die Daten werden dabei auf ihrem öffentlichen Weg mittels Verschlüsselungstechnik gesichert
  - > VPN stellt eine reine Software-Lösung dar

→ Man unterscheidet drei VPN-Typen



End-to-Site-VPN

- Ein Heimarbeitsplatz oder mobiler Benutzer benötigt Zugriff auf ein Unternehmensnetzwerk
- Der externe Mitarbeiter verwendet einen VPN-Client der sich mit dem VPN-Gateway des Unternehmens verbindet
- Der externe Mitarbeiter kann arbeiten als wenn er sich im Unternehmen befindet

Site-to-Site-VPN

- Es werden mehrere lokale Netzwerke von Außenstellen oder Niederlassungen zu einem virtuellen Netzwerk zusammengeschaltet
- Die Router der jeweiligen Netze bauen untereinander VPN-Verbindungen auf
- I.d.R. werden so bestimmte Dienste fremder Unternehmen ins eigene Netzwerk integriert
- Dem externen Unternehmen wird somit der Zugriff auf Teilbereiche des eigenen Netzes gewährt

End-to-End-VPN

- Hierbei wird eine direkte Verbindung zwischen zwei Clients aufgebaut
- Der VPN-Tunnel deckt somit die gesamte Verbindung ab und die Verschlüsselung wird nicht am Eingangspunkt in das Netzwerk aufgelöst
- Beide Seiten benötigen entsprechende VPN-Software und Konfiguration

Virtualisierung
Flexibel gestalten: Cloud Computing

*"**Cloud Computing** is a style of computing in which massively scalable IT-related capabilities are provided "as a service" using Internet technologies to multiple external customers. "*

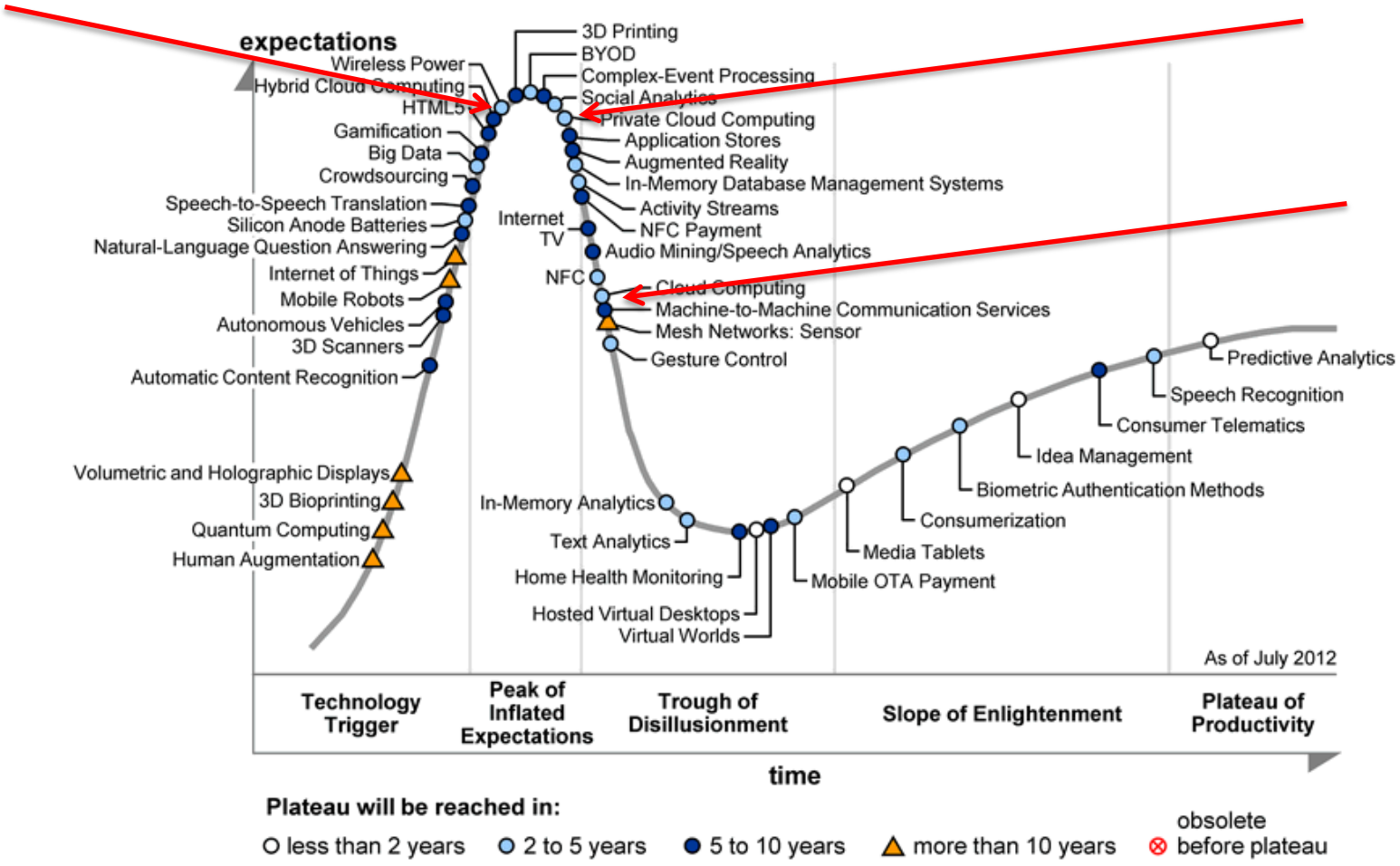
[Gardner]

*"**Cloud Computing** is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."*

[National Institute of Standards and Technology]

Cloud Computing
Was steck dahinter?

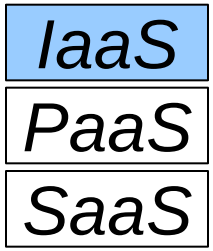
Cloud Compouting
Der Gartner Hype Cycle



SaaS
Software as a Service

PaaS
Platform as a Service

IaaS
Infrastructure as a Service

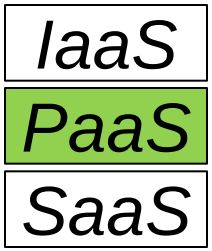


Infrastructure as a Service (IaaS)

- Bietet Rechen- und Speicherleistungen an, wie auch Netzwerkressourcen und anderes
- Aus Anbietersicht: Software zur Verwaltung virtueller Rechner
- Benutzer kann beliebige Software nutzen, bis hin zum Betriebssystem
- Administrative Kontrolle des virtuellen Rechners beim Benutzer

Beispiele:

- *Amazon WebServices*
- *Apple mobile.me*
- *Emulab*



Platform as a Service (PaaS)

- Virtualisierte Hosting-Umgebung für den Benutzer
- Programmiersprachen, IDEs, usw. werden vom Provider vorgegeben
- Benutzer kann ggf. Hosting-Umgebung einstellen und die Anwendungen aktivieren und konfigurieren (Deployment)

Beispiele:

- *Microsoft Azure Services Platform*
- *Bungee*
- *Google AppEngine*

IaaS
PaaS
SaaS

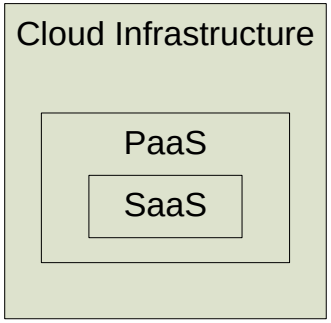
Software as a Service (SaaS)

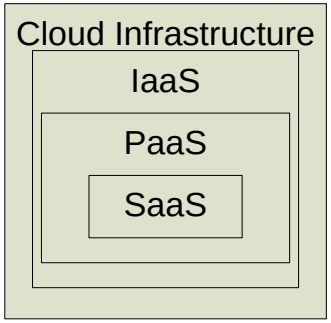
- Der Provider liefert dem Benutzer einen Zugriff auf eine Anwendung in der Cloud
- Zugriff über Remote Console oder Web-Browser möglich
- Benutzer kann nur Anwendungseinstellungen konfigurieren

Beispiel:

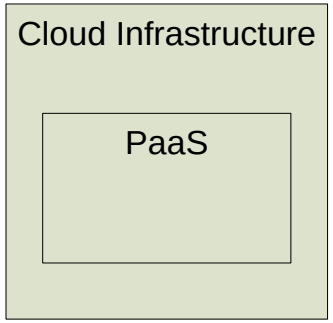
- *Oracle OnDemand*
- *Microsoft Office Live*
- *salesforce.com*

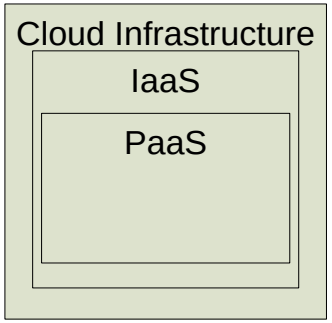
Cloud Computing
Dienstmodelle



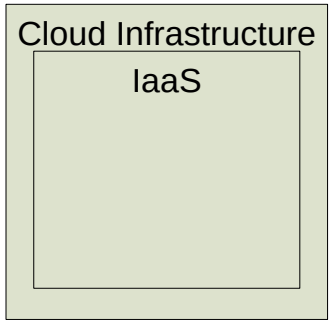


Software as a Service (SaaS) Architectures





Platform as a Service (PaaS) Architectures



Infrastructure as a Service (IaaS) Architectures

Cloud Computing
Dienstmodelle

= Managed for You	Standalone Servers	IaaS	PaaS	SaaS
Applications	✗	✗	✗	✓
Runtimes	✗	✗	✓	✓
Database	✗	✗	✓	✓
Operating System	✗	✗	✓	✓
Virtualization	✗	✓	✓	✓
Server	✗	✓	✓	✓
Storage	✗	✓	✓	✓
Networking	✗	✓	✓	✓

Private cloud

- *Exclusives für ein Unternehmen betriebenes "Rechenzentrum"*
- *Manche Spezifikationen fordern, dass Cloud-Provider und Nutzer sich in einem Unternehmen befinden, andere erlauben den Betrieb über einen externen Service-Provider*

Public cloud

- *Öffentlich verfügbare Infrastruktur*

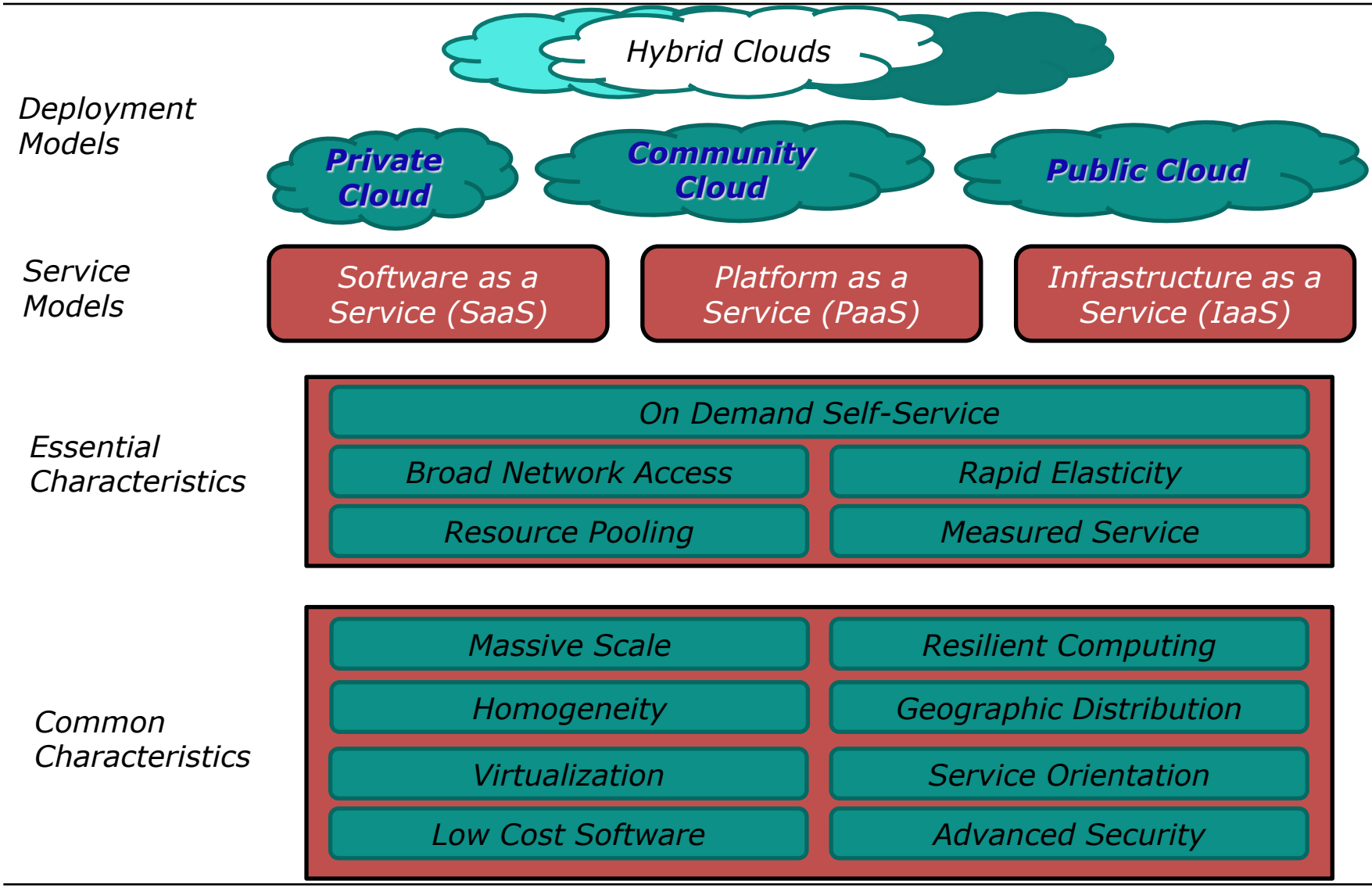
Community cloud

- *Geteilte Infrastruktur für eine spezifische Nutzergruppe, z.B. Ärzte, Anwälte oder Entwicklungsverbund*

Hybrid cloud

- *Zusammenwirken 2er oder mehrerer Clouds*

Cloud Computing
Der NIST-Framework



Cloud Computing
Was erhoffen sich die Nutzer?

- Reduktion der IT-Kosten
- Veränderte Kostenstrukturen (OpEx statt CapEx)
- Gewinn an Agilität und Flexibilität
- Innovation und Time-To-Market
- Erhöhte Professionalität und Verfügbarkeit
- Konzentration auf das Kerngeschäft
- Weltweiter Zugriff auf Daten
- Unterstützung von Kollaboration
- …

Cloud Computing
Was erhoffen sich die Provider?

- Neue Geschäftsmodelle und Distributionskanäle
- Economy of Scale
- Deckungsbeiträge
- Effiziente Bereitstellung der eigenen Software
- Kundenbindung
- …

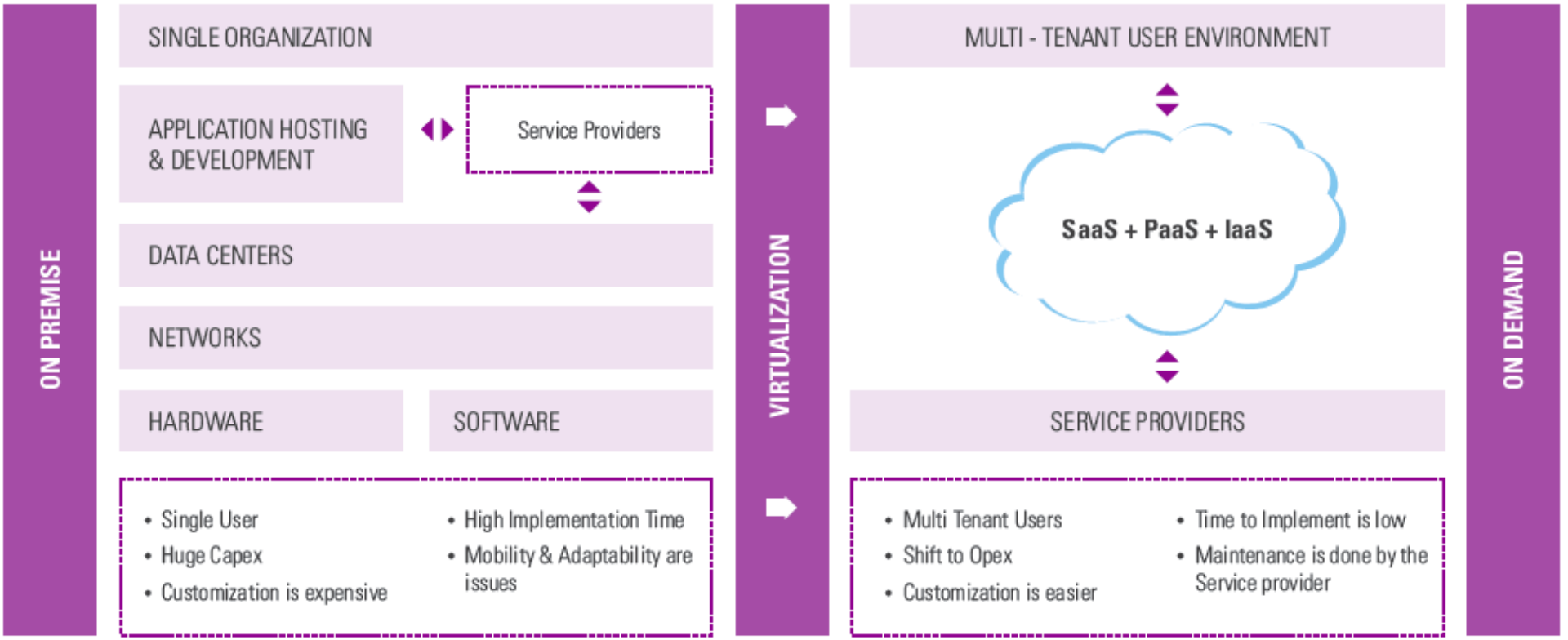
Cloud Computing
Wie arbeiten Nutzer eigentlich mit einer Cloud?

- Web-Schnittstelle/Ajax/HTML5
- Web Service / RESTful Service
- Entwicklungsumgebung mit Cloud-Deployment-Möglichkeit
- Remote-Desktop-Protokoll
- SSH
- Einbinden der Daten (WebDav, HDFS, S3, ODBC/JDBC, …)

Cloud Computing
Was benötigt man als Service-Provider?

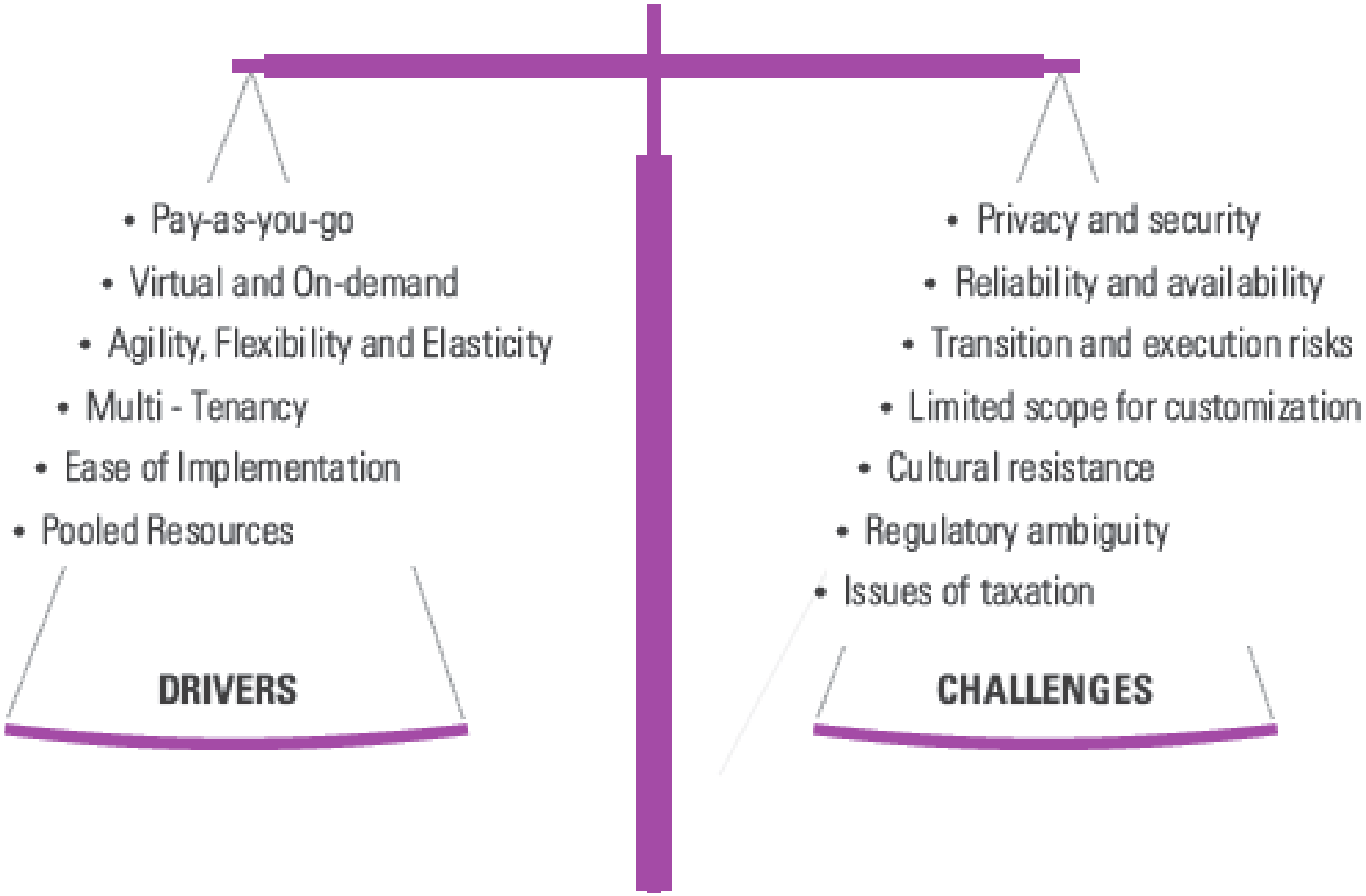
- Virtualisierungssoftware (+ Hardware) oder einen entsprechenden Anbieter
- Möglichkeiten des Resource Pooling (Kostenvorteile)
- Mandantenfähige Software
- Saubere Definition der Prozesse
- Schnittstellen für die Dienste (RESTful, Web Service, Web/Ajax, Remote Console)
- Ein Abrechnungsmodell und dessen Realisierung
- Tools (Governance, Monitoring, …)

Cloud Computing
Veränderte IT-Landschaft

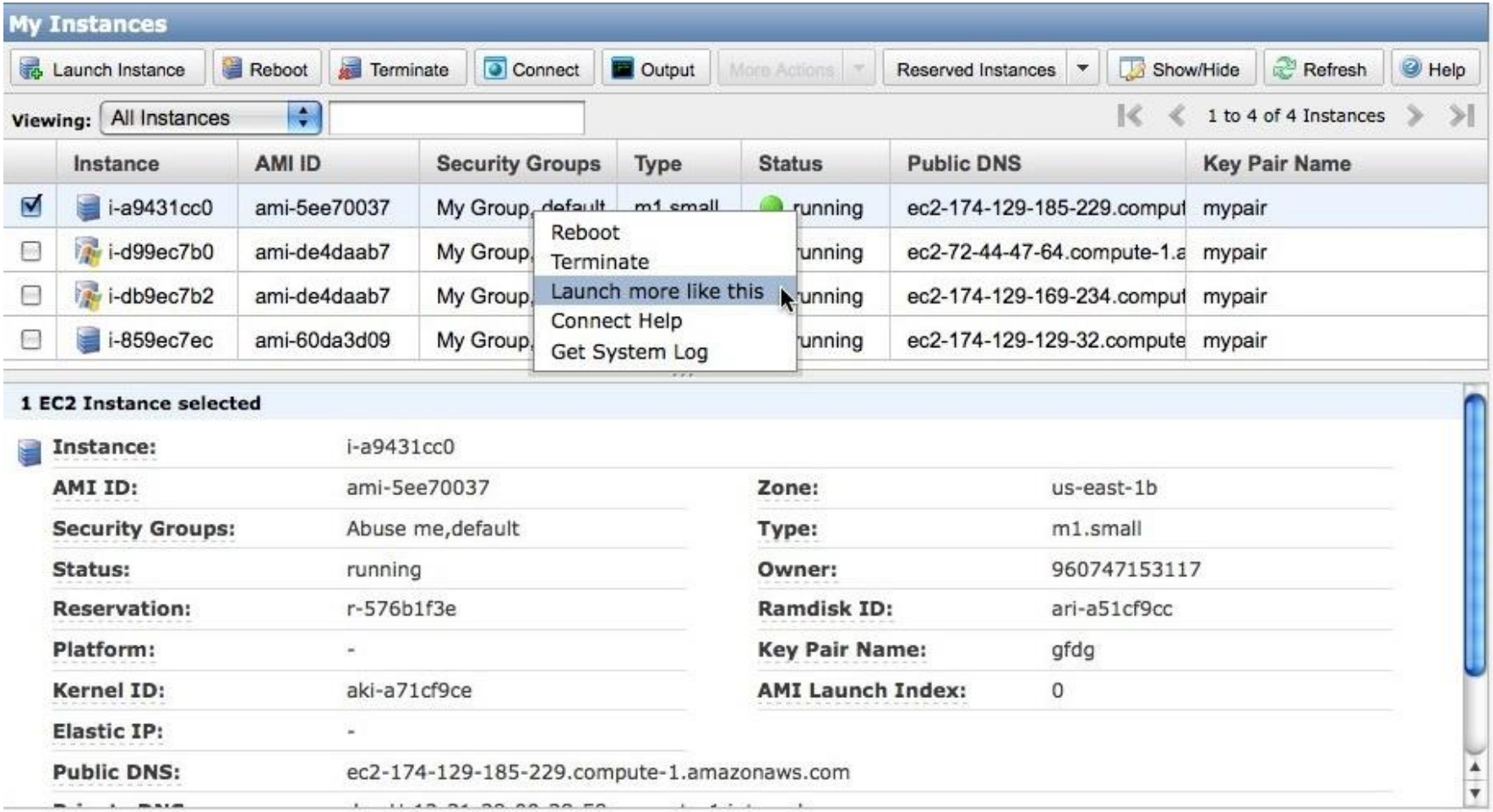


Source: KPMG's The Cloud: Changing the Business Ecosystem, 2011

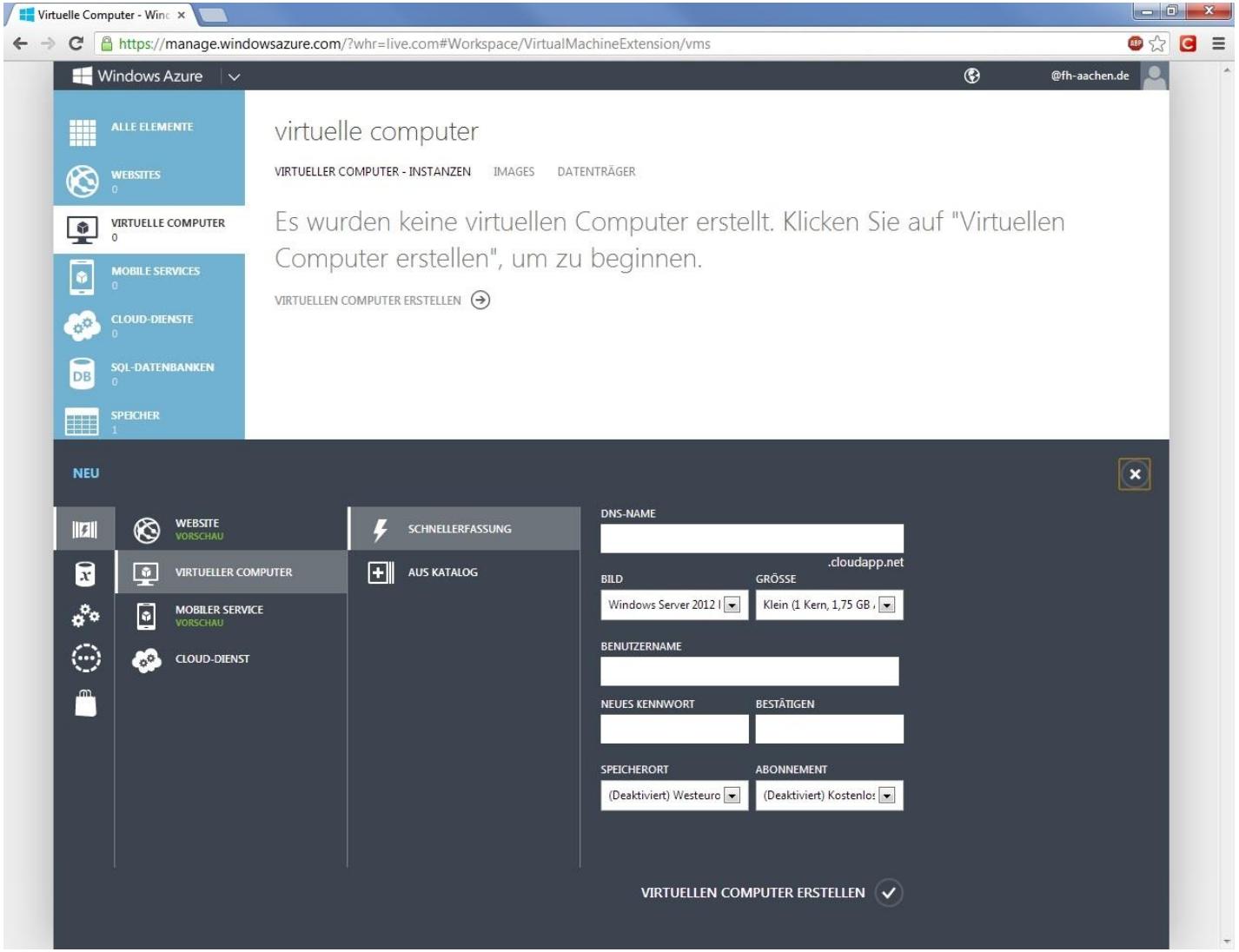
Cloud Computing
Veränderte IT-Landschaft

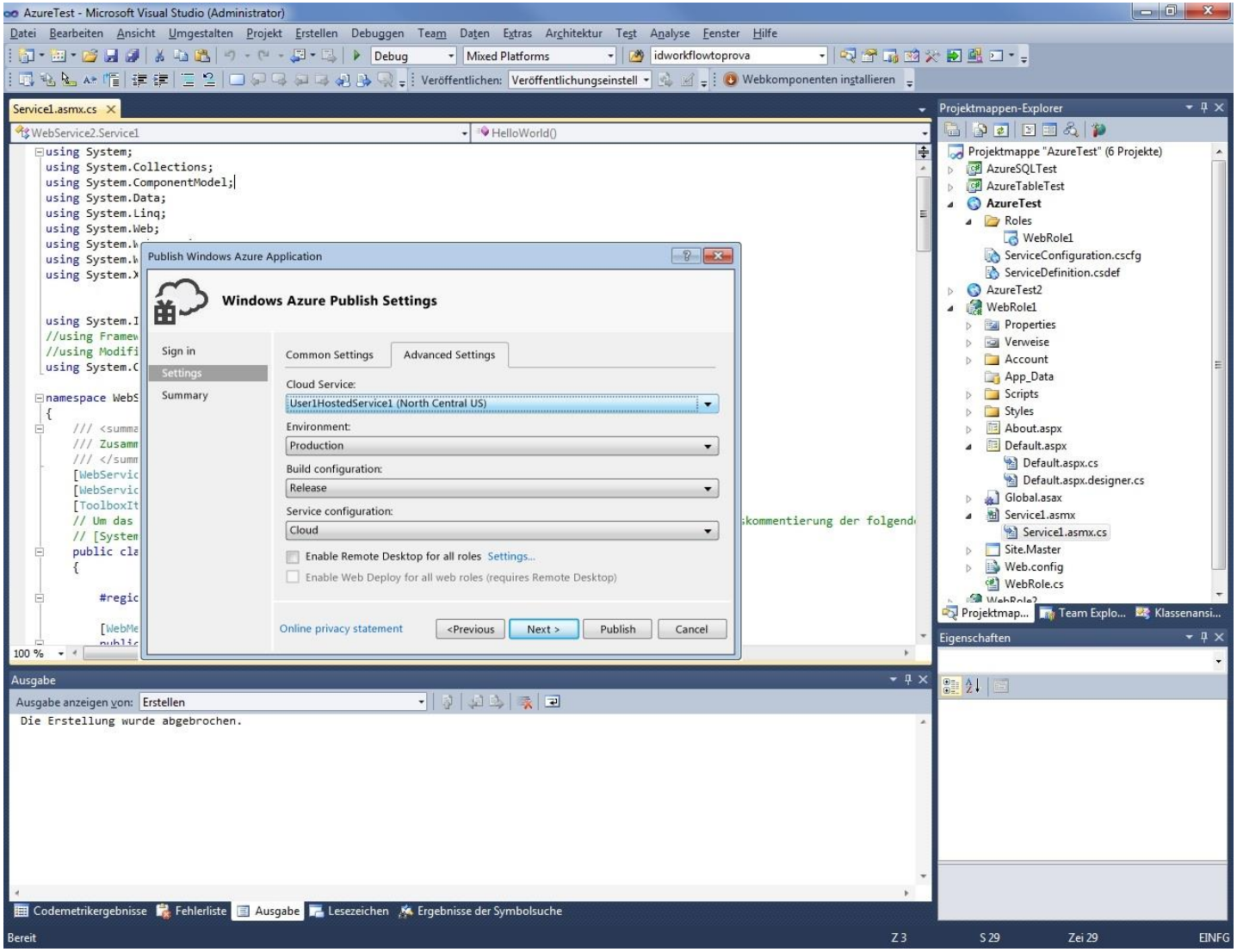


Cloud Computing
Amazon IaaS



Cloud Computing
Microsoft Azure




AzureTest - Microsoft Visual Studio (Administrator)
Publish Windows Azure Application
Windows Azure Publish Settings
Sign in
Settings
Summary
Common Settings
Advanced Settings
Cloud Service:
User1HostedService1 (North Central US)
Environment:
Production
Build configuration:
Release
Service configuration:
Cloud
Enable Remote Desktop for all roles Settings...
Enable Web Deploy for all web roles (requires Remote Desktop)
Online privacy statement
<Previous
Next >
Publish
Cancel
Projektmappen-Explorer
Eigenschaften
Ausgabe
Die Erstellung wurde abgebrochen.

- *Neben der Nutzung von SaaS-Angeboten sollte man auch die Chancen durch Deployment einer eigenen mandantenfähigen Anwendung über einen PaaS-Anbieter berücksichtigen*
  - *Einfachere Wartung und Rollout neuer Versionen*
  - *Oftmals besser gesicherter Betrieb*
  - *Neues Preismodell kann neue Kunden binden*
- *Der SaaS-Markt wird nach Gartner-Prognosen bedingt durch den Markt in China weiter wachsen*

- *Es gibt aber auch Probleme*
  - *Vendor-Lock-in*
  - *Service Level Agreements passen nicht*

Cloud Computing
Aber nicht ganz einfach: Rechtliche Anforderungen

- Rechtlicher Rahmen des Vertrages mit einem Anbieter
  - Zumeist typengemischt mit Mietcharakter (Storage), Werkcharakter (Implementierungen) Dienstcharakter (Support)
  - **Exit-Szenarien**: Was passiert bei einer Insolvenz des Anbieters?
  - Dokumentation und Nachweisbarkeit der vom Anbieter erbrachten Leistungen
  - Offenlegung der Standorte
- Lizensierungsfragen, insbesondere bei SaaS
  - SaaS-Anbieter verkaufen oft eine Dienstleistung -> Keine Lizenz!!!!
  - Browser-Cache bedingt Kopierbarkeit im Sinne des Urheberrechts
  - Es ist umstritten, ob Open Source Software unter GPLv2 in einer Cloud öffentlich zugänglich gemacht werden dürfen
- Datenschutz schon bei Kundendaten oder Mitarbeiter-Login
  - Nur durch Datenübermittlung (§ 28 BDSG) oder einer **Auftragsdatenverarbeitung** (§8 u. §11 BDSG) in die Cloud möglich
  - Vorsicht bei Auslagerung aus dem EWR (Drittstaaten-Cloud)! Safe Harbor
- Steuer- und handelsrechtliche Anforderungen (Compliance)

Cloud Computing
Aus dem Eckpunktepapier des BSI

- Definiertes Vorgehensmodell für alle IT-Prozesse (ITIL)
- Ausreichende Informationssicherheit durch Sicherheits-architektur, die zumindest den IT-Grundschutz abdeckt
  - Verschlüsselte Kommunikation mit 2-Faktor-Authentifizierung (Wissen, Besitz)
  - Schutz vor Trojanern, DDoS-Abwehr
  - Sichere Isolierung der Anwendungen und Kundendaten (z.B. auch durch den Einsatz zertifizierter Hypervisoren)
  - Zuverlässiges Löschen der Daten
  - Möglichst RBAC-konforme Zugangskontrollen
  - Einhaltung von Sicherheits-Mindeststandards der zur Verfügung gestellten Webanwendungen
- Kunden müssen die Möglichkeit haben, messbare Größen, wie im SLA vereinbart, zu überwachen
- Standardisierte oder offen gelegte Schnittstellen

Cloud Computing
Ja oder Nein? Leitfaden der BITKOM

- Ermittlung der eigenen IT-Kosten
  - Gebäude, Infrastruktur, Lizenzen, Betrieb
  - Schulungen, Management und Verwaltung
- Kosten der Einführung
  - Kosten der Provider-Selektion (inkl. Anwaltskosten)
  - Migrationskosten (Daten, Arbeitsablauforganisation, Integration)
  - Monitoring (ggf. Verschiebung)
  - Schulung (ggf. Verschiebung)
- Laufende Kosten
  - Daten- und Netzwerkvolumina
  - Transaktionen
  - Service Instanzen oder IP-Adressvergabe
- Bewertung, z.B. nach Check-Liste