



Lernfeld 10b.2

Virtualisierung

- 10b.2.1** Was ist Virtualisierung
- 10b.2.2** Hardware-Virtualisierung
- 10b.2.3** Software-Virtualisierung
- 10b.2.4** Desktop-Virtualisierung
- 10b.2.5** Container-Virtualisierung



Designing Education
Connecting People

Virtuell:

- Nicht echt, nicht in Wirklichkeit vorhanden, aber echt erscheinend ...
- Stimmt das wirklich?

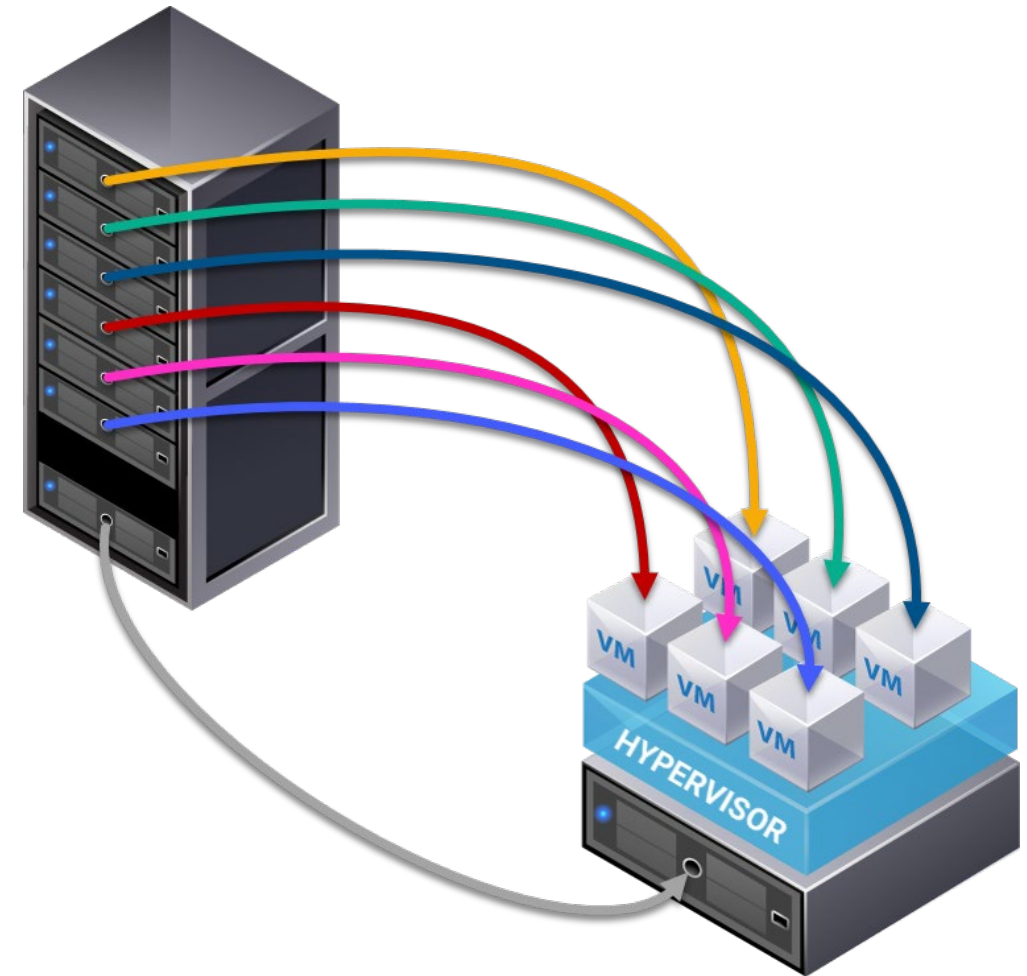


10b.2.1 Was ist Virtualisierung?

10.2.1 Was ist Virtualisierung?

Definition

- Als Virtualisierung bezeichnet man die Nachbildung eines Hardware- oder Softwareobjekts durch ein ähnliches Objekt vom selben Typ
- Dadurch lassen sich virtuelle (nicht physische) Geräte oder Dienste, wie z. B. Hardware, Betriebssysteme, Datenspeicher oder Netzwerkressourcen erzeugen
- Virtualisierung erlaubt es Ressourcen transparent zusammenzufassen oder aufzuteilen oder ein Betriebssystem innerhalb eines anderen auszuführen



10.2.1 Was ist Virtualisierung?

Abgrenzung: Obwohl sich die drei Konzepte ähneln, muss hier unterschieden werden!

Virtualisierung

- Abstraktion realer Hardware
 - Die realen Ressourcen sollen auf virtueller Ebene bereit gestellt werden
 - Damit können sie flexibel und bedarfsgerecht bereit gestellt werden

Emulation

- Stellt die Funktionen des realen Gegenstücks zur Verfügung
 - Hard- und Software werden nachgebildet
- Soll dieselben Ergebnisse erzielen, wie das Original
 - z. B. Konsolenemulationen (SNES, Playstation usw.) oder Android-Emulation auf Desktop-Computern

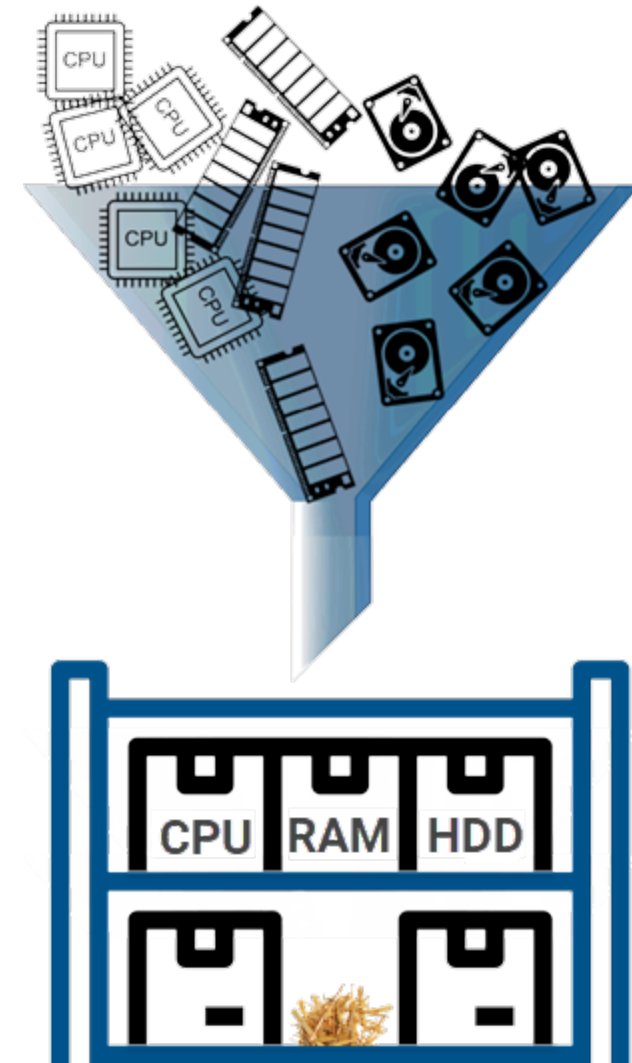
Simulation

- Vollständige Nachbildung eines Systems mittels Software
- Software, die in einer Simulation läuft, ist nicht für den Praxiseinsatz vorgesehen
- (Ein Flugsimulator simuliert den Flug, der Pilot erhebt sich dabei aber nicht in die Luft)

10.2.1 Was ist Virtualisierung?

Arten der Virtualisierung

- Häufig versteht man unter Virtualisierung die Nutzung von virtuellen Computern: VMs (virtual machines)
- Es gibt in der IT aber auch viele andere Möglichkeiten, z. B.:
 - Storage Virtualisierung:
 - z. B. RAIDs, Partitionen, Pools
 - Netzwerk Virtualisierung:
 - z. B. VLAN und VPN

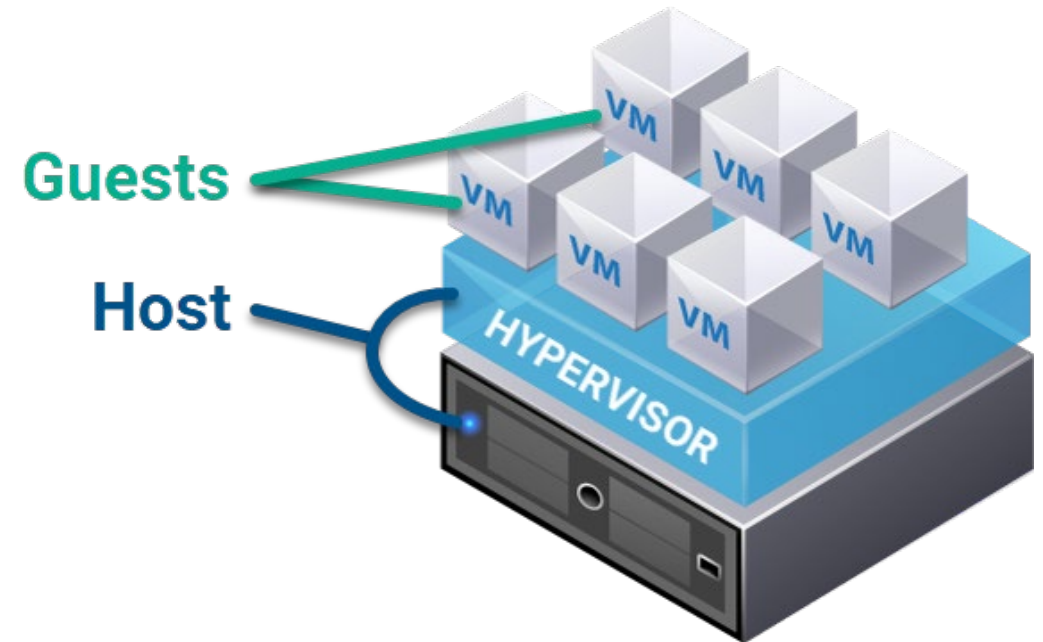


10.2.1 Was ist Virtualisierung?

Gründe für Virtualisierung

○ Isolierung von Applikationen

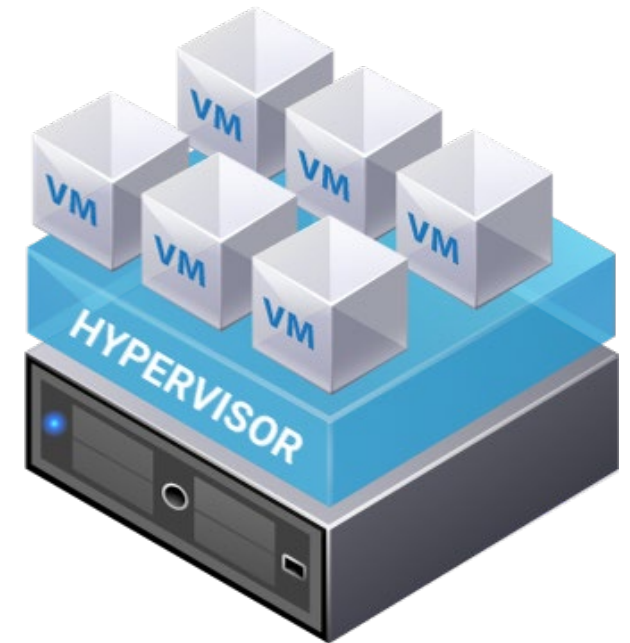
- Um Dienste und Applikationen sauber laufen zu lassen, wurden diese lange Zeit auf separater Hardware installiert
- Dadurch wird sichergestellt, dass die Programme sich nicht gegenseitig stören, oder dass der Absturz eines Programms oder dessen OS nicht alle Anwendungen betrifft
- Durch Virtualisierung kann man auf einer Hardware jeder Anwendung ihre eigene (von den anderen getrennte) Umgebung zur Verfügung stellen
- Sollte ein Programm verrückt spielen, beeinflusst das die anderen Umgebungen nicht



10.2.1 Was ist Virtualisierung?

Virtualisierung per Hypervisor

- Mehrere Software-Systeme laufen auf einer Hardware
 - Das können z. B. unterschiedliche OS sein
- Anstatt viele Computer mit wenig Auslastung laufen zu lassen, kann man so die Ressourcen parallel für weitere Systeme nutzen
- Da die Virtualisierung noch viele weitere Vorteile bringt, kann es sinnvoll sein, bestimmte Systeme von vornherein als virtuelle Systeme zu planen und dementsprechend starke Hosts zu verwenden



10.2.1 Was ist Virtualisierung?

Vorteile

- Kosten und Energieeinsparung
 - Hardware kann besser ausgenutzt werden
- Einfache Skalierung
 - Reale Ressourcen lassen sich je nach Anforderung auf die VMs verteilen
- Einfrieren von VMs durch Snapshots
 - Dadurch lassen sich u. a. Änderungen einfach rückgängig machen
- Einfaches Klonen von VMs
 - Dadurch lassen sich neue VMs sehr schnell bereitstellen

Nachteile

- Fällt das Host-System aus, betrifft das alle darauf gehosteten VMs
- VMs müssen sich die realen Ressourcen teilen
 - Verbraucht eine VM viel Leistung, kann das die anderen VMS beeinflussen



Designing Education
Connecting People

Wir täuschen Software echte Hardware vor



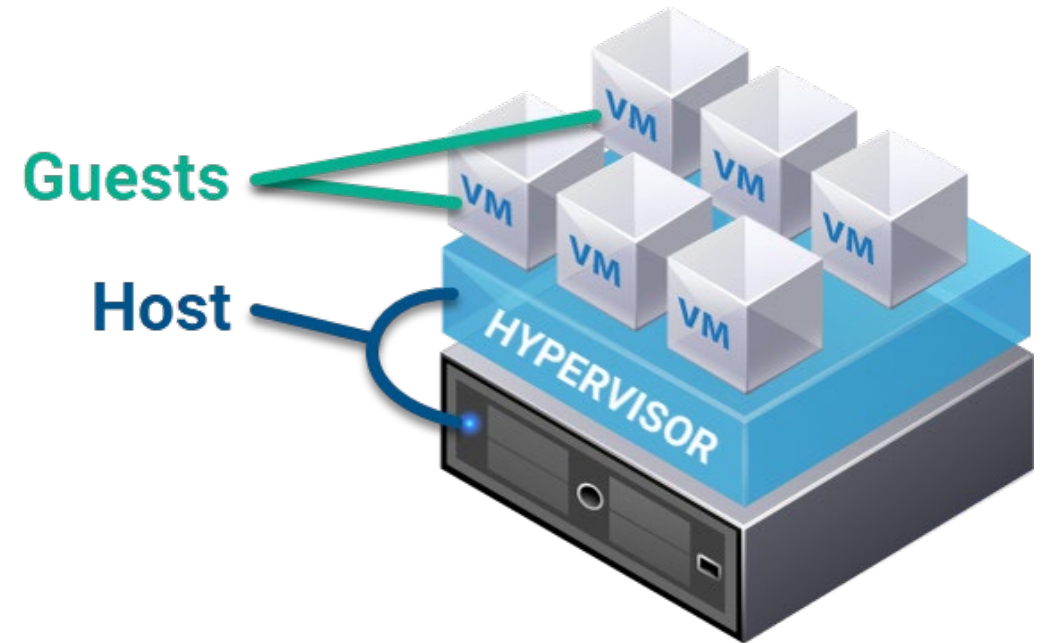
10b.2.2 Hardware- Virtualisierung

10.2.2 Hardware-Virtualisierung

Host und Guest

Zur Unterscheidung echter und virtueller Umgebungen werden die Systeme als Host oder Guest bezeichnet

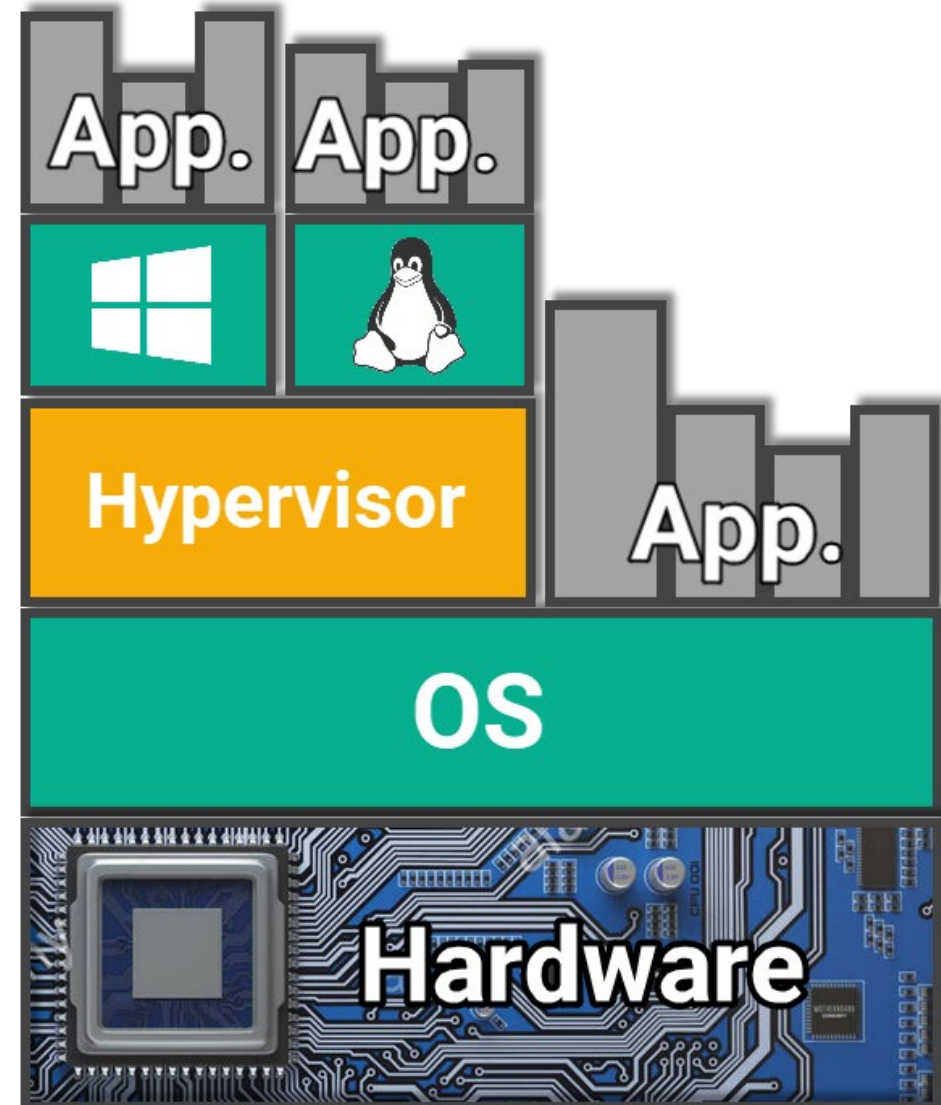
- **Host** (Wirt):
 - Ist die Ebene, die der echten Hardware am nächsten ist
- **Guest** (Gast):
 - Ist die auf dem Wirt ausgeführte Umgebung



10.2.2 Hardware-Virtualisierung

Hypervisor Type 2: Hosted

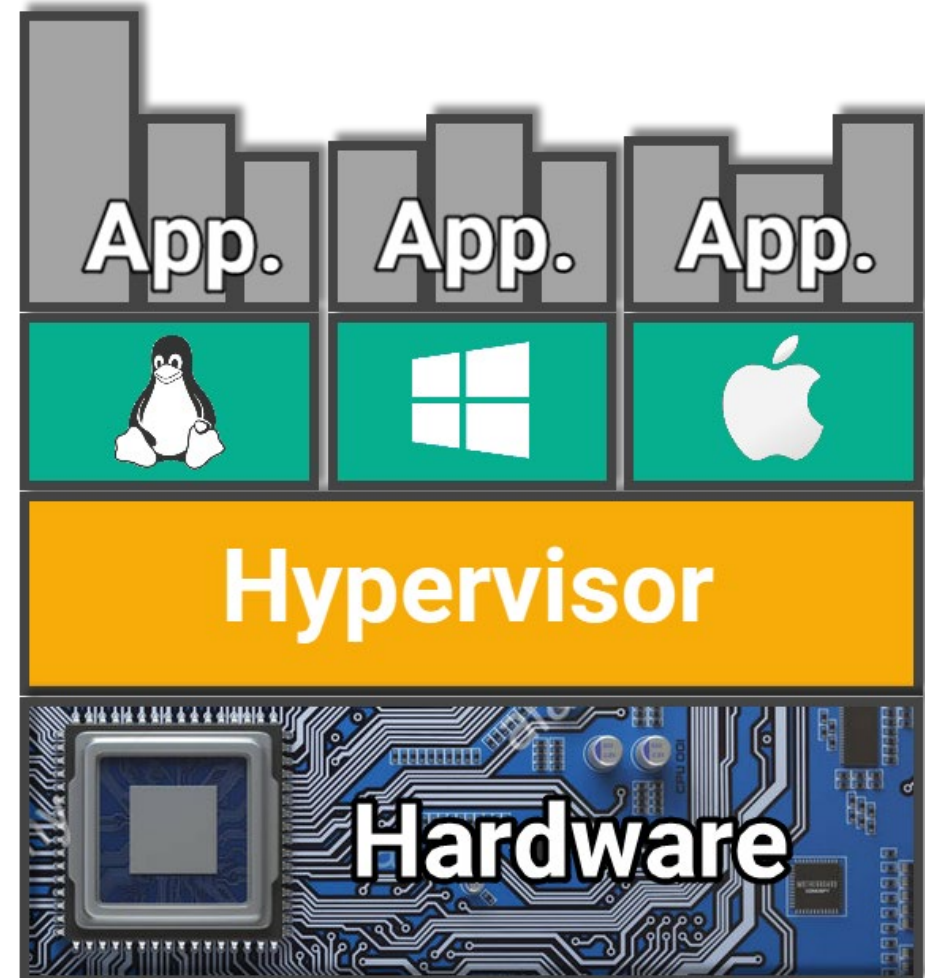
- Der Type 2 wird privat am häufigsten genutzt
- Auf dem Hostsystem wird ein vollwertiges Betriebssystem installiert
 - Auf dem Betriebssystem läuft, neben anderen Programmen und Diensten, ein Hypervisor
 - Auf diesem lassen sich VMs erstellen
- + Kann auf Standardhardware umgesetzt werden
- Das eigentliche OS des Hostsystems verbraucht viele Ressourcen
- Beispiele: Virtualbox, VMware Workstation



10.2.2 Hardware-Virtualisierung

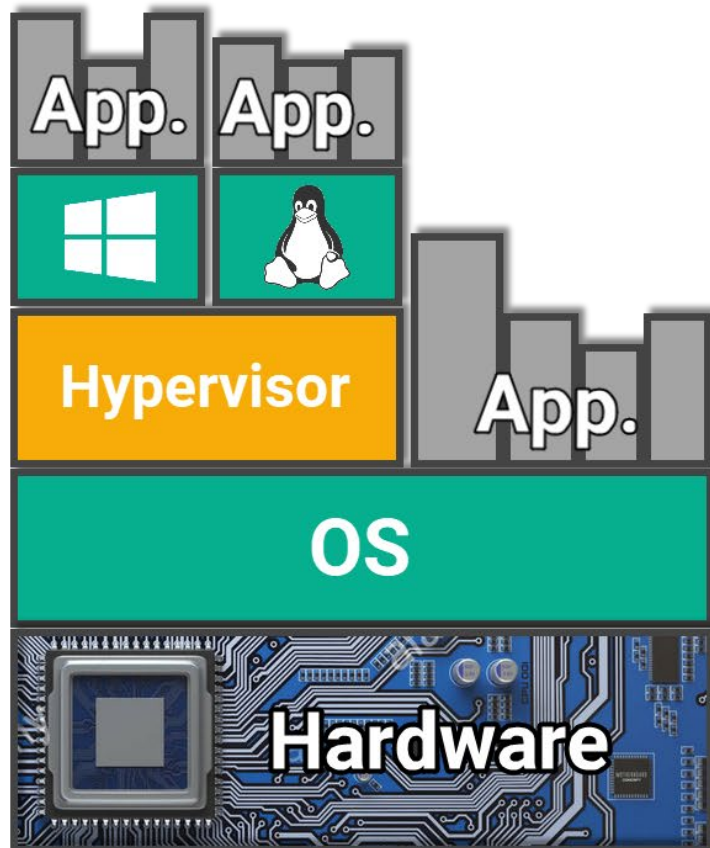
Hypervisor Type 1: Bare Metal

- Der Type 1 wird eher in Unternehmensumgebungen genutzt
- Der Hypervisor setzt direkt auf eine spezielle Hardware auf
- + Da die Hardware kein zusätzliches Betriebssystem versorgen muss, werden Ressourcen gespart
- + Die VMs können ohne Umweg auf das OS zugreifen
- Diese Lösung ist durch spezielle Hardware und Lizenzen oft viel teurer
- Beispiele: Hyper-V, VMware ESXi

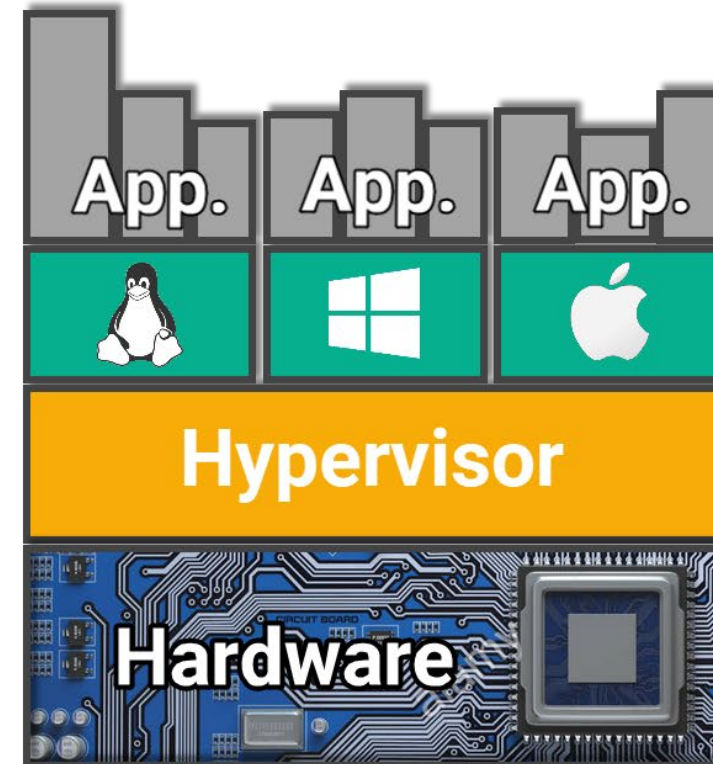


10.2.2 Hardware-Virtualisierung

Hypervisor Type 2: Hosted



Hypervisor Type 1: Bare Metal



10.2.2 Hardware-Virtualisierung

Voll-Virtualisierung

- Der Hypervisor spielt jeder VM eine komplette Hardwareumgebung vor
- Das Gastsystem weiß nichts von der echten Hardware
 - oder dass es überhaupt auf virtueller Hardware läuft
- + Es lässt sich praktisch jedes OS virtualisieren, da es echte Hardware vorgetäuscht bekommt
- Die Virtualisierung verbraucht viele Ressourcen, da für jede VM alle benötigten Ressourcen bereitgestellt werden müssen

Para-Virtualisierung

- Der Hypervisor stellt dem Gast-OS eine API (**A**pplication **P**rogramming **I**nterface) zur Verfügung
 - über diese können sie die echte Hardware des Hosts direkt ansprechen
- Voraussetzung ist, dass das Gast-OS per API mit dem Hypervisor kommunizieren kann
 - Windows kann das z. B. nicht
- + Bessere Performance als bei der Voll-Virtualisierung
- Es lassen sich nur modifizierte Gastsysteme (Kernel) virtualisieren



Designing Education
Connecting People

Jetzt ist nicht mal mehr die Software echt ...

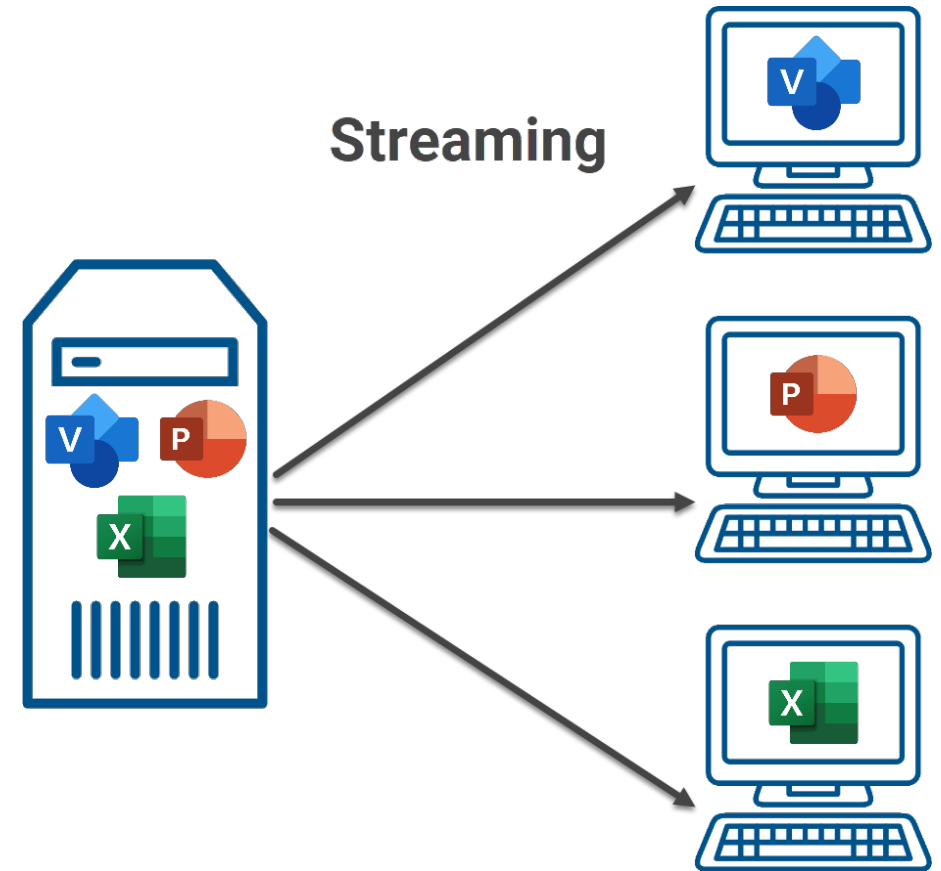


10b.2.3 Software – Virtualisierung

10.2.3 Software-Virtualisierung

Application Virtualization

- + Anwendungssoftware kann zentral bereitgestellt, verwaltet und gewartet werden
- + Durch die Isolierung der Anwendung ist das darunterliegende System vor Schadcode geschützt
- + Da sie nicht installiert wird, bleiben keine Rückstände auf dem System zurück
- Einige Anwendungen lassen sich nicht virtualisieren
 - z. B. weil sie eng mit dem OS arbeiten oder spezielle Treiber benötigen
- Lizenzierung der virtualisierten Software ist häufig nicht eindeutig geklärt



Ein Konzept, bei dem ganze Desktop-Umgebungen zentral bereit gestellt werden

Die Clients greifen über das Netzwerk auf die eigentlichen Arbeitsumgebungen zu

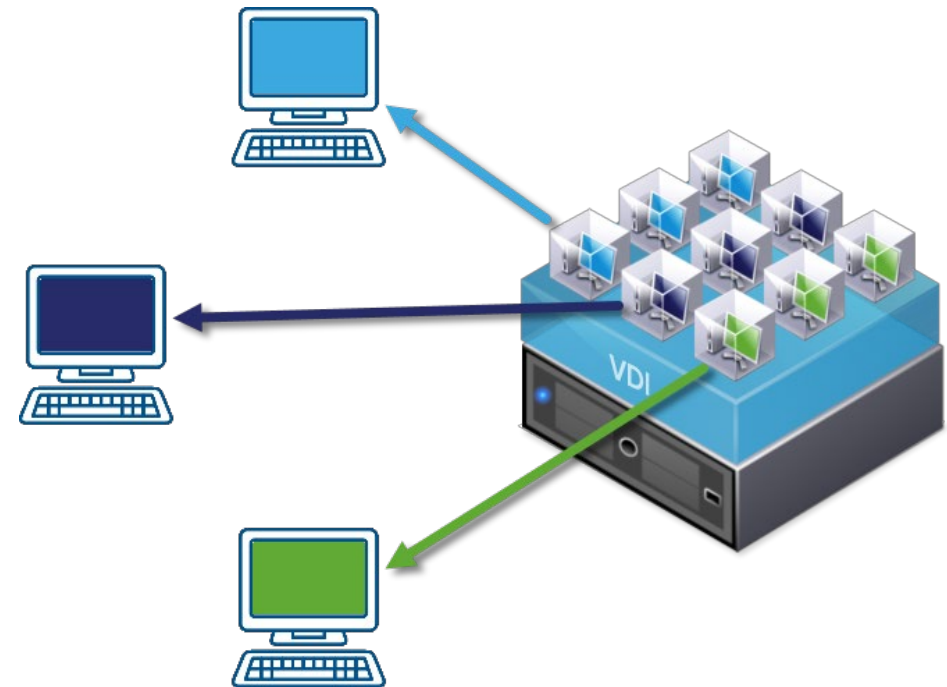
- Sie sind nur noch das Medium, um die Inhalte zu empfangen und anzuzeigen sowie Eingaben weiterzuleiten



10b.2.4 Desktop – Virtualisierung

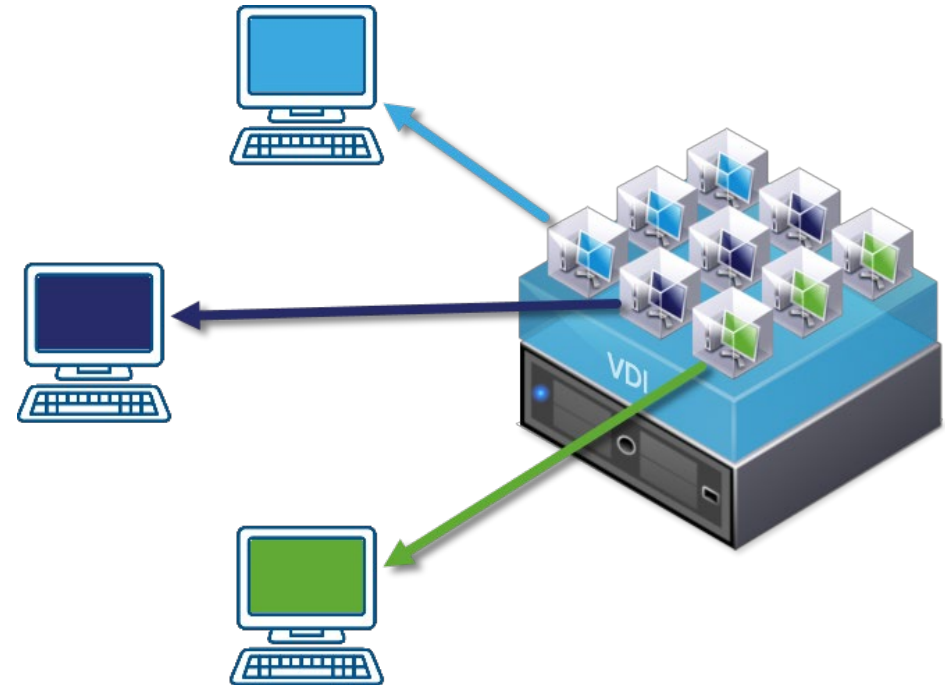
10.2.4 Desktop – Virtualisierung

- Statt jedem Arbeitnehmer einen komplett eingerichteten Arbeitsplatz zur Verfügung zu stellen, werden diese in einer virtuellen Umgebung eingerichtet
- Auf diese kann man sich, solange man das Netzwerk erreicht, von überall verbinden
 - Mitarbeiter sind nicht mehr an einen Platz gefesselt, sie können sich von überall auf „ihren“ virtuellen Computer verbinden
- Dafür wird keine leistungsstarke Hardware benötigt, es reichen Thin- oder Zeroclients
- Führende Anbieter:
 - Microsoft, VMware und Citrix



10.2.4 Desktop – Virtualisierung

- + Ermöglicht die zentrale Administration von Desktop-Umgebungen
 - Dies ermöglicht u. A. zentrale Backups aller Desktops
- + Nutzer können über verschiedene Endgeräte auf ihren virtuellen Desktop zugreifen
- + Thin- und Zero-Clients sind in der Anschaffung und im Verbrauch sehr viel günstiger als Thick-Clients
- Erfordert meist eine konstante Netzwerkverbindung
- Hohe Anforderungen an die Leistung des Servers, der Speicherkapazität und Netzwerkbandbreite





Designing Education
Connecting People

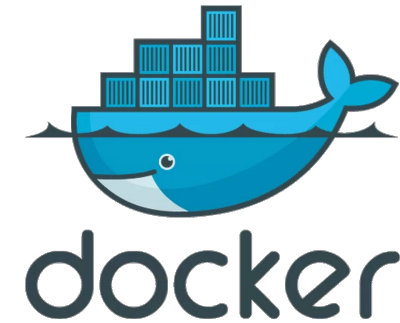
Auch Betriebssystem-Virtualisierung genannt



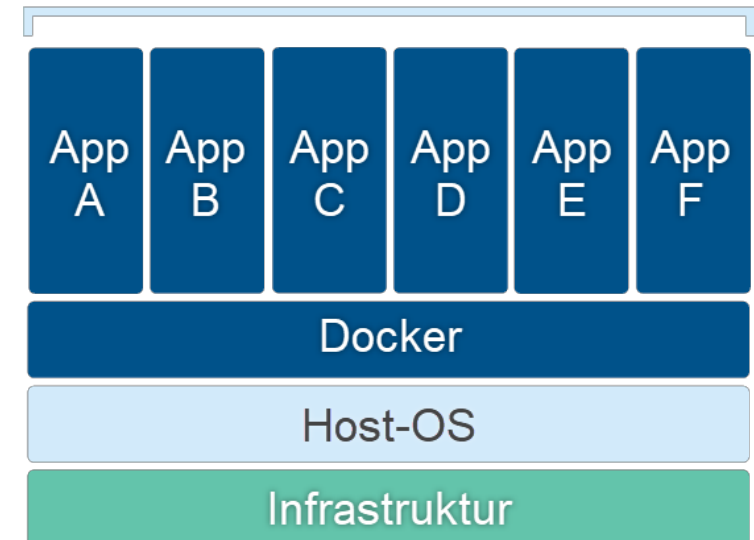
10b.2.5 Container – Virtualisierung

10.2.4 Container – Virtualisierung

- Die Virtualisierungsebene läuft als eine Anwendung innerhalb des Betriebssystems
 - Im Gegensatz zur Virtualisierung per Hypervisor bekommen die Gäste kein vollständig installiertes Betriebssystem zur Verfügung gestellt
- Die Container teilen sich den Kernel des OS
 - Das OS stellt isolierte Instanzen von sich selbst zur Verfügung
 - Jeder Container enthält die **komplette Laufzeitumgebung** (Runtime Environment) der enthaltenen Anwendung
 - Dadurch werden die Container sehr portabel, da sie zusätzlich nur noch das Host-OS benötigen
 - Sie können einfach erstellt und verteilt werden



Containerized Applications

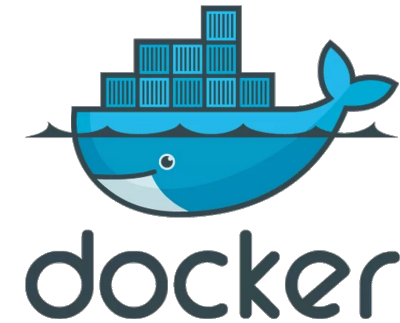


10.2.4 Container – Virtualisierung

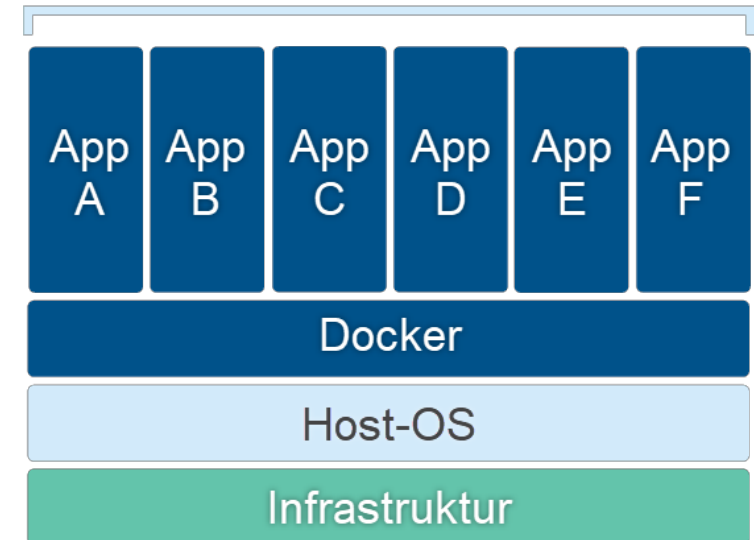
Merkmale

Geringer Installationsaufwand

- Container werden aus Images gestartet
 - Sie enthalten ein Serverprogramm und alle benötigten Komponenten wie Bibliotheken, Hilfsprogramme und Konfigurationsdaten
 - Dadurch lassen sich Unterschiede von verschiedenen Distributionen ausgleichen
- Das Installieren geschieht über das Eintippen einer Kommandozeile



Containerized Applications



10.2.4 Container – Virtualisierung

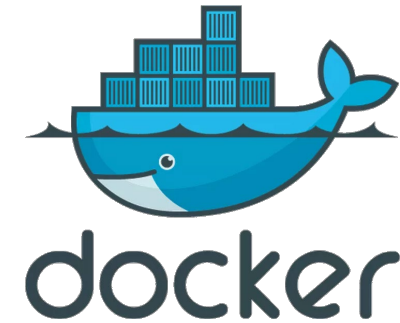
Merkmale

Plattformunabhängig

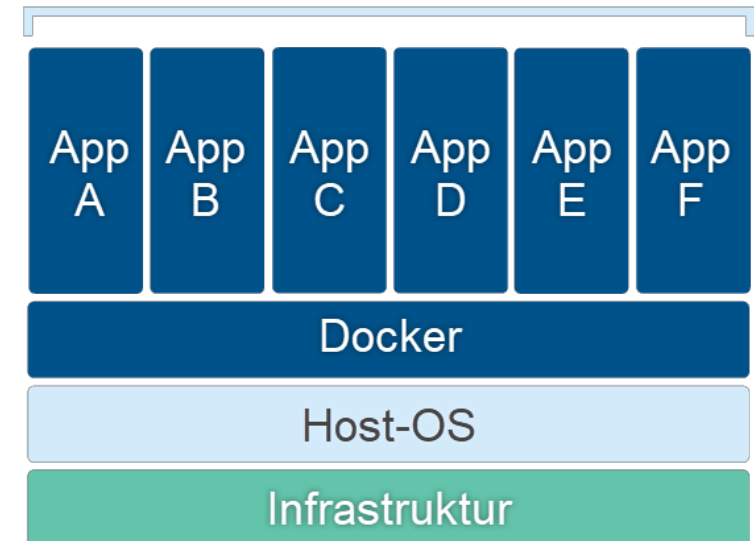
- Die Images lassen sich bequem von einem auf ein anderes System übertragen
- Um einen Container zu starten, benötigt man lediglich ein OS mit der passenden Container-Plattform

Isolierte Anwendungen

- Jede Anwendung in einem Container läuft isoliert von anderen Software-Containern auf dem gleichen OS
- Dadurch lassen sich Anwendungen mit widersprüchlichen Anforderungen parallel auf demselben System betreiben



Containerized Applications

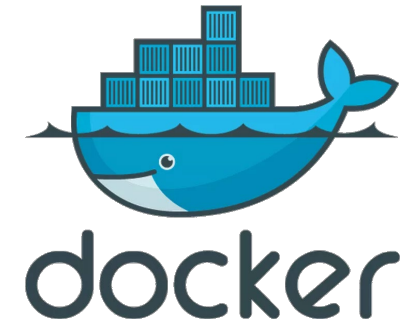


10.2.4 Container – Virtualisierung

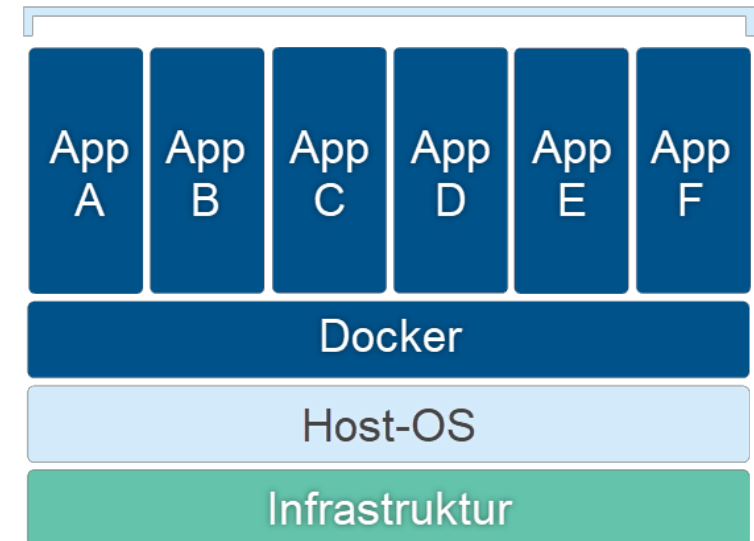
Merkmale

Minimaler Leistungsverlust durch die Virtualisierung

- Nutzt man Linux und Docker, benötigt man nur sehr wenige Daten
 - Minimalinstallationen von 100 MB möglich
- Zusätzlich werden keine Ressourcen für die Virtualisierung der Hardware verbraucht



Containerized Applications

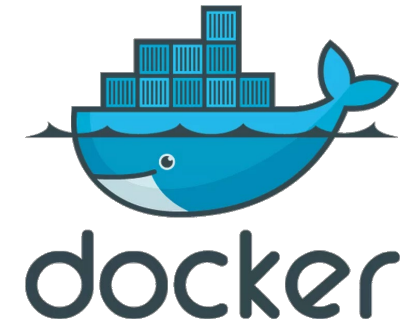


10.2.4 Container – Virtualisierung

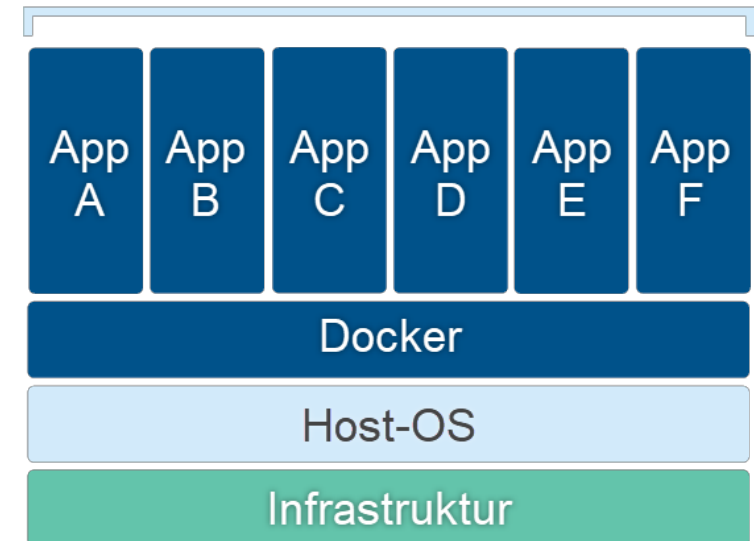
Merkmale

Einheitliche Administration und Automatisierung

- Die Verwaltung aller Container innerhalb einer Container-Plattform erfolgt mit denselben Werkzeugen
- Dadurch wird das Automatisieren sehr vereinfacht
 - So kann man z. B. je nach Bedingung dafür sorgen, dass eine neue Instanz eines Servers geöffnet oder geschlossen wird
- Orchestrierung von Containern lässt sich z. B. per Kubernetes umsetzen



Containerized Applications

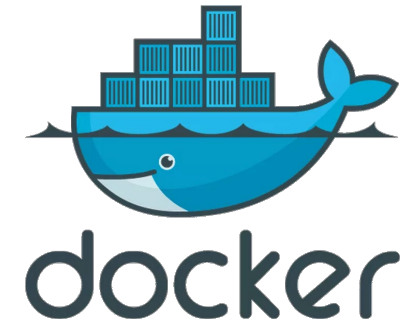


10.2.4 Container – Virtualisierung

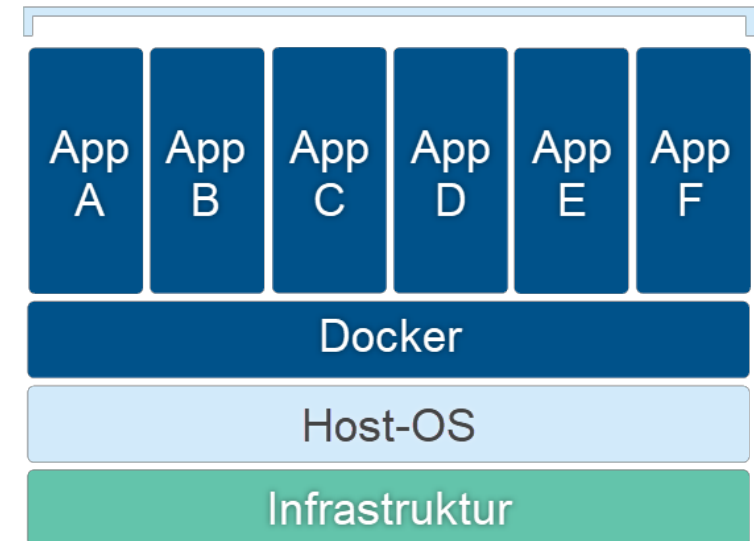
Sicherheit

Reduzierter Sicherheitslevel

- Die vielen Vorteile der Container-Virtualisierung gehen mit einer eingeschränkten Sicherheit einher
- Container sind nicht im gleichen Maße voneinander getrennt wie virtuelle Maschinen
 - Sicherheitslücken im System wirken sich nicht nur auf das System, sondern unter Umständen auf alle Container aus
- Aus diesem Grund werden Container andere Arten der Virtualisierung vorerst nicht vollkommen ersetzen



Containerized Applications



**Auf in das nächste
Modul!**