

# Virtualisierung und Cloud Computing

## Cloud-Computing – woher kommt der Begriff?

Informatiker und Computersystemarchitekten zeichnen Netzwerkverbindungen und Komponenten, die variabel sind oder nicht von vorne herein feststehen, in Diagrammen gerne als Wolke. Dies gilt insbesondere für Verbindungen über das Internet. So kam Cloud-Computing zu seinem Namen. Nicht mehr Computer im Rechenraum oder dem Firmenrechenzentrum erledigen die Arbeit, sondern Server „in der Cloud“. Mit anderen Worten: *Es gibt keine Cloud; es handelt sich nur um die Computer, die einem nicht gehören.*

## Auslagerung von Rechnerkapazitäten

Rechen- und Speicheraufgaben werden auf Computer verlagert, die sich nicht im eigenen Rechnerraum befinden, sondern bei einem Anbieter, der dies als Leistung zur Verfügung stellt. Damit dies möglich ist, muss eine Verbindung zu dessen Servern hergestellt werden. Die Verbindung wird über das Internet hergestellt.

## Virtualisierung als Treiber von Cloud-Computing

Die Erfolgsgeschichte des Cloud-Computing ist mit der der Rechnervirtualisierung eng verknüpft: Die Leistung eines großen Computers kann in kleinere Einheiten aufgeteilt werden. Jede dieser Einheiten ist von der anderen durch ein Rechteckmanagement abgegrenzt. Auch die Gesamtrechenleistung des

Systems kann so gemanagt werden, dass den einzelnen Einheiten nur jeweils ein Teil davon zur Verfügung steht.

Entsprechend wird auch mit dem Arbeitsspeicher verfahren. Jede einzelne Einheit hat nur Anspruch auf einen Teil des RAMs. Diese Aufteilung wird softwaremäßig vorgenommen, wobei die Hardware, insbesondere die Prozessorarchitektur diese Aufteilung unterstützt.

## **Virtuelle und physikalische Maschinen**

Die einzelnen Einheiten nennt man „*virtuelle Maschinen*“. Der Begriff „*Maschine*“ ist eins zu eins dem englischen Sprachgebrauch entnommen. Anwender bezeichnen ihren Computer oder Server als Maschine. Um den eigentlichen Server zu benennen, verwendet man oft den Begriff „*physikalische Maschine*“.

Die virtuellen Maschinen arbeiten unabhängig voneinander mit unterschiedlicher Konfiguration und gegebenenfalls auch mit unterschiedlichem Betriebssystem und unterschiedlicher Softwareausstattung. Grundsätzlich kann man auf einer physikalischen Maschine unter Umständen in den virtuellen Untereinheiten sehr unterschiedliche Betriebssysteme laufen lassen, solange diese auf einer gemeinsamen Rechnerarchitektur aufsetzen, zum Beispiel auf der von Intel oder AMD.

## **Emulation**

Man kann eine Virtualisierung soweit treiben, dass eine bestimmte Rechnerarchitektur komplett *simuliert* beziehungsweise *emuliert* wird. Solche Emulatoren waren zu Zeiten von Commodore 64 oder Atari ST schon verbreitet, um Software auf einem Rechner zu betreiben, für den die Programme eigentlich nicht vorgesehen waren.

Emulatoren sind Software. Der Computer muss also mit Hilfe dieser Software eine Art Übersetzungsleistung vollbringen, um

ein bestimmtes Hardware-Modell zu simulieren. Dabei müssen Befehle, Ein- und Ausgabeanforderungen sowie möglicherweise Hardware-Eigenschaften übersetzt werden, um den Programmen, die in dieser Emulation laufen, eine entsprechende Umgebung vorzugaukeln. Es liegt in der Natur der Sache, dass entsprechende *Wirtscomputer* sehr leistungsfähig sein müssen, da diese ja auch die Emulation realisieren müssen. Mit jeder weiteren virtuellen Maschine, die zur Emulation verwendet wird, steigt der Aufwand entsprechend.

## **Teilvirtualisierung**

In der Praxis genügt es oft, nur einen Teil des Systems zu virtualisieren, zum Beispiel, wenn sich das Betriebssystem in den einzelnen virtuellen Maschinen gar nicht so sehr unterscheidet. Verschiedene Linuxdistributionen nutzen zum Beispiel oft den gleichen Linux-Kern. So muss nicht die komplette Umgebung virtualisiert werden, sondern nur ein Teil. Diese virtuellen Maschinen teilen sich also den Linuxkern, je nach Ausführung auch die Treiber, die das System mit anderen Einheiten verbindet.

## **Vollvirtualisierung**

Meist möchte man aber in der Lage sein, auf den virtuellen Maschinen durchaus unterschiedliche Betriebssysteme zu verwenden, zum Beispiel diverse Linuxdistributionen und verschiedene Versionen von Windows. Gemeinsam ist diesen Betriebssystemen dann, dass sie mit einem bestimmten Prozessortyp, nämlich mit dem des Wirtssystems harmonieren.

Eine zentrale Aufgabe bei der Virtualisierung von Computersystemen ist, die einzelnen virtuellen Maschinen voneinander strikt zu isolieren. Eine weitere zentrale Aufgabe ist, die Aufteilung der Ressourcen Prozessorzeit/Rechenzeit, Arbeitsspeicher, Festplattenspeicher auf die einzelnen virtuellen Maschinen. Dabei können unterschiedliche Konfigurationen verwendet werden, also unterschiedlich viel

Rechenkapazität, RAM und Plattenplatz zugeteilt werden. Dies geschieht durchentsprechende Software. Der entsprechende Begriff für diese Art von Software ist *Hypervisor*.

## **Vorteile virtueller Systeme**

Welchen Vorteil bieten nun virtuelle Server? Auch wenn in der Computer- und Informationstechnik ein Preisverfall zu beobachten ist, erfordern Rechner, Rechnerräume und Rechenzentren bedeutsame Investitionen.

### **Uneinheitliche, nicht vorhersehbare Lastverteilung**

Gerade im Internet-Kontext wurde deutlich, dass zum Beispiel die meisten Websites selten bis sehr selten abgerufen werden, einige wenige jedoch sehr häufig. Leider lässt sich dies kaum vorhersagen und schwankt über verschieden kurze oder lange Zeiträume stark. Im Ergebnis wäre es eine große Verschwendung, für jede Website einen eigenen, relativ teuren physikalischen Server zu verwenden.

So war der erste Schritt, um kosteneffiziente Lösungen und Produkte zum Betrieb von Websites bereit zu stellen, die Aufteilung eines großen Servers in viele kleine Einheiten, *virtuelle Hosts* genannt. Jedem Nutzer beziehungsweise Kunden wurde ein Teil des Plattenplatzes zugeteilt, der jeweils von den anderen Teilen isoliert war, jeder Kunde also nur auf den von ihm gemieteten Teil Zugriff hatte. So konnten sich viele Websites einen physikalischen Rechner teilen. Das ist das Geschäftsmodell, das man *Shared Hosting* nennt.

Der Nachteil dabei ist, dass zwar der Plattenplatz zuteilbar ist, die *Prozessorkapazität* und der *Arbeitsspeicher* allerdings nicht. Deswegen mussten in der Praxis zum Beispiel Skriptlaufzeiten und die Verfügbarkeit von Arbeitsspeicher global limitiert werden, also von Haus aus, gültig für alle

Kunden, deren Sites auf diesem physikalischen Server untergebracht sind.

Der Grund dafür ist, dass das System die Anforderungen im Prinzip nacheinander abarbeitet. Um eine Blockade der Ressourcen durch einzelne Websites zu erschweren, müssen Vorkehrungen getroffen werden, was die technischen Möglichkeiten solcher Lösungen und Produkte am Ende limitiert.

Websites, die häufig abgerufen werden, also viel Last erzeugen, werden technisch zunächst genauso behandelt wie alle anderen auch auf diesem System. Dies ist auf Dauer unbefriedigend. Für diejenigen Kunden, die entsprechend viel Last erzeugen, war der Umstieg auf einen eigenen Hardwareserver, man sagt *dedizierten Server*, dann das Mittel der Wahl, um sich den Begrenzungen virtueller Hosts auf einem *Shared Server* zu entziehen. Allerdings springen dann auch die Betriebskosten nach oben.

Entsprechend war ein Trend zu beobachten, nach dem Rechenzentrenbetreiber, die dedizierte Server anboten, auf kleinere, stromsparende Einheiten setzten, die preissensitivere Kunden mieten konnten. Große Hardwareserver mit üppiger Ausstattung schienen auf einmal weniger gefragt zu sein. Der Bedarf an kostengünstigen physikalischen Maschinen wuchs, so dass die Großrechner, die man auch als Mainframes bezeichnete, als Dinosaurier galten.

Aufgrund der Weiterentwicklung der Informations- und Prozessortechnologie und einem allgemeinen Preisverfall war es möglich, vergleichsweise billige Standardrechner in virtuelle Maschinen zu unterteilen.

Neu war die Idee allerdings nicht.

## **Vorbild Mainframes**

Mainframes konnten bereits in den 1960er und 1970er Jahren so verwendet werden, dass darauf laufende Programme

„gleichzeitig“ verwendet werden konnten. 1987 schaffte man es, DOS auf Unix laufen zu lassen, 1989 konnte man Dos- und Windowsprogramme auf einem Apple Mac betreiben.

## **Kosten und Hardware-Ressourcen teilen**

Mit virtuellen Maschinen wurde das Prinzip der virtuellen Hosts aufgegriffen, damit sich viele Anwender sozusagen die Kosten des physikalischen Rechners teilen können. Der wesentliche Unterschied an diesem Punkt ist, dass Rechenkapazität und Arbeitsspeicher für jeden Kunden individuell festgelegt werden können.

In dieser Entwicklung ging der Trend dann wieder zu sehr leistungsfähigen größeren Servern, mit vielen Prozessoren und üppiger RAM-Ausstattung, um möglichst viele virtuelle Maschinen darauf betreiben zu können.

## **Neue Geschäftsmodelle**

Auf Basis der Virtualisierung von Computerhardware wurde es möglich, neue Geschäftsmodelle zu entwerfen. Das wesentliche an Computern in der Cloud ist, dass der Betreiber von Websites oder Softwareangeboten, die über das Netz zur Verfügung gestellt werden („*Software as a Service*“, kurz SaaS) keine Investitionen in Serverhardware tätigen muss und kein eigenes Rechenzentrum betreiben muss.

## **Mieten statt besitzen**

Statt dessen werden Kapazitäten angemietet. Dadurch ist es möglich, schnell auf steigenden Bedarf zu reagieren. Solche Lastspitzen können saisonal bedingt entstehen, etwa durch das Weihnachtsgeschäft im Onlinehandel oder durch andere Umstände.

## **Einfache Skalierbarkeit**

Umgekehrt kann man allerdings auch gemietete Kapazitäten wieder kündigen, wenn man sie nicht mehr braucht. In der

Praxis sieht das dann so aus, dass man zu zum Beispiel einem virtuellen Server weitere Server hinzufügen kann und diese über einen Loadbalancer, der die Last gleichmäßig verteilt, zuweisen kann. Aufgrund weiterentwickelter Replikationstechnologien ist es heute kein Problem mehr, dafür zu sorgen, dass auf jedem dieser Server die gleiche Information vorhanden ist. Dies ist auch mit Datenbanken möglich.

Statt ein eigenes Rechenzentrum zu betreiben, gehen Anwender dazu über, Kapazitäten von einem größeren Rechenzentrumsanbieter zu mieten, der seinerseits allerdings entsprechend große Hardwareressourcen bereit halten muss.

## **Unabhängigkeit von der Hardware**

Wie die einzelnen Kapazitäten in der Praxis realisiert werden, braucht den Mieter nicht zu interessieren. Entscheidender ist, welche Kosten für die Miete entstehen und welches Service-Niveau damit zu erreichen ist. Schließlich ist der Aufbau eines virtuellen Systems komplexer: Es gibt mehr Fehlerursachen, denn es ist zusätzliche Software nötig. Diese kann ausfallen, mit anderen Elementen unverträglich sein, sie kann angegriffen werden und veralten.

## **Virtuelle Maschinen kopieren und verschieben**

Auf der Seite, auf der die Vorteile stehen, befinden sich Flexibilität und Agilität. Anders als ein Hardwaresystem kann man eine virtuelle Maschine kopieren und verschieben, um sie in anderen Rechenzentren einzusetzen oder auf anderer Hardware zu betreiben. Das ermöglicht zum Beispiel schnelles Reagieren bei Hardwareausfall oder sprunghaftem Wachstum. Aktuell findet im Unternehmensumfeld ein Umstieg von eigenen Rechnerressourcen auf Ressourcen „in der Cloud“ statt. Firmen tendieren dazu, sich Kapazitäten zu mieten statt diese selbst mit eigenen Servern zu produzieren. Treiber dieser Veränderung sind nicht nur Kostenvorteile, sondern auch steigender Bedarf an der Verarbeitung großer Datenmengen zu Analysezwecken

(Stichwort *Big Data*), Verarbeitung von Sensordaten (Stichwort *Internet of Things*) und der Notwendigkeit, mit dem eigenen Softwareprodukt schnell weltweit verfügbar zu sein (Stichwort *Globalisierung*).