



H2 – Virtualisatie & Cloud

Inleiding

**HO
GENT**

2. Virtualisatie & Cloud

- 2.1 Wat is virtualisatie?
- 2.2 Concepten van virtualisatie
- 2.3 Multi-Tenancy
- 2.4 Cloud Computing

**HO
GENT**

2.1 Wat is virtualisatie?

**HO
GENT**

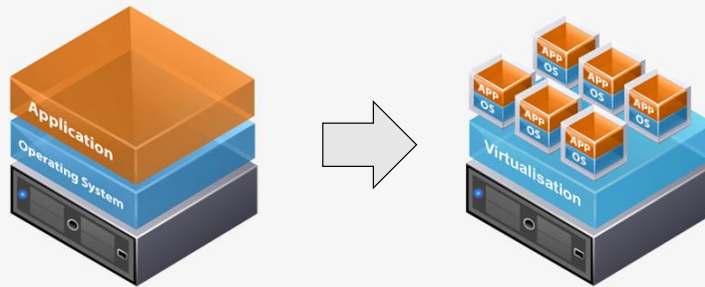
In computing, virtualization refers to the act of **creating a **virtual** (rather than actual) **version** of **something**, including virtual computer hardware platforms, storage devices, and computer network resources.**

<https://en.wikipedia.org/wiki/Virtualization>



Binnen de wereld van IT verwijst virtualisatie naar het creëren van een virtuele (in plaats van een daadwerkelijke) versie van iets, zoals virtuele computer hardware, virtuele opslagapparaten of virtuele netwerkbronnen.

Virtuele Architectuur



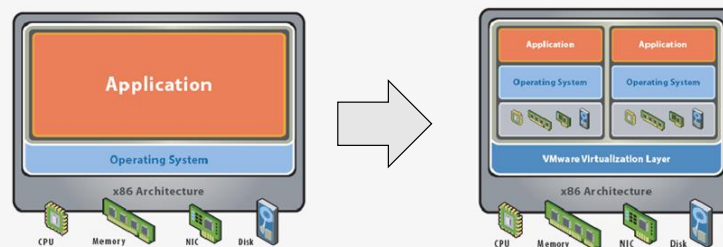
**HO
GENT**

Traditionele vs. virtuele architectuur

Traditioneel gebruiken we op een computer één besturingssysteem, zoals Microsoft Windows. Het besturingssysteem spreekt de hardware aan, en één of meerdere applicaties draaien bovenop het besturingssysteem. Het is wel mogelijk om op één computer verschillende besturingssystemen naast elkaar te installeren, maar deze kunnen niet tegelijk opgestart worden. Om te wisselen moet je de computer dus opnieuw opstarten. Door de opkomst van virtualisatie is het echter wel mogelijk om meerdere besturingssystemen gelijktijdig te gebruiken.

Met behulp van virtualisatiesoftware is het mogelijk om op één computer meerdere computers te laten draaien. Dit zijn geen fysieke computers, maar virtuele computers. Deze virtuele computers werken net zoals afzonderlijke computers, maar delen dus onderliggend de fysieke hardware.

Virtuele Hardware



**HO
GENT**

Virtuele Hardware

Een traditionele computer bestaat uit verschillende hardware bronnen, zoals de processor, het geheugen, de netwerkkaart en harde schijven. Met behulp van virtualisatiesoftware gaan we een virtuele versie maken van deze bronnen. Elke virtuele machine heeft onder andere een virtuele processor (vCPU), virtueel geheugen (vRAM), één of meerdere virtuele netwerkkaarten en één of meer virtuele harde schijven (virtual disks).

De virtualisatiesoftware zorgt er voor dat de fysieke hardware gedeeld wordt over de virtuele hardware. Zo worden virtuele harde schijven bewaard als bestanden op de fysieke harde schijf, zal elke virtuele machine een stukje van het fysieke geheugen reserveren, en zullen virtuele processoren met behulp van scheduling (zie later) elk een deel krijgen van de beschikbare processorcapaciteit.

Voordelen van virtualisatie



Efficiënter gebruik van de beschikbare hardware



Goedkoper dan aparte fysieke systemen



Lagere ecologische voetafdruk

**HO
GENT**

Voordelen van virtualisatie

Met behulp van virtualisatie kunnen we dus meerdere virtuele computers op één toestel laten draaien, die de onderliggende fysieke hardware delen. Het belangrijkste voordeel van virtualisatie is dat we hierdoor de hardware van een machine optimaal kunnen benutten. Wanneer we op een krachtig computersysteem, zoals bv. een server in een datacenter, maar één besturingssysteem en een beperkt aantal applicaties zouden draaien, zal vaak een groot deel van de beschikbare capaciteit niet gebruikt worden. Door virtualisatie toe te voegen op een dergelijke krachtige machine kunnen we dus de beschikbare hardware bronnen veel efficiënter gaan gebruiken, omdat we verschillende besturingssystemen naast elkaar kunnen draaien, elk met hun eigen applicaties.

Voor virtualisatie worden vaak vrij dure en krachtige computersystemen gebruikt, die wel een grotere investering zijn dan 'normale' computers. Doordat er echter verschillende virtuele computers naast elkaar kunnen draaien zijn er veel minder fysieke computers nodig, waardoor de meerprijs snel terugverdiend is. Daardoor is het gebruik van virtuele computers dus uiteindelijk veel goedkoper dan het gebruiken van aparte fysieke computers. Een ander voordeel van virtualisatie is de mogelijke

stroombesparing. Een krachtig computersysteem zal relatief wel iets meer stroom verbruiken dan een normale computer, maar toch verbruikt zo'n systeem veel minder stroom dan wanneer je zou werken met verschillende afzonderlijke machines. Dit is niet alleen goed voor de kostprijs, maar ook voor het milieu.

Om aan de slag te gaan met virtualisatie heb je trouwens niet noodzakelijk een super krachtige computer nodig! In datacenters worden weliswaar vaak heel krachtige servers gebruikt, die meer dan 10 virtuele machines gelijktijdig kunnen laten draaien, maar op een moderne laptop of desktop computer kan je tegenwoordig ook reeds 2 tot 5 virtuele computers gelijktijdig opstarten.

Virtualisatiesoftware



Virtualisatiesoftware

Virtualisatie bestaat in vele maten en vormen, en er is tegenwoordig een groot gamma aan virtualisatiesoftware. Eén van de bekendste spelers op de markt is VMware, de eerste fabrikant was die virtualisatie op het Intel-platform mogelijk maakte. Ook Microsoft heeft zijn eigen set aan virtualisatiesoftware, bekend onder de naam Microsoft Hyper-V. Xen, KVM en Oracle VirtualBox zijn gratis alternatieven om virtuele machines te laten draaien. Parallels Desktop is geoptimaliseerd voor Mac, en wordt vooral gebruikt om Windows te draaien bovenop MacOS.

Docker is vrij nieuw, en is ook een speciale vorm van virtualisatie. Bij docker worden er geen virtuele machines gebruikt, maar wordt een Linux kernel gedeeld door meerdere containers.

2.2 Concepten van virtualisatie

**HO
GENT**

Virtuele Machine

Een virtuele machine (VM) is een computerprogramma dat een **volledige** computer nabootst, waar andere (besturings-)programma's op kunnen worden uitgevoerd.



**HO
GENT**

Virtuele Machine

Binnen de wereld van virtualisatie wordt vaak gebruik gemaakt van virtuele machines (VMs). Een virtuele machine is een computerprogramma dat een volledige computer nabootst, en je kan binnen een virtuele machine een besturingssysteem en programma's installeren, zoals je dit kan bij een fysieke machine. Je kan binnen een virtuele machine ook verschillende virtuele hardwarebronnen configureren, waaronder:

- Virtueel geheugen (vRAM)
- Eén of meerdere virtuele processoren (vCPU)
- Eén of meerder virtuele harde schijven (virtual disk)

Virtuele machines worden beheerd door een hypervisor (zie verder).

Soorten Virtuele Machines

- **Programmeertaal-specifiek**

vb. Java Virtual Machine (JVM)



- **Emulator**

vb. VirtualBox, VMWare, ...



- **Applicatie-specifiek**

vb. Docker



**HO
GENT**

Soorten Virtuele Machines

Virtuele Machines bestaan echter in verschillende maten en vormen. De bekendste vorm is wellicht de emulator, dit is een virtuele machine die een volledige computer nabootst.

- **Programmeertaal-specifiek:** Een virtuele machine voor een programmeertaal biedt een *abstractielaag* voor de werkelijke computer: een verzameling basisfuncties waar programma's in de programmeertaal gebruik van moeten maken om de functies van de computer aan te spreken. Dit is precies wat een besturingssysteem doet, en een dergelijke machine fungeert dus als besturingssysteem voor in die taal geschreven programma's; de reden dat van een virtuele machine wordt gesproken is dat de virtuele machine zelf vaak een programma is dat wordt uitgevoerd op een bestaand besturingssysteem. De virtuele machine is de verbindende laag tussen de uitgevoerde code en de computerhardware (zoals de microprocessor) waarop het uiteindelijke programma wordt uitgevoerd, eventueel via een ander besturingssysteem. De op een virtuele machine uitgevoerde code spreekt niet direct de 'echte' hardware aan, maar gebruikt alleen de functies die worden aangeboden door de virtuele machine.

Wanneer voor een bepaald platform (hardware en eventueel besturingssysteem) een virtuele machine gemaakt is, kan elk programma dat voor deze virtuele machine geschreven is, worden uitgevoerd. De implementatie van een virtuele machine kan in elke andere programmeertaal gebeuren. Op deze manier wordt platform-onafhankelijkheid bereikt: programma's kunnen worden uitgevoerd op elk systeem waarvoor de virtuele machine is geïmplementeerd.

Overdraagbaarheid wordt een kwestie van het implementeren van een virtuele machine, in plaats van een probleem dat voor elk programma afzonderlijk moet worden opgelost. Een voorbeeld van een programmeertaal-specifieke virtuele machine is de **Java Virtual Machine (JVM)**. Het resultaat van de compilatie van een Javaprogramma wordt bytecode genoemd. Een JVM kan dan deze bytecode-instructies één voor één interpreteren door ze te vertalen naar één of meerdere echte machine-instructies. Ook zijn er JVM's die de bytecode *just in time (JIT)* compileren naar de instructies van het platform. Dit levert uiteraard prestatiewinst op in vergelijking met interpretatie. Het idee van bytecode is niet recent. Sommige varianten van de programmeertaal Pascal hadden dit concept reeds; hier heette dit p-code of pseudocode. Ook veel Prolog implementaties zijn op een virtuele machine gebaseerd: de *Warren Abstract Machine*.

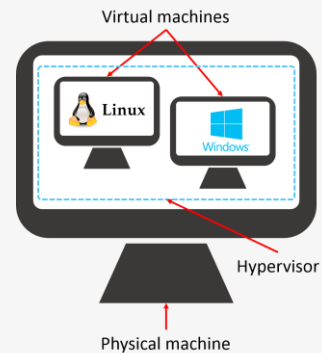
- **Emulatie:** Een *virtuele machine* die de hardware van de computer emuleert, is niet gericht op een programmeertaal (zoals Java of Pascal), maar emuleert een fysieke computer (dat wil zeggen de processor en andere hardware), zodanig dat een bestaand besturingssysteem op deze emulatie kan draaien, alsof het een fysieke computer betreft. Met een dergelijke virtuele machine is het dus bijvoorbeeld mogelijk om Windows te emuleren in een GNU/Linux-omgeving, of omgekeerd.
- **Applicatie-specifiek:** deze vorm is vrij recent, en hierbij wordt er meestal geen volledige computer nagebootst. Een bekend voorbeeld van een applicatie-specifieke virtualisatietechnologie is Docker: hierbij draaien de applicaties in een zogenaamde container, en deze containers spreken rechtstreeks de kernel aan van een Linux besturingssysteem dat op de host computer draait.

Hypervisor

Een hypervisor is de software die gebruikt wordt om **virtuele machines** aan te **maken** en op te **starten**.

Een hypervisor wordt ook vaak Virtual Machine Monitor (VMM) genoemd.

Een hypervisor regelt de virtualisatie. Met een hypervisor kan één hostcomputer meerdere VM's gelijktijdig laten draaien door de (hardware) **bronnen** virtueel te **delen**, zoals het geheugen en de processor.



**HO
GENT**

Hypervisor

Virtuele machines worden beheerd door een hypervisor. Een **Hypervisor** of **Virtual Machine Monitor** is een opstelling die ertoe dient om meerdere besturingssystemen tegelijkertijd op een hostcomputer te laten draaien. De hypervisor zorgt er voor dat de fysieke hardware gedeeld wordt over de verschillende virtuele machines, en via de hypervisor kan je ook nieuwe VMs aanmaken of de configuratie van bestaande VMs wijzigen. Er bestaan 2 types van hypervisors (zie volgende slide).

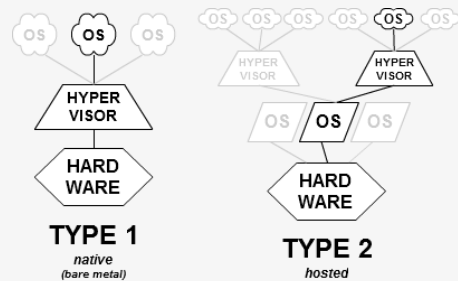
Type 1 vs. Type 2 Hypervisors

Type 1 hypervisor

- Native (bare metal) hypervisor
- Rechtstreeks op de hardware, er is **geen onderliggend OS**
- Kan **rechtstreeks** de nodige hardwarebronnen aanspreken

Type 2 hypervisor

- **Programma** bovenop een OS
- Aanspreken van hardware gebeurt via functies van het **besturingssysteem**



**HO
GENT**

Type 1 vs. Type 2 hypervisors

Een Type 1 Hypervisor ligt direct op de hardware, dat wil zeggen dat er geen besturingssysteem tussen ligt, hierdoor kunnen er meer resources aan de virtuele machines gegeven worden. Eén van de nadelen van een Type 1 hypervisor is dat je enige technische kennis moet hebben om er mee te kunnen werken. Type 1 hypervisors worden dan vooral ook gebruikt voor en door servers. Enkele voorbeelden: VMWare ESXi, Citrix Xen, KVM & Microsoft Hyper-V.

Een Type 2 Hypervisor ligt in tegenstelling tot een Type 1 niet direct tegen de hardware aan, dit wil zeggen dat er wel een Besturingssysteem onder ligt. Dit kunnen er verschillende zijn zoals: Microsoft Windows, Apple MacOS en Linux. Voordelen die je hebt bij Type2 is dat hij aan de praat te krijgen is zonder al te veel technische kennis, meestal kunnen Type2 Hypervisors geïnstalleerd worden zoals een programma. Een nadeel is dat Type2 niet zo krachtig en efficiënt zijn als Type1. Enkele voorbeelden van Type2 Hypervisors zijn: Oracle VirtualBox, VMWare Workstation, Parallels Desktop.

Wanneer op een toestel een Type 1 Hypervisor geïnstalleerd is, kan je geen Type 2 er

naast installeren. Daarentegen is het wel mogelijk om meerdere Type 2 hypervisors op hetzelfde fysieke toestel te installeren.

Type 1 vs. Type 2 Hypervisor (cont.)

Type-1 hypervisor

- Geen overhead door OS
- Efficiënter maar ook complexer
- Vooral gebruikt op servers, soms ingebouwd in hardware
- Hyper-V, VMWare ESXi, ...

Type-2 hypervisor

- Eenvoudiger om te installeren (programma)
- Vooral gebruikt op persoonlijk toestel (bv. Windows VM op Mac)
- Niet zo krachtig en efficiënt als Type-1
- VirtualBox, VMWare Player, Parallels Desktop, ...

**HO
GENT**

2.3 Multi-tenancy

**HO
GENT**

Single-Tenant vs. Multi-Tenant



- Voor één gezin
- Lager gebruik van beschikbare ruimte
- Meer afscherming
- Duurder

Single-Tenant omgeving



- Meerdere gezinnen
- Hoger gebruik van beschikbare ruimte
- Minder afscherming
- Goedkoper

Multi-Tenant omgeving

**HO
GENT**

Single-Tenant vs. Multi-Tenant

Multi-tenancy (meerdere huurders) is een concept dat al veel ouder is dan computers. Wanneer we bijvoorbeeld kijken binnen de immo markt, kan je ook een onderscheid maken tussen single-tenant en multi-tenant omgevingen.

Een alleenstaande woning is een mooi voorbeeld van een single-tenant omgeving. Het gebouw (huis) is typisch gebouwd voor één afzonderlijk gezin, bevat redelijk wat bouwgrond, en biedt meer afscherming dan een appartement, maar tegen een hogere kostprijs. Een appartement is een gebouw dat gedeeld is tot meerdere (soms zelfs honderden) gezinnen, en gebruikt de beschikbare oppervlakte dus veel optimaler, doordat de gezinnen dichter bij elkaar leven. Hierdoor is er minder afscherming tussen de gezinnen, maar een appartement is dan ook vaak goedkoper dan een alleenstaande woning.

Ook binnen de IT-wereld kunnen we dit onderscheid maken. Zo wordt jouw computer bijvoorbeeld enkel door jou gebruikt, terwijl een fysieke server in een datacenter gedeeld kan worden over verschillende klanten, bijvoorbeeld in de vorm van virtuele machines. En ook software kan single-tenant of multi-tenant zijn, zoals bijvoorbeeld

Microsoft Word die je lokaal kan installeren voor privégebruik, maar waarvan er ook een gedeelde versie beschikbaar is via de web browser via Office 365 Web Apps.

Kenmerken van Multi-Tenancy

- Bronnen worden **gedeeld**, in tegenstelling tot een dedicated of isolated omgeving
- Een **tenant** (huurder) is een gebruiker of groep van gebruikers met gemeenschappelijke toegang
- Multi-tenancy kan geïmplementeerd worden in **verschillende vormen**, zowel op niveau van hardware als software
- **Virtualisatie** speelt een belangrijke rol bij multi-tenancy
- Multi-tenancy is een belangrijk kenmerk van **Cloud Computing**

**HO
GENT**

Kenmerken van Multi-Tenancy

Een fundamenteel kenmerk van multi-tenancy is dat (computer)bronnen mogelijk gedeeld worden over verschillende tenants. Dit is in tegenstelling tot een dedicated of isolated omgeving, waar elke tenant zijn eigen (fysieke) hardware bronnen heeft, zoals bv. een dedicated server in een datacenter.

Een tenant (NL: huurder) is in deze context een individuele gebruiker, of een groep van gebruikers met gemeenschappelijke toegang. Multi-tenancy kan geïmplementeerd worden in verschillende vormen, afhankelijk van wat gedeeld is. Zo kan hardware virtueel gedeeld worden, waarbij elke tenant recht heeft op een deel van de beschikbare hardware (bv. in de vorm van een virtuele machine), maar het is ook mogelijk om multi-tenancy te implementeren op niveau van de software, bijvoorbeeld door een online webapplicatie te ontwikkelen die gebruikt wordt door verschillende klanten (= de tenants).

Het spreekt voor zich dat virtualisatie een belangrijke rol speelt bij multi-tenancy. Via virtualisatie kunnen we immers fysieke hardware opdelen in virtuele hardware, en dus virtuele bronnen toekennen aan de verschillende gebruikers. Tenslotte is het zo

dat multi-tenancy een heel belangrijke rol speelt binnen Cloud Computing, zoals we verderop in dit hoofdstuk zullen zien.

Voor- en nadelen

AYES

- Efficiënter gebruik van de beschikbare bronnen, want meerdere eindgebruikers kunnen bediend worden door één toestel of instantie van de applicatie
- Daardoor leidt multi-tenancy tot lagere operationele kosten dus goedkoper voor de eindgebruiker

NOES

- Minder isolatie en verhoogde beveiligingsrisico's, in het geval van een inbreuk op de beveiliging op één enkele instantie kunnen meerdere tenants worden getroffen
- Minder prestatie-isolatie, grote tenants kunnen de prestaties van kleinere tenants negatief beïnvloeden

Een multi-tenant omgeving moet zowel een duidelijke scheiding van prestaties als gegevens garanderen. Hoewel één hardware- of software-instantie wordt gedeeld, moet deze zich ten opzichte van elke tenant als een aparte instantie gedragen.

Voor- en nadelen

Het toevoegen van multi-tenancy, op niveau van de hardware of op niveau van de software, heeft enkele belangrijke voordelen, maar brengt ook enkele mogelijke risico's met zich mee. Het belangrijkste voordeel van multi-tenancy is dat dit, net zoals virtualisatie, kan zorgen voor een efficiënter gebruik van de beschikbare bronnen. Het spreekt voor zich dat het voordeliger is om één zware fysieke server te delen onder verschillende klanten, dan voor elke klant een aparte fysieke server te installeren en configureren die mogelijks het meeste van de tijd in rust (idle) is. Daarom leidt multi-tenancy dus ook voor lagere operationele kosten (minder servers = minder stroom en minder koeling nodig) wat ook ten goede komt van de eindgebruiker, die de nodige bronnen kan huren aan een lagere prijs.

Doordat bronnen gedeeld zijn brengt multi-tenancy echter ook enkele risico's met zich mee. Het eerste risico situeert zich op vlak van de beveiliging: als een gedeelde server gecompromitteerd is, zal dit mogelijks invloed hebben op verschillende onafhankelijke klanten (de tenants die van deze server gebruik maken), wat niet het geval is bij dedicated hardware. Daarnaast kunnen de verschillende tenants elkaars prestaties negatief beïnvloeden: als een zware/grote tenant intensief gebruik maakt

van de fysieke computerbronnen kan het zijn dat kleinere tenants hiervan de impact ondervinden, en hun applicaties dus traag zijn ondanks ze amper een belasting plaatsen op de server.

Een goede multi-tenant omgeving moet dus zorgen voor een duidelijke scheiding van zowel de gegevens (= data) als de prestaties. Hoewel multi-tenancy draait rond het delen van bronnen, zou dit eigenlijk niet merkbaar mogen zijn voor de eindgebruiker: de gedeelde instantie moet zich dus gedragen als een aparte (dedicated) instantie ten opzichte van elke tenant.

2.4 Cloud Computing

**HO
GENT**

Cloud Computing is het via een netwerk – vaak het internet – op aanvraag beschikbaar stellen van hardware, software en gegevens, ongeveer zoals elektriciteit uit het lichtnet.

https://nl.wikipedia.org/wiki/Cloud_computing



Bij Cloud computing worden computerbronnen, zoals hardware, software en gegevens, op aanvraag beschikbaar gesteld via een netwerk (vaak het internet). De term is afkomstig uit de schematechnieken uit de informatica, waar een groot, decentraal netwerk (zoals het internet) met behulp van een wolk wordt aangeduid.

Kenmerken van Cloud Computing

- **Bronnen** beschikbaar **op aanvraag** (on demand), vaak zonder tussenkomst van een fysiek persoon
 - Bronnen = hardware, software en/of opslag
 - Vaak: groot aantal servers in datacenter die door iedereen gehuurd kunnen worden, bijvoorbeeld als VM's
- Vaak via een **pay-as-you-go** pricing model (kostprijs afhankelijk van effectief verbruik)
- **Elasticiteit**: mogelijkheid om automatisch aan te passen in functie van vraag

**HO
GENT**

Kenmerken van Cloud Computing

Een belangrijk kenmerk van Cloud Computing is dus dat bronnen op aanvraag beschikbaar worden gesteld aan de eindgebruikers. Wanneer eindgebruikers bronnen aanvragen, bijvoorbeeld in de vorm van een virtuele machine, worden deze ook vaak volledig automatisch geprovisioneerd, dus zonder tussenkomst van een fysiek persoon aan de kant van de aanbieder. Onder bronnen verstaan we hier vooral een combinatie van hardware, software en/of opslag. Een Cloud omgeving is dan ook vaak een collectie van een heel groot aantal servers (soms zelfs duizenden of meer), die zich bevinden in één of meerdere datacenters en waarop virtuele machines gemaakt kunnen worden voor de eindgebruikers.

Vaak kunnen deze bronnen gehuurd worden via een pay-as-you-go pricing model. Dit wil zeggen dat de eindgebruiker een huurprijs betaalt op basis van de bronnen die hij effectief gebruikt, of op basis van het aantal bronnen dat voor hem gereserveerd is.

Een ander belangrijk kenmerk van Cloud computing is elasticiteit, wat wil zeggen dat het aantal voorziene bronnen automatisch aangepast kan worden op basis van de huidige belasting (vraag) – zie verder.

Opkomst Cloud Computing

- Eerste definitie 1997 (*Ramnath K. Chellappa*)
- 2000 opkomst Software as a Service
- 2005-2007 opkomst Cloud Computing
 - Amazon, Google, IBM en universiteiten
 - Hand in hand met opkomst virtualisatie
 - VMWare, VirtualBox, Microsoft en Citrix
- Analogie met mainframes (servers) en terminals (clients)
 - Geen centraal mainframe maar een gedistribueerde omgeving

**HO
GENT**

Opkomst Cloud Computing

De term Cloud Computing bestaat ondertussen al meer dan 20 jaar, maar kende vooral een grote groei in de periode 2005-2007 toen bedrijven als Google en Amazon het voor het eerst gingen gebruiken. Tegenwoordig is Cloud Computing niet meer weg te denken uit ons dagelijks leven, en de kans is dus groot dat je reeds enkele Cloud diensten gebruikt (zoals bv. Office 365, Dropbox, OneDrive, of Google Apps).

Cloud Computing lijkt wat op het gebruik van mainframes en terminals, in die zin dat krachtige computers (servers) gebruikt worden om meerdere eindgebruikers te bedienen (de clients). In tegenstelling tot mainframes is het echter zo dat er bij Cloud Computing geen centraal mainframe meer wordt gebruikt maar een gedistribueerde serverfarm, waarbij servers vaak verspreid zijn over verschillende geografische locaties.

Cloud

- Wolk van computers
- Virtualisatie van serveromgeving
 - Eindgebruiker weet niet
 - Waar de instanties precies draaien
 - Soms ook niet op hoeveel servers
 - Eindgebruiker is
 - Geen eigenaar meer van hardware/software
 - Niet verantwoordelijk voor onderhoud
- Schaalbare, virtuele infrastructuur



**HO
GENT**

Cloud

De *Cloud* (Nederlands: wolk) staat voor een netwerk dat met al de computers die erop aangesloten zijn een soort 'wolk van computers' vormt, waarbij de eindgebruiker niet weet op hoeveel of welke computer(s) de software draait of waar die computers precies staan. De gebruiker hoeft op deze manier geen eigenaar meer te zijn van de gebruikte hard- en software en is dus ook niet verantwoordelijk voor het onderhoud. De details van de informatietechnologische infrastructuur worden aan het oog onttrokken en de gebruiker beschikt over een "eigen", in omvang en mogelijkheden schaalbare, virtuele infrastructuur. De Cloud is dus een techniek waarmee schaalbare online diensten kunnen worden aangeboden. Zonder de mogelijkheid tot schalen heeft een aangeboden online dienst geen betrekking op Cloud computing.

Deployment modellen

- **Publieke** Cloud omgeving
 - beschikbaar voor iedereen, via het internet
- **Private** Cloud omgeving
 - toegang beperkt tot één of meerdere organisaties
 - kan in privaat (self hosted) of publiek (colocated) datacenter
- **Hybride** Cloud omgeving
 - combinatie van meerdere modellen

Er bestaan ook vele afgeleide vormen

Bv. Community Cloud, Distributed Cloud, Multi-Cloud, ...

**HO
GENT**

Deployment modellen

Een Cloud omgeving kan uitgerold worden volgens verschillende deployment modellen.

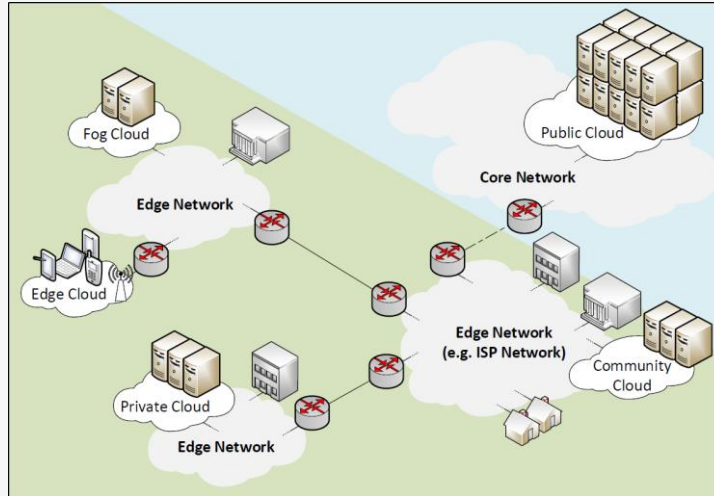
In de traditionele zin van *Cloud computing* werkt men publiek of extern. De software en data staan dan volledig op de servers van de externe dienstverlener en er wordt een generieke (voor alle afnemers gelijke) functionaliteit geleverd. De cloud omgeving is dus beschikbaar voor iedereen, en wie wil kan dus virtuele bronnen aanvragen via het internet.

Bij een private 'Cloud' werkt men op een (virtueel) private ICT-infrastructuur. In deze wolk heeft de gebruiker volledige controle over data, beveiliging en kwaliteit van de dienst. De applicaties die via de Private Cloud beschikbaar worden gemaakt, maken gebruik van gedeelde infrastructuurcomponenten die worden ingezet voor meerdere afnemers, bijvoorbeeld afdelingen van een bedrijf, maar worden zelf niet gedeeld met andere klanten. De verantwoordelijkheid voor het onderhouden van de private Cloud kan worden uitbesteed aan een leverancier van ICT-diensten. De fysieke locatie van de infrastructuurcomponenten kan zowel de Cloud leverancier als de klant zelf zijn.

Vaak worden er meerdere Cloud omgevingen met verschillende deployment modellen gecombineerd, om de voor- en nadelen van beide te combineren. Als meerdere interne en/of externe Cloud omgevingen samen worden gebruikt wordt er gesproken van een hybride Cloud. Dit kan bijvoorbeeld nodig zijn om legale redenen: als gevoelige data door een bepaalde wetgeving niet bewaard mag worden op een server in het buitenland, kan deze opgeslagen worden binnen een private Cloud omgeving, terwijl zwaar rekenwerk verricht kan worden op een publieke Cloud omgeving met een grotere pool aan computerbronnen.

In het algemeen zijn publieke Cloud omgevingen veel groter dan private Cloud omgevingen, en beschikken deze dus over veel meer computerbronnen (zie ook verder). Om een private Cloud omgeving op te zetten heb je echter niet veel nodig, en tegenwoordig kan je dit zelfs al doen op één krachtig toestel met enkele virtuele machines, bijvoorbeeld via de gratis OpenStack software.

2.4 Cloud Computing



**HO
GENT**

Service modellen (publieke cloud)

- **Infrastructure** as a Service (IaaS)
 - Infrastructuur virtueel beschikbaar voor gebruiker
 - Gebruiker heeft controle over besturingssysteem, software en (virtuele) hardware
 - Amazon EC2, ...
- **Platform** as a Service (PaaS)
 - Platform en diensten (bv. toegangsbeheer) aangeboden
 - Gebruiker heeft controle over software, maar geen controle over onderliggende hardware
 - Microsoft Azure, Google AppEngine, ...
- **Software** as a Service (SaaS)
 - Aanbieden applicaties
 - Gmail, Office 365, Salesforce, ...

Er bestaan ook vele afgeleide vormen (BPaaS, FaaS, DaaS, DBaaS, SECaaS, ...)

**HO
GENT**

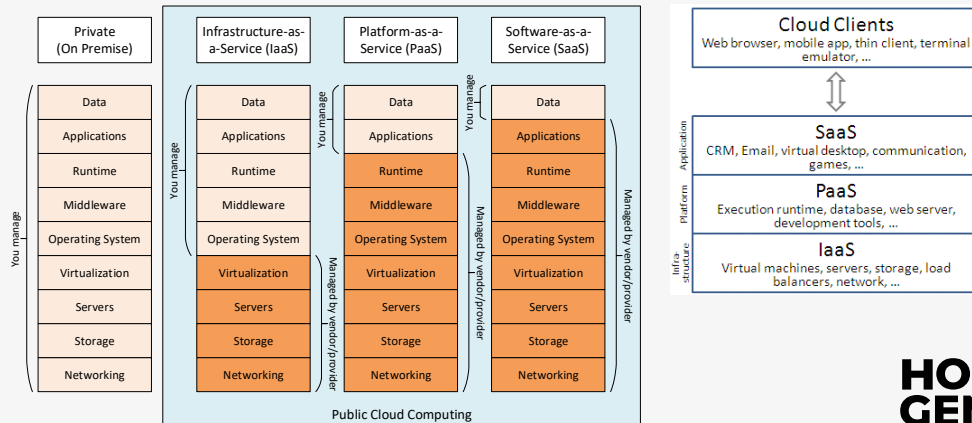
Service modellen (publieke cloud)

Binnen de Cloud worden diensten vaak aangeboden volgens één van onderstaande 3 service modellen. Er bestaan ook vele andere modellen, maar deze zijn altijd afgeleid van één van deze basismodellen.

- Bij **software as a service (SaaS)** biedt de aanbieder eindapplicaties aan "via de Cloud". Deze applicaties kunnen van allerlei soort zijn, bijvoorbeeld e-mail, klantenbeheer, personeelsbeheer, videoapplicaties enz. De dienstaanbieder heeft de volledige controle over de applicaties, maar de klant of een derde partij die het beheer uitvoert voor de klant, kan in veel gevallen wel de applicatie configureren en functioneel beheren. In veel gevallen zijn de SaaS-applicaties te gebruiken via een webbrowser op een computer.
- Bij **platform as a service (PaaS)** worden door de aanbieder een aantal diensten aangeboden bovenop de infrastructuur. Voorbeelden van diensten in deze laag zijn toegangsbeheer, identiteitenbeheer, portaalfunctionaliteiten en integratiefaciliteiten. De PaaS-laag bevat ook vaak één of meerdere frameworks zoals Python, .NET of Java, waardoor gebruikers wel beperkter zijn in mogelijkheden dan bij IaaS.

- Bij **infrastructure as a service (IaaS)** wordt infrastructuur aangeboden via virtualisatie of hardware-integratie. In deze vorm heeft de gebruiker volledige vrijheid over de hardware, en kan de gebruiker dus onder andere het besturingssysteem, de software en (virtuele) hardware kiezen. Typisch worden bij IaaS virtuele machines aangeboden aan de eindgebruiker, waarop deze extern kan inloggen (bv. via SSH).

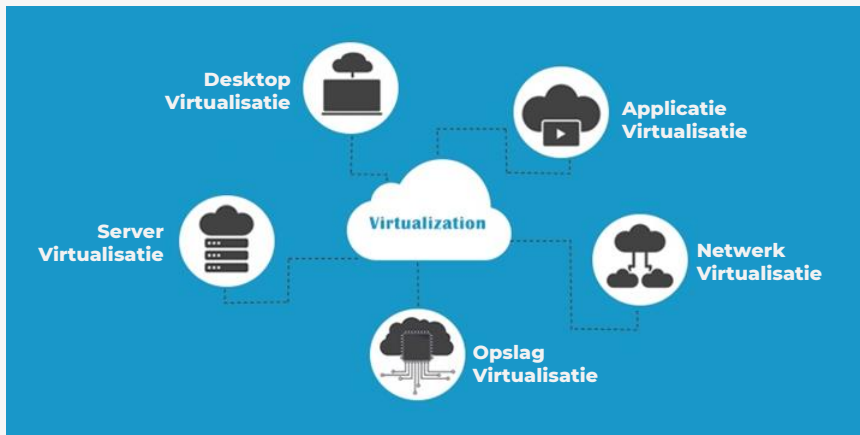
Service modellen (publieke cloud)



Service modellen (publieke cloud)

Bovenstaande figuur toont het onderscheid tussen de verschillende deployment modellen. Het belangrijkste verschil tussen de modellen is welke delen (lagen) beheerd worden door de Cloud provider, en welke beheerd worden door de gebruiker van de cloud-diensten. Bij public cloud computing heeft de eindgebruiker dus de meeste controle binnen IaaS, maar dit vereist ook de meeste technische kennis van de eindgebruiker, terwijl de controle door de eindgebruiker vrij beperkt is binnen SaaS.

Cloud Computing = virtualisatie



**HO
GENT**

Cloud computing = virtualisatie

Virtualisatie speelt een belangrijke rol binnen Cloud Computing. Bij cloud computing worden dan ook vaak virtuele bronnen beschikbaar gesteld aan de eindgebruiker, bijvoorbeeld in de vorm van virtuele machines. Zonder virtualisatie zouden er veel minder klanten gebruik kunnen maken van de Cloud omgeving, en zouden er veel bronnen verspild worden. Naast het gebruik van virtuele machines (server virtualisatie) worden er binnen Cloud omgevingen ook vele andere vormen van virtualisatie gebruikt, zoals virtualisatie van opslag, netwerk virtualisatie, virtualisatie van applicaties en zelfs virtualisatie van volledige desktopomgevingen.

Elasticiteit van de Cloud

- De mate waarin het aanbod zich afstemt op een stijgende of dalende vraag
 - Meer resources voorzien tijdens piekbelasting, minder bronnen tijdens daluren
- Automatische provisie van benodigd aantal instanties
 - Hoe voorspellen?
 - Autonoom, en dus zelflerend?
- Publieke Cloud: “oneindige” hoeveelheid bronnen
 - Elasticiteit is een belangrijk kenmerk van publieke cloud omgevingen!

**HO
GENT**

Elasticiteit van de Cloud

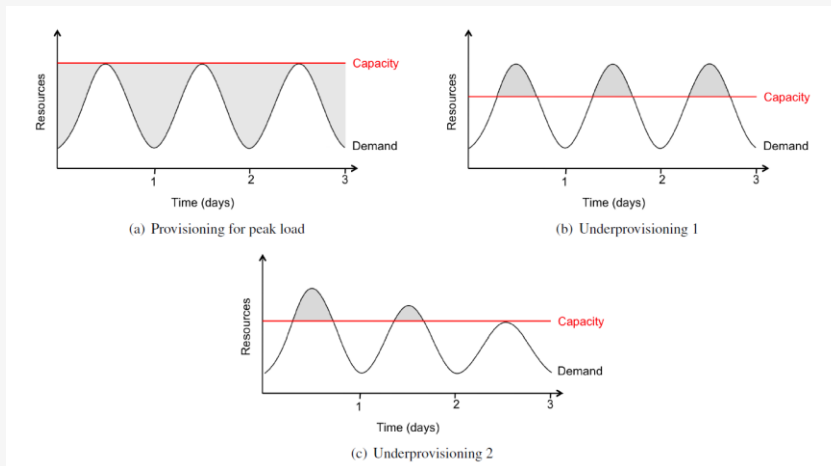
Elasticiteit is een belangrijk begrip dat Cloud computing onderscheidt van andere vormen zoals grid computing. Elasticiteit is de mate waarin het aanbod zich afstemt op een stijgende of dalende vraag. Concreet wil dit zeggen dat er meer (virtuele) bronnen ingezet worden wanneer er meer gebruikers de cloud applicatie gebruiken (piekbelasting), en er eventueel bronnen vrijgegeven kunnen worden wanneer de belasting op de applicaties daalt.

Sommige Cloud omgevingen bieden de mogelijkheid om dit proces te automatiseren. Hierbij worden er automatisch een aantal instanties voorzien, en dit aantal kan wijzigen in de tijd, afhankelijk van de huidige belasting. Een uitdaging hierbij is om het aantal benodigde instanties correct in te schatten. Dit kan zowel proactief (voorspellen op voorhand) als reactief (als antwoord op wijziging in vraag) zijn. Vaak wordt hier gebruikgemaakt van zelflerende modellen, die op basis van historische data een voorspelling doen voor de mogelijke belasting in de toekomst.

Publieke Cloud omgevingen bestaan vaak uit duizenden servers, en bieden een “oneindige” hoeveelheid aan bronnen. Het werkelijke aantal bronnen is uiteraard

eindig, maar dergelijke omgevingen bevatten vaak meer bronnen dan één gebruiker ooit nodig zal hebben, vandaag de term “oneindig”. Door deze oneindige hoeveelheid kunnen individuele applicaties dus opschalen of neerschalen naargelang de vraag, en elasticiteit is dus een belangrijk kenmerk van publieke Cloud omgevingen!

2.4 Cloud Computing



Bron: "Above the Clouds: A Berkeley View of Cloud Computing", Armbrust et al.

**HO
GENT**

Elasticiteit van de Cloud

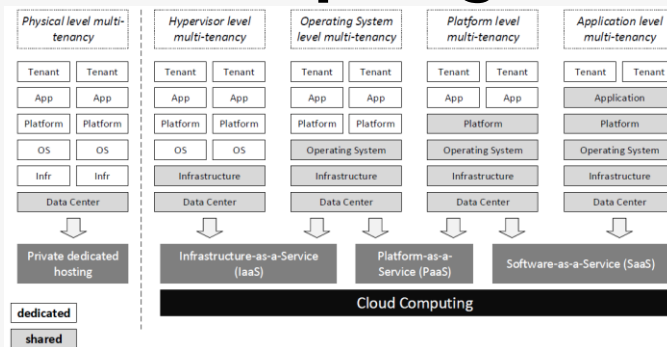
Idealiter zijn er altijd net voldoende bronnen voorzien voor de verschillende applicaties die draaien op de tijd, afhankelijk van de huidige belasting van de eindgebruikers. Soms kan het aantal bronnen echter verkeerd ingeschat zijn, en spreken we van over- of under-provisioning.

Over-provisioning wil zeggen dat er meer middelen toegewezen zijn dan nodig is. Dit dient vermeden te worden vermeden aangezien dit onnodige kosten met zich meebrengt. De applicaties gaan altijd vlot draaien, maar vaak zullen er instanties zijn die niets staan te doen. De meeste Cloud providers rekenen echter het aantal voorziene (= provisioned) bronnen aan, en dus niet het werkelijke aantal gebruikte bronnen, waardoor over-provisioning dus zorgt voor een meerkost die vermeden kan worden.

Under-provisioning wil zeggen dat er minder middelen beschikbaar zijn dan nodig om de applicaties correct te laten draaien. Under-provisioning moet vermeden worden, want dit kan er voor zorgen dat applicaties heel traag reageren of soms zelfs volledig crashen, met mogelijks verlies van klanten tot gevolg.

2.4 Cloud Computing

Cloud Computing = Multi-Tenancy



Bron: "A dynamic Tenant-Defined Storage system for efficient resource management in cloud applications", Maenhaut et al.

**HO
GENT**

Cloud Computing = Multi-Tenancy

Cloud computing is een mooi voorbeeld van multi-tenancy. Zeker bij public Cloud computing maken dus vaak meerdere eindgebruikers gebruik van dezelfde set fysieke hardware. De figuur in deze slide dient ter illustratie, maar toont mooi hoe verschillende vormen van multi-tenancy gerelateerd zijn aan de verschillende service modellen van public Cloud computing.