

# Operating Systems HoGent

JeeVeeVee

2020/2021

# Contents

<b>1</b>	<b>besturingssysteem</b>	<b>2</b>
1.1	Wat is een besturingssysteem . . . . .	2
1.2	Soorten besturingssystemen . . . . .	2
1.3	Concepten van besturingssystemen . . . . .	2
1.3.1	verschillende lagen . . . . .	2
1.3.2	programma's en taken . . . . .	3
1.3.3	processen . . . . .	3
1.3.4	resources . . . . .	3
1.3.5	Scheduling . . . . .	3
1.3.6	Concurrency . . . . .	3
<b>2</b>	<b>Virtualisatie &amp; Cloud</b>	<b>4</b>
2.1	Wat is virtualisatie? . . . . .	4
2.2	Concepten van virtualisatie . . . . .	4
2.2.1	Virtuele Machine . . . . .	4
2.2.2	Soorten VM's . . . . .	4
2.2.3	Hypervisor . . . . .	4
2.3	Multi-tenacy . . . . .	5

# Chapter 1

## besturingssysteem

### 1.1 Wat is een besturingssysteem

Een besturingssysteem is de link tussen hardware en de gebruiker. Zonder besturingssysteem zou de gebruiker rechtstreeks de hardware moeten aanspreken, wat quasi onmogelijk is. Een besturingssysteem is een programma dat het mogelijk maakt de hardware van een computer te gebruiken. Een afkorting die vaak wordt gebruikt is OS (Operating System) De taken van een OS zijn :

- opslaan en ophalen van info
- programma's afschermen
- gegevensstroom regelen
- prioriteiten regelen
- beheer en delen van bronnen
- tijdelijke samenwerking tussen programma's mogelijk maken
- reageren op fouten
- tijdsplanning maken

Voorbeelden van OS's zijn Windows, MacOS, linux distros,...

### 1.2 Soorten besturingssystemen

We maken een onderscheid tussen :

- single-tasking systemen
- multi-tasking single user systemen
- multi-user systemen

### 1.3 Concepten van besturingssystemen

#### 1.3.1 verschillende lagen

Veel OS's implementeren de interface tussen gebruiker en computer als een reeks stappen of lagen. In de toplaag zijn de functies vastgelegd en de onderste laag bevat de details van het laagste niveau om deze functies uit te voeren. De gebruiker communiceert altijd met de bovenste laag, deze laag noemen we de shell of de command interpreter. De shell geeft op zijn beurt de opdrachten door aan de laag onder hem. Uiteindelijk komt de opdracht (versnippert) aan bij de onderste laag : de kernel. De kernel is het hart van het OS.

### 1.3.2 programma's en taken

Een OS zorgt ervoor dat taken uitgevoerd worden, we maken hierbij een onderscheid tussen :

- interactieve programma's  
een programma dat de gebruiker vanaf de terminal activeert, over het algemeen is dit een korte opdracht, er wordt vaak een **snelle respons** verwacht.
- batch programma's  
programma's die geen directe respons verwachten, de opdrachten worden opgeslaan in een file, en worden toegevoegd aan een batch queue.
- real-time programma's  
Er is sprake van een tijdsbeperking, vaak nog snellere respons dan bij interactieve programma's

### 1.3.3 processen

Een proces zijn 1 of meerdere opdrachten die door een programma worden beschouwd als 1 eenheid. Een programma of taak bestaat vaak uit 1 of meerdere processen. Elk proces is dus onafhankelijk en dingt mee naar het gebruik van bronnen. Een OS houdt zich vaak voornamelijk bezig met het uitvoeren van processen, het verdeelt de bronnen en bepaald de volgorde.

### 1.3.4 resources

In eerste instantie moet een proces resources aanspreken. Het OS moet dan :

- zorgen voor voldoende geheugen voor het proces  
Een proces moet zijn instructies en gegevens kunnen opslaan. Geheugen is echter eindig, en dus moet het OS het verdelen.
- het gebruik van de CPU regelen  
Er zijn gewoonlijk meer processen dan CPU's, en dus moet het OS het gebruik ervan regelen, dit gebeurt op basis van prioriteit.
- de gegevensstroom regelen van of naar randapparatuur  
bv printers waar meerdere mensen op willen printen
- bestanden of records kunnen lokaliseren  
het lokaliseren van files en records

### 1.3.5 Scheduling

Bij multi-tasking, en in het bijzonder multi-user systemen is scheduling heel erg belangrijk. Scheduling verwijst naar de manier waarop processen een prioriteit krijgen toegewezen, vaak in combo met een prioriteiten wachtrij, dit komt later nog aan bod.

### 1.3.6 Concurrency

Processen zijn vaak niet onafhankelijk, ze zijn dan concurrent. Dit wil zeggen dat 2 processen soms ook dezelfde resources aanspreken, dit zorgt voor conflicten. Het OS moet de volgorde van de processen zodanig regelen, dat die conflicten worden vermeden, dit noemt **synchronisatie**.

## Chapter 2

# Virtualisatie & Cloud

### 2.1 Wat is virtualisatie?

Virtualisatie verwijst naar het creëren van een virtuele versie van iets, zoals virtuele computer hardware, opslagapparaten, netwerkbronnen,.... Traditioneel gebruiken we 1 OS op een computer, dat OS spreekt de hardware aan, en de applicaties draaien er bovenop. Door middel van virtualisatie kan men op 1 computer meerdere (virtuele) computers laten draaien. Die virtuele computers delen van fysieke hardware.

De voordelen van virtualisatie is dat het efficiënter gebruik maakt van de beschikbare ruimte, het is goedkoper en de ecologische voetafdruk verkleint eveneens.

Virtualisatie bestaat in veel vormen en maten. 1 van de bekendste spelers op de markt is VMware, maar ook grote bedrijven als Microsoft en Oracle mengen zich. Docker is een vrij nieuwe en speciale vorm van virtualisatie, er worden geen virtuele machines gebruikt, maar er wordt een linux kernel gebruikt door meerdere containers.

### 2.2 Concepten van virtualisatie

#### 2.2.1 Virtuele Machine

Een virtuele machine (VM) is een computerprogramma dat een volledige pc nabootst, en waar andere programma's op kunnen worden uitgevoerd. Een VM bezit vRAM, vCPU en een virtual disk.

#### 2.2.2 Soorten VM's

- Programmeertaal-specifiek : JVM
- Emulator : VirtualBox, VMWare
- Applicatie-Specifiek : Docker

#### 2.2.3 Hypervisor

Een hypervisor is de software die gebruikt wordt om VM's aan te maken en op te starten, deze worden soms ook Virtual Machine Monitor genoemd (VMM). Deze regelt de virtualisatie, en door er van gebruik te maken, kan 1 host pc meerder VM's tegelijkertijd laten draaien.

- Type 1
  - native hypervisor
  - er is geen onderliggend OS
  - kan de hardware rechtstreeks aanspreken
  - geen overhead door OS
  - efficient maar ook complex
  - handig voor servers

- Type 2  
programma bovenop OS (VMWare)  
het aanspreken van hardware gebeurt via functies van het onderliggend OS  
eenvoudig  
vooral gebruikt op persoonlijke toestellen  
niet zo krachtig

## 2.3 Multi-tenacy

Multi-tenacy is een concept dat veel ouder is dan computers, het stamt uit de immo markt. Een single tenacy is als er maar 1 huurder in een gebouw woont, zoals in een alleenstaande woning. Een appartement is multi-tenacy omdat het meerdere gezinnen kan huisvesten. De link met IT is dat een persoonlijke computer alleen door jou wordt gebruikt, terwijl een fysieke server in een datacenter gedeeld wordt met veel klanten.

kenmerken :

- bronnen worden gedeeld
- een huurder is een gebruiker of een groep van gebruikers met gemeenschappelijke toegang
- er zijn verschillende vormen, zowel op niveau van hard als software
- virtualisatie speelt een belangrijke rol
- multitency is belangrijk kenmerk van Cloud Computing

voordelen	nadelen
Efficienter gebruik van de beschikbare bronnen, er kunnen immers meerdere eindgebruikers door 1 toestel worden bediend	Minder isolatie en verhoogde beveiligingsrisico's : 1 inbreuk op 1 instantie kan andere tenants treffen
lagere operationele kosten, en dus goedkoper voor de eindgebruiker	minder prestatie-isolatie, grote tenants kunnen de prestatie van kleinere tenants negatief beïnvloeden

## 2.4 Cloud Computing

Bij Cloud computing worden computerbronnen, zoals hardware, software en gegevens, op aanvraag beschikbaar gesteld via een netwerk (meestal internet). De term is afkomstig uit de schematechnieken uit de informatica.

kenmerken :

- bronnen beschikbaar op aanvraag, vaak zonder tussenkomst van een fysieke persoon
- vaak via een pay-as-you-go pricing model
- elasticiteit : mogelijk om automatisch aan te passen in functie van de vraag.

### 2.4.1 Cloud

De cloud is een wolk van computers, een virtualisatie van de serveromgeving. De eindgebruiker weet niet waar de instanties precies draaien, en weet ook niet op hoeveel servers. Daardoor is de eindgebruiker geen eigenaar van hard/software, en is hij niet verantwoordelijk voor onderhoud. De cloud is een virtuele en schaalbare infrastructuur.

### 2.4.2 Deployment modellen

- **Publieke** Cloud omgeving : beschikbaar voor iedereen, via het internet
- **Private** Cloud omgeving : beperkte toegang tot 1 of meerdere organisaties, kan zelf in privaat of publiek datacenter.
- **Hybride** Cloud omgeving : combinatie van meerdere modellen.