

Operating Systems HoGent

JeeVeeVee

2020/2021

Contents

1	besturingssysteem	2
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
2	Virtualisatie & Cloud	4

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