

# Theorie Informatica



Inhoud:

- 1. Systeem**
- 2. Programmeren**
- 3. Informatiesystemen**
- 4. Internet**
- 5. Besturingssystemen**
- 6. Theorie bij praktijk**

# 1. Systeem

## **CPU = Central Processor Unit**

Het voert alle processen in de computer uit en bestaat, onder meer, uit een CU en een ALU. De CU (Control Unit) leest de programma instructies, decodeert deze en zet deze om in signalen die de verschillende onderdelen van de computer in gang zetten. De ALU (Arithmetic Logic Unit) kan zowel rekenen als logische operaties verrichten. Logische operaties bestaan uit Booleaanse Logica: AND, OR, XOR en NOT.

## **Chip**

Een geïntegreerde schakeling is een elektronisch circuit, waarbij alle componenten gemonteerd zijn op een plak halfgeleider materiaal. Jack Kilby heeft de geïntegreerde schakeling uitgevonden (1958).

## **North and South bridge**

South bridge zorgt voor communicatie tussen CPU en relatief langzame delen van de computer. Harde schijf, toetsenbord, USB. North bridge regelt communicatie tussen de CPU en de snellere onderdelen binnen de computer. RAM geheugen.

## **Von Neumann model**

Computer werkt volgens dit zich herhalend proces. Code wordt opgehaald uit het werkgeheugen, gedecodeerd in de CPU, uitgevoerd en weer terug gestopt in het geheugen. Ook wel fetch and execute cycle genoemd.

## **De wet van Moore**

Gordon Moore heeft in 1965 voorspeld dat het aantal transistoren in een chip elke twee jaar verdubbeld door de technologische vooruitgang.

## **Halfgeleider**

Dit is een stof die heel geschikt is voor het maken van elektronische componenten. Omdat het, qua geleiding, het midden houdt tussen een isolator en een geleider.

## Geheugen<sup>1</sup>

Geheugen is een lijst van cellen waar nummers in worden opgeslagen. De software die draait op de computer maakt van deze nummers een zinvol draaiend programma. Er zijn verschillende soorten geheugen. De belangrijkste zijn:

- 1) Registers, welke aan de CPU toebehoort. Dit zijn zeer snelle maar ook dure geheugencellen.
- 2) RAM geheugen(Random Access Memory), geheugencellen waarvan de data alleen gelezen en ingeschreven kan worden wanneer de computer aan staat.
- 3) Rom geheugen (Read Only Memory), hierin worden de start-up functies van de computer geschreven. De software hierin wordt nooit veranderd.

---

<sup>1</sup> [http://en.wikipedia.org/wiki/Computer#Control\\_unit](http://en.wikipedia.org/wiki/Computer#Control_unit)

## *Von Neumann model*

De moderne computer werkt nog steeds volgens het model dat in 1945 voor het eerst, door Von Neumann, beschreven werd: Een processor voert steeds de drie stappen van Von Neumann uit: *Haalt de volgende instructie op - decodeert de instructie - voert de instructie uit*<sup>2</sup>.

Alle data in de computer zijn gecodeerd in de vorm van **bitrijen** - rijen van eentjes en nullen. Om een programma uit te kunnen voeren, moet de processor een instructie in de vorm van een bitrij ophalen, vervolgens uit deze bitrij afleiden welke instructie de bitrij voorstelt en tenslotte de instructie uitvoeren.

Tijdens het uitvoeren van dit werk heeft de processor een plaats nodig om de invoer en eventuele tussenresultaten op te slaan. Hiervoor wordt een deel van het RAM, het zogenaamde *werkgeheugen*, gebruikt. Ook wordt, om tijd te besparen, gebruik gemaakt van het geheugen dat in de processor zit.

Dit is het registergeheugen en is direct benaderbaar door de processor, waardoor er geen vertraging optreedt door toedoen van de bus<sup>3</sup>. In het registergeheugen is echter slechts zeer beperkt plaats.

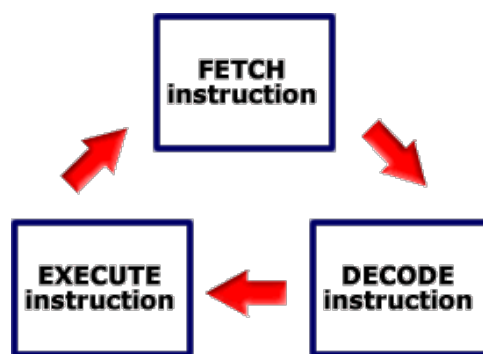


Fig.1. Het Von Neumann principe

---

<sup>2</sup> <http://www.wydea.com/topic/insidecomputer>

<sup>3</sup> Een microprocessor communiceert met het werkgeheugen via de bus. Dit is een kanaal dat de transport van digitale data voor zijn rekening neemt.

## 2. Programmeren

### Besturingssysteem

Een programma dat na het opstarten van de computer in het geheugen geladen wordt en ervoor zorgt dat andere programma's uitgevoerd kunnen worden. Een voorbeeld is Windows.

### Grafische userinterface

Met behulp van grafische beelden wordt er met de computer gecommuniceerd. De basis bestaat uit vensters, menu's, knoppen en pictogrammen. Er wordt gebruik gemaakt van een aanwijsapparaat, de muis, dat het toetsenbord aanvult. Xerox<sup>4</sup> kwam het eerst met een GUI ( Graphical User Interface)

### Command line interface<sup>5</sup>

De gebruiker spreekt de computer aan met regels tekst die een of meerdere opdrachten bevatten.

### Touch screen<sup>6</sup>

Een beeldscherm dat ook als invoerapparaat voor een computer gebruikt kan worden. Door het scherm aan te raken wordt er met de computer gecommuniceerd.

### Dennis Ritchie<sup>7</sup>

De programmeur van de taal C en Unix.

### Machinetaal

De taal die het dichtst bij de processor staat. Zij is van de eerste generatie computertalen. Een opdracht in machinetaal kan zijn: 1101 1001111 1100.

Gegevens worden opgeslagen door middel van elektrische signalen. Er zijn slechts twee toestanden mogelijk. Aan of uit. 1= aan. 0 = uit.

Getallen kunnen slechts twee waarden aannemen. Dit zijn binaire bits. Verbasterd in BITS.

---

<sup>4</sup> [http://upload.wikimedia.org/wikipedia/en/f/f2/Xerox\\_8010\\_compound\\_document.jpg](http://upload.wikimedia.org/wikipedia/en/f/f2/Xerox_8010_compound_document.jpg)

<sup>5</sup> [http://www.voidspace.org.uk/python/movpy/images/command\\_line\\_big.jpg](http://www.voidspace.org.uk/python/movpy/images/command_line_big.jpg)

<sup>6</sup> <http://www.wydea.com/topic/touchscreens>

<sup>7</sup> [http://boingboing.net/wp-content/uploads/2011/10/dennis\\_ritchie.jpg](http://boingboing.net/wp-content/uploads/2011/10/dennis_ritchie.jpg)

Het nadeel van deze taal is dat de mens deze moeilijk kan lezen en dat iedere CPU zijn eigen machinetaal heeft.

Om die rede zijn er 'hogere' programmeertalen gemaakt.

### **Hogere programmeertalen**

Deze talen zijn niet hardware afhankelijk, zoals dat bij machinetalen wel het geval is. De programma's worden in een algemene symbolische taal geschreven. Hiervoor is geen kennis van de CPU en zijn registers nodig. Een programma in een hogere taal moet eerst naar machinetaal worden vertaald, vóór het kan worden uitgevoerd. De vertaling naar machinetaal heet compileren en gebeurt door een vertaalprogramma dat compiler heet.

### **For loop**

Een for-loop maakt het mogelijk een bepaalde bewerking verschillende malen uit te voeren met betrekkelijk weinig code. Ewr wordt gebruik gemaakt van een teller die bijhoudt hoeveel keer een bepaalde instructie al is uitgevoerd.

```
for ( $teller = 0; teller <= 10; $teller += 1 ) {  
    print $teller;  
}
```

1. Zet de teller variable op een beginwaarde.
2. Kijkt of de voorwaarde nog steeds waar is.
3. Voert de code uit die in de loop staat.
4. Verhoogt de teller met steeds 1 met behulp van de loop.

### **While loop**

Een while-loop maakt het ook mogelijk een bepaalde bewerking verschillende malen uit te voeren met betrekkelijk weinig code, maar zij is anders als de for-loop. Bij de while-loop wordt eerst gekeken of een bepaalde conditie nog steeds geldt alvorens de instructie uitgevoerd wordt.

```
$teller = 0;  
while ( $teller <= 10 ) {  
    print $teller;  
    $teller ++;  
}
```

1. Zolang de conditie waar is, wordt de code uitgevoerd.
2. De code in de while loop wordt uitgevoerd.
3. De code begint weer bovenaan.
4. Als de conditie onwaar wordt stopt de loop.

Hieronder een while-loop in *pseudo-code*:

```
while (true) {  
    //doen een actie  
    if (Conditie)  
        break;  
    //weer iets anders  
}
```

## **Object**

Een object een een specifieke uitgave van een algemene klasse. Een klasse heeft algemene variabelen die in het object specifiek gemaakt worden. Het bakent een specifiek gedeelte van de functionaliteit binnen een programma af. Hierdoor is het voor de programmeur gemakkelijker het programma logisch te ordenen en bij elkaar te houden.

## **Methode**

Hiermee kunnen andere methodes aangeroepen worden of objecten worden aangemaakt. Het gebruik hiervan vereenvoudigd de code waardoor fouten voorkomen kunnen worden.

### **3. Informatiesystemen**

*Voorbeeld: Schoolinformatiesysteem*

Wanneer een schoolinformatiesysteem niet goed functioneert, komt de informatie niet op tijd bij de leerlingen.

De belangrijkste aspecten van een goed werkend informatiesysteem zijn: betrouwbaarheid, geoorloofdheid, en gebruiksvriendelijkheid.

Onder de betrouwbaarheid van een goed werkend systeem verstaan we dat de gegevens die verwerkt worden juist, volledig en tijdig beschikbaar zijn.

Het geoorloofd zijn zodat de informatie alleen beschikbaar is voor de mensen die het nodig hebben.

Het systeem moet dus voor de docenten en leerlingen gebruiksvriendelijk zijn zodat het makkelijk toegankelijk is voor beide partijen.

Gegevens worden verwerkt en opgeslagen in een database, zodat deze gegevens via een netwerk op een andere locatie opgeroepen kunnen worden.

Eenmaal bij de desbetreffende persoon aangekomen zijn de gegevens informatie geworden. De grens van het informatiesysteem wordt bereikt als de informatie niet specifiek nodig is, maar wel op een ander informatiesysteem te raadplegen is.

Als de docenten zichzelf kunnen inloggen op een bepaald systeem, kunnen ze zichzelf ziekmelden. Deze gegevens moeten vervolgens automatisch doorgevoerd worden op het informatiebord in de centrale hal en het secretariaat.

Ook zou het handig zijn als deze gegevens op het intranet van de school toegankelijk zouden zijn zodat de leerlingen ook thuis de ziekmeldingen van docenten kunnen volgen.



Om het systeem goed te kunnen laten functioneren is het dus van belang dat gegevens op het intranet na een ziekmelding en ook een betermelding automatisch geupdate worden.

### **Database**

Een grote hoeveelheid informatie die elektronisch is opgeslagen en die (op een makkelijke manier) doorzocht kan worden. Een voorbeeld is een database met klantgegevens, waarbij men een klant kan zoeken op naam, of op geboortedatum. De bedoeling is dat gegevens gemakkelijk aangevuld en weer opgevraagd kunnen worden.

### **Website en database**

Een website kan tegenwoordig niet meer zonder onderliggende database. Alle informatie die op een website wordt geplaatst moet ergens blijven. Deze wordt opgeslagen in databases. De ideale manier om informatie en gegevens op te slaan is in een SQL database. Deze kunnen klein blijven en toch veel informatie opslaan.

### **Relationeel model: Codd**

Gegevens die in een relationeel model zijn gestructureerd, worden in een relationele database geïmplementeerd. Dit zijn een verzameling tabellen. Tabellen worden hier "relaties" genoemd, naar analogie van het concept van een wiskundige relatie. De kracht van dit model treedt aan de dag wanneer bevestigingen van verschillende tabellen gecombineerd worden. De theorie achter het relationele model werd ontwikkeld door Ted Codd. Voor het bevestigen van de gegevens in een relationele database wordt de querytaal SQL gebruikt.

### **Tabel**

Een tabel is een matrix van kleine eenheden die netjes opgeborgen moet worden.

Zodat deze overzichtelijk gepresenteerd kan worden.

### **Query**

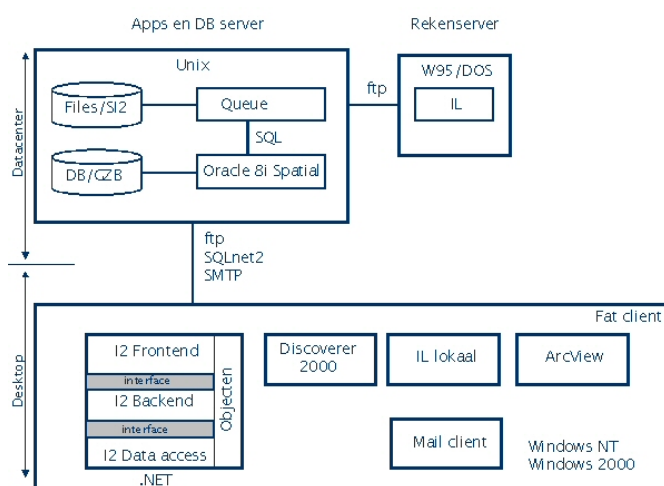
Is een vraag die aan een database gesteld kan worden en waarop de database gegevens terug zal geven.

## Systeme architectuur

Systeme architectuur is het conceptuele model dat de structuur, het gedrag, en meer weergaven van een systeem bepaalt.

De architectuur beschrijving is een formele beschrijving en representatie van een systeem. Het is zo georganiseerd dat redeneren over de structuur van het systeem mogelijk wordt. Onderdelen van het systeem kunnen los aangekoppeld worden, als de architectuur goed georganiseerd is.

De zichtbare eigenschappen van de componenten, de relaties tussen hen, en het koppelen van een bestaand systeem aan andere systemen is de bedoeling van een architectuur beschrijving. De taal voor architectuur beschrijving heet architectuur description language (ADL).



## 4. Internet

### Protocollen

Afspraken die gemaakt zijn om computers over netwerken met elkaar te laten communiceren.

### TCP/IP

Afspraken die gemaakt zijn om computers over netwerken met elkaar te laten communiceren. TCP houdt in de gaten of de pakketjes aankomen en dat dit in de juiste volgorde gebeurt. IP is

verantwoordelijk voor het routeren van het pakket naar het juiste adres.

### **Arpanet**

Het **ARPANET** (ARPA Network) was het eerste operationele packet switching computernetwerk, de voorloper van het internet, opgezet in de late jaren '60 , door het Amerikaanse ministerie van Defensie.

### **Internet Protocol**

Het is een communicatie protocol dat data-pakketten verstuurt over een internet-netwerk, gebruik makend van de Internet Protocollen. Dit zijn communicatie protocollen die op het internet gebruikt worden om de communicatie tussen computernetwerken mogelijk te maken.

### **Lan en Wan**

Local Area Network en Wide Area Network. De eerste is een computer netwerk dat computers in een beperkte omgeving met elkaar verbindt. Zoals een school, een bedrijf of een kantoor. Het WAN is een computer netwerk dat landen of regio's met elkaar verbindt. Data wordt over grote afstanden verzonden.

### **Packet switching<sup>8</sup>**

In een netwerk worden gegevens opgesplitst om in kleinere pakketten van variabele grootte verstuurd te worden. Door deze opdeling kunnen verschillende computers tegelijkertijd gebruik maken van een netwerk. Op het moment van verzending of ontvangst hoeft er geen verbinding te zijn tussen de stations. Als er even geen verkeer is, kunnen pakketjes van andere gebruikers over hetzelfde kanaal verzonden worden.

Voordelen:

- a) Gelijkmatige belasting van alle verbindingen.
- b) Wanneer bepaalde delen van het netwerk uit de lucht zijn neemt het pakketje een andere route.

Nadelen:

- a) De pakketjes kunnen wel eens in de verkeerde volgorde aankomen.

---

<sup>8</sup> <http://www.olavkwakman.nl/?cat=23>

b) De pakketjes kunnen ook niet aankomen.

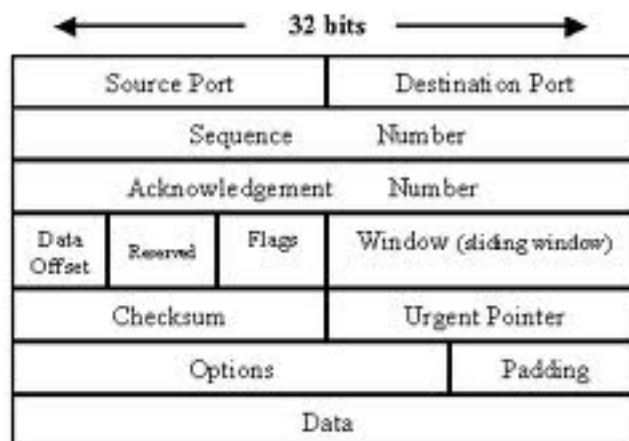
Kijk op:

[http://www.youtube.com/watch?v=a-e4HR7ZlaQ&feature=player\\_embedded](http://www.youtube.com/watch?v=a-e4HR7ZlaQ&feature=player_embedded)

## Packet

Een pakketje kan door een pakketgeschakeld netwerk verstuurd worden omdat het de juiste gegevens hiertoe bevat. En die zijn:

- a) Total length = Totale lengte van het pakketje
- b) Time to live = Na hoeveel tijd het pakketje weggegooid wordt omdat het kennelijk de weg niet heeft gevonden. Anders blijft het eeuwig rond cirkelen op internet.
- c) Source address = Wat is het ip-adres van de verzender.
- d) Destination address = Wat is het ip-adres van de ontvanger.



## Circuit switching

Een oudere vorm van communicatie over een netwerk. Een bepaalde verbinding krijgt gedurende de gehele overdracht een vast communicatiekanaal toegewezen. Zoals dat bij een telefoongesprek gebeurt. Tijdelijk wordt er een 'end-to-end' verbinding tot stand gebracht die exclusief is voor gebruik.

## Router

Een apparaat dat twee of meer verschillende netwerken met elkaar verbindt. Het is een soort schakelapparaat voor datapakketten. De router heeft informatie over de nabijgelegen andere routers en kan zo de datapakketten naar het juiste volgende station sturen.

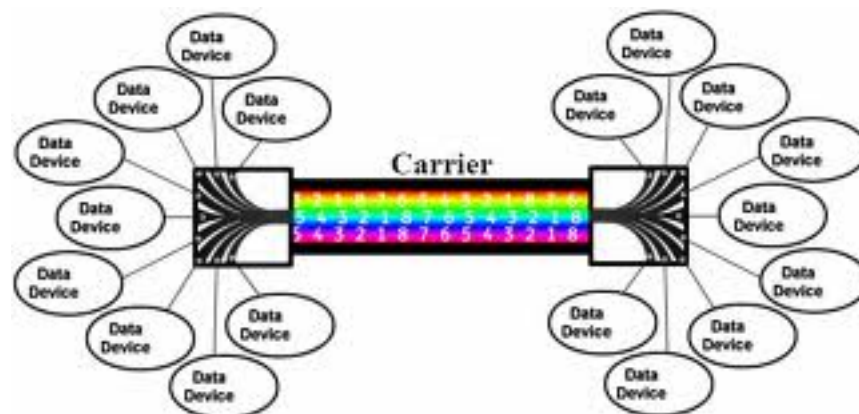
## Routeringsalgoritme

Hiermee wordt besloten over welke tussenstations in een netwerk de datapakketjes het snelst en veiligst verzonden worden. Een voorbeeld van zo'n algoritme is het Dijkstra algoritme:

<http://www.youtube.com/watch?v=wZYMgdB37Kw&feature=related>

## Multiplexing

Een proces waarbij meerdere analoge signalen of digitale datastromen worden gecombineerd tot één signaal. Het grote voordeel hiervan is dat een duur communicatiemedium gedeeld kan worden, en dus efficiënter gebruikt kan worden. Zo kunnen bijvoorbeeld verschillende telefoongesprekken over één lijn gevoerd worden, of kunnen verschillende datastromen over één dure glasvezelkabel verstuurd worden.



## 5. Besturingssystemen

Besturingssystemen zijn elementair in het gebruik van computers. Een belangrijke taak van het systeem is dat hij zorgt voor een vloeiende samenwerking tussen software en hardware.

Een besturingssysteem moet ervoor zorgen dat er geen lange wachttijd komt voor de processor. Daarbij wordt dan gebruik gemaakt van multitasking. De processen krijgen tijd toegewezen. De ene langer dan de andere. De ene ook eerder dan de andere, denk bijvoorbeeld aan je muis.

Ook heeft het als taak bestanden weer vindbaar te maken voor de gebruiker. Dit doet hij doormiddel van FAT en NTFS. Dit zijn tabellen waar wordt bijgehouden waar welk bestand staat en welk deel het omgeeft.

Hoofdtaken van het besturingssysteem zijn:

1. Het opstarten van het systeem.
2. Het verdelen van toegang, tot een systeembron als RAM, voor de verschillende programma's die tegelijkertijd draaien.
3. Uitvoeren van programma's. (Deze worden eerst naar het RAM geschreven en vervolgens door de CPU uitgevoerd)
4. Communicatie met randapparatuur.
  - a) Invoer: toetsenbord, muis.
  - b) Uitvoer: printer, monitor.
5. Multitasking: welk programma draait er op welk moment.
6. Energiebeheer bij computers die op batterijen werken.

## 6. Theorie bij praktijk

Nieuwe stof, die hoort bij de praktijkopdrachten

### Een webwinkel

Het leren van Php, MySQL

Het doel is een overzicht te geven van de samenwerking tussen verschillende talen. Het belang is om het onderscheid goed te blijven maken.

Het koppelen van een website aan een database met behulp van Php. Belangrijk hier is aan te geven dat de leerlingen in het eerste gedeelte van de vierde klas de HTML en CSS al hebben gedaan en dat de pagina's met de naam 'bekijken' en 'invoeren' werken met PHP-code. Dit is om de database te kunnen bevragen. De SQL-code wordt hierin 'ingebed'. De vergelijking met de Fetch and Execute Cycle van Von Neumann kan hier gemaakt worden. In de Informatica herhalen processen zich en komen in een andere gedaante steeds terug.

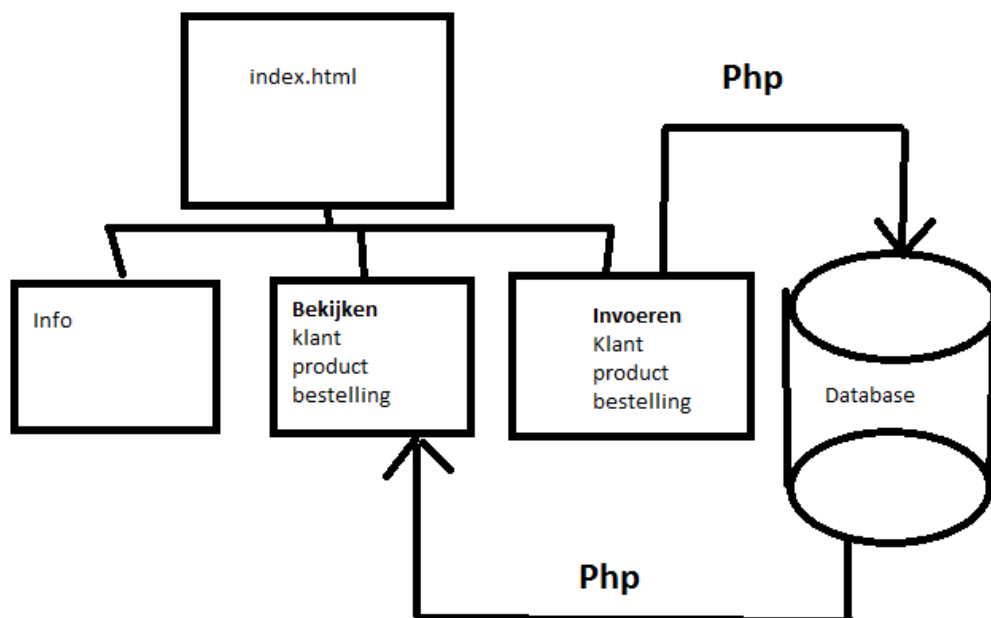


Fig. 2. Diagram van de verschillende talen. Css staat hier niet bij.

De logica wordt een stuk complexer omdat een platte webpagina nu met twee andere entiteiten, talen, verbonden wordt om tot extra functionaliteit te komen. Deze functionaliteit heeft het fenomeen Web 2.0 mogelijk gemaakt. Waarbij de gebruiker invloed krijgt op het web. Het internet is een interactief medium geworden.

Web 3.0 is waarbij internettoepassingen met elkaar gaan samenwerken. Internet 'weet' wie jij bent als gebruiker. Het is het volgende stadium in de ontwikkeling van internet. Web 1.0 is dat de gebruiker alleen kan kijken naar de webpagina. Hij kan hier geen interactie mee hebben.

De database is een opslag van gegevens in een vast format. Met ieder gegevens-object in een aparte tabel. Met behulp van koppelingstabellen worden de losstaande tabellen toch met elkaar in verband gebracht.

Het voordeel is dat er geen lange rijen van gegevens komen, zoals dat het geval zou zijn wanneer alles in een enkele tabel wordt gebracht. En dat er geen gegevens verloren gaan wanneer er, bijvoorbeeld, een klant wordt uitgethaald.

Een ander voordeel van SQL is dat het een gestructureerde taal is, over de hele wereld hetzelfde. Omdat de gegevens gestructureerd staan opgeslagen kan ieder gewenst stuk data opgehaald worden en gepresenteerd. Ook kunnen gegevens met elkaar gecombineerd worden.



## 8. Systeem

Nieuw

De samenwerking tussen de verschillende onderdelen in de computer en de kosten afgezet tegen snelheid, worden in het hieronder staande plaatje schematisch weergegeven.

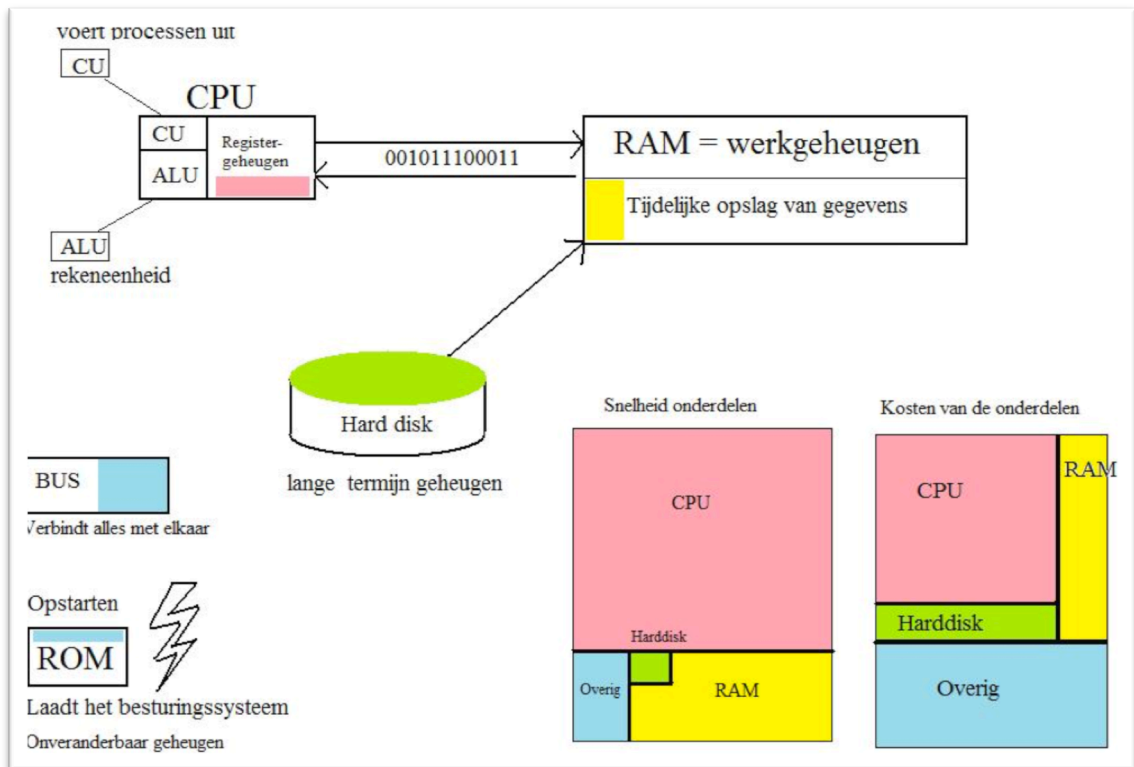


Fig. 3. Schema van het systeem