

# **WEB-BASED INFORMATION SYSTEMS**

**DELFT, JULI 2001**

Mark Dumay

**Technische Universiteit Delft**  
**Faculteit Informatie Technologie en Systemen**  
Afdeling Technische Informatica



## Voorwoord

Dit verslag behoort bij het vak State of the Art Applications (in4068), wat deel uitmaakt van het keuzeprogramma voor Technische Informatica. Dit vak, dat gegeven wordt in twee blokken van elk 7 colleges, gaat in op enkele huidige ontwikkelingen op het gebied van de informatica.

Het onderwerp Web-based information systems is tijdens de collegereeks in een tweetal colleges aan bod gekomen. Door het maken van dit rapport is een verdieping in het daar aangereikte materiaal bewerkstelligd.

Mark Dumay,

Delft, juni 2001.

## **Inhoudsopgave**

<b>VOORWOORD</b>	<b>2</b>
<b>1 INLEIDING</b>	<b>4</b>
<b>2 DEFINITIE VAN INFORMATIESYSTEMEN</b>	<b>5</b>
2.1 VAN INFORMATIETECHNOLOGIE NAAR INFORMATIESYSTEEM	5
2.2 OPKOMST VAN INTERNET	6
2.3 ORIËNTATIE VANUIT HET WWW	6
<b>3 ARCHITECTUUR VAN INFORMATIESYSTEMEN</b>	<b>7</b>
3.1 DE N-TIER ARCHITECTURE	7
3.2 VOORDELEN VAN DE 3-TIER ARCHITECTUUR	9
3.3 ARCHITECTUUR VOOR WEB APPLICATIES	9
3.4 ASP MODEL	9
3.5 HUIDIGE ONTWIKKELINGEN	10
<b>4 SPECIFIEKE KARAKTERISTIEKEN WIS</b>	<b>11</b>
4.1 MARKTUITDAGINGEN	11
4.2 TECHNISCHE UITDAGINGEN	11
<b>5 CONCLUSIE</b>	<b>13</b>
<b>REFERENTIES</b>	<b>14</b>

# 1 Inleiding

In dit verslag wordt in gegaan op een nieuw fenomeen binnen de informatica, de zogenaamde web-based information systems (WIS). Deze nog jonge ontwikkeling biedt niet alleen veel nieuwe mogelijkheden, maar heeft ook te kampen met zijn eigen specifieke problemen.

Allereerst wordt een beschrijving gegeven van informatiesystemen in het algemeen, en een WIS in het bijzonder. Vervolgens wordt vanuit een technisch perspectief uiteengezet hoe een WIS zich verhoudt tot een traditioneel informatiesysteem, ook wel aangeduid als *enterprise application*. Tenslotte wordt in gegaan op enkele specifieke kenmerken van een WIS, gericht op de marktsituatie en de bijbehorende technische uitdagingen.

Aan de hand van de hiervoor beschreven onderwerpen wordt beargumenteerd of web-based information systems een geheel nieuw onderzoeksgebied op zich zijn, of een ontwikkeling van bestaande theorieën en praktijkervaring van traditionele informatiesystemen.

## 2 Definitie van informatiesystemen

Juist vanwege de grote vlucht die informatiesystemen de laatste paar decennia hebben gemaakt, worden er veelal ook meerdere betekenissen toegekend aan het begrip informatiesysteem. In de bewoordingen van Alter komt het globaal op het volgende neer: 'Een informatiesysteem is een systeem dat informatie technologie gebruikt om informatie, die gebruikt wordt in een of meerdere bedrijfsprocessen, te verkrijgen, te verzenden, op te slaan, op te vragen, te manipuleren of weer te geven.'<sup>1</sup> Deze definitie leert dat informatiesystemen hoe dan ook te maken hebben met processen binnen een bedrijf, en daar ook niet zelfstandig van kunnen opereren.

### 2.1 Van informatietechnologie naar informatiesysteem

Vanwege de veranderende structuur van vele organisaties, veranderen ook de eisen aan de gebruikte informatie technologie. In de begin jaren '60, toen IT toepassingen voor het eerst op brede schaal werden ingevoerd, waren vele organisaties opgedeeld in verscheidene functionele domeinen. Doel was om taken die binnen het bedrijf voorkwamen zoveel mogelijk te bundelen en afzonderlijk aan te sturen. Met de komst van de eerste automatiseringsvraagstukken, werden deze afdelingen echter veelal afzonderlijk bediend, wat sterk werd beïnvloed door de tot dan toe geldende norm van onafhankelijke aansturing. Stapsgewijze invoer van automatisering had echter als belangrijk nadeel dat er zogenaamde informatie-eilanden ontstonden, met ieder een andere aanpak.

Aan het eind van de 20<sup>e</sup> eeuw werden vele organisaties heringedeeld, veelal gebaseerd op bedrijfsprocessen<sup>2</sup>. Organisaties werden opgedeeld in zogenaamde business units, oftewel in kernactiviteiten. Deze opdeling ging over de tot dan toe geldende functionele indeling heen, wat als gevolg had dat tot dan toe zelfstandig opererende afdelingen plotseling met elkaar moesten gaan samenwerken. Dit had ook gevolgen voor de al ingevoerde IT-systemen. Deze konden niet langer meer afzonderlijk van elkaar opereren, maar moesten met elkaar kunnen communiceren. Hierbij ontstonden ook problemen bij verschillende systemen, die in wezen hetzelfde procesondersteunde. Een klantenregistratie is bijvoorbeeld op meerdere plekken in de organisatie nodig, maar elke keer zijn de eisen enigszins anders in detail.

Vanwege de opdeling van bedrijven in bedrijfsprocessen, kwam ook het inzicht dat IT niet langer losstaand hiervan kan worden ingezet. Dit leidde tot het ontstaan van het begrip informatiesysteem. Belangrijk om hierbij op te merken is, dat beide onderdelen sterk op elkaar ingrijpen, iets wat ook wel wordt aangeduid als het *informatieparadigma*<sup>3</sup>.

---

<sup>1</sup> Zie [01] (p2, pp4).

<sup>2</sup> Zie [01] (p59).

<sup>3</sup> Zie [06] (p20).

## 2.2 Opkomst van internet

Met de opkomst van het internet, en met name het zogenaamde worldwide web (WWW) is communicatie een nog sterker bepalende factor geworden bij het toepassen van IT. Na algemene aanvaarding van een technisch protocol voor het onderling communiceren van verschillende netwerken (TCP/IP)<sup>4</sup>, heeft de communicatie via intra- en internet voor vele organisaties een grote vlucht genomen. Daarnaast krijgen organisaties ook een nieuw communicatiemiddel met hun klanten.

De toepassing van internet in de organisatie gaf ook nieuwe eisen aan de bestaande informatiesystemen. Dit fenomeen kan als volgt worden aangeduid: een WWW-gebaseerd informatiesysteem kan worden gedefinieerd als een toepassing die niet alleen informatie verspreidt, maar ook actief interactie heeft met de gebruiker om hem te ondersteunen in zijn taak [07]. Hierbij is de term WWW-gebaseerd informatiesysteem overeenkomstig met Web-based information system (WIS).

## 2.3 Oriëntatie vanuit het WWW

In dit hoofdstuk is voor het ontstaan van WIS een oriëntatie vanuit de informatiesysteemkant gekozen, en dan met name de zogenaamde *enterprise applications*. In dit licht lijkt WIS ten opzichte van een traditioneel IS dan ook een logische ontwikkeling. Vanuit het perspectief van het World wide web is er echter een veel grotere verschuiving te zien. In [05] wordt dit verwoord als zijnde een dynamisch en databasegecentreerd systeem. Dit houdt in dat de pagina's die getoond worden in de browser zowel dynamisch aan elkaar gekoppeld zijn, als dynamisch worden aangemaakt. De stap van statische pagina's naar dynamische applicaties is hierdoor veel minder eenduidig en met veel meer problemen gaande.

---

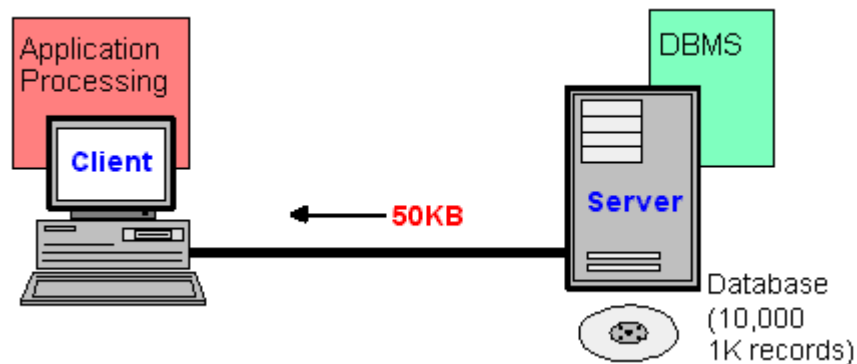
<sup>4</sup> Zie [08] (p35).

### 3 Architectuur van informatiesystemen

Bij het implementeren van informatiesystemen, ofwel de zogenaamde *enterprise applications*, is de laatste jaren een ontwikkeling te zien in de gebruikte architecturen.

#### 3.1 De n-tier architecture

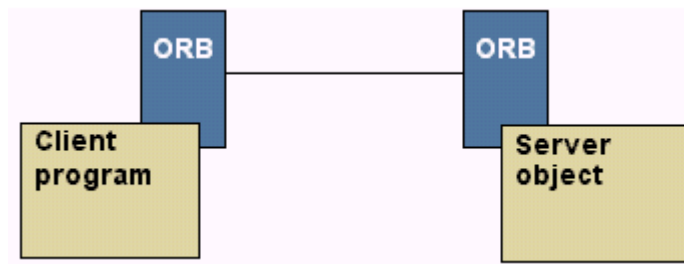
Traditioneel gezien werden informatiesystemen als client/server architectuur geïmplementeerd. De server bevat hierbij de dataopslag, gereguleerd middels een specifiek DBMS. Applicatielogica kan hierbij zowel op de server als op de client staan. Het is hierbij van belang dat de data consistent gehouden wordt, aangezien meerdere clients tegelijkertijd kunnen werken met dezelfde data.



figuur 3-1 two-tier client/server architectuur

In figuur 3-1 staat een voorbeeld van een 2-tier client/server architectuur, waarbij de eigenlijke programmacode zich op de client bevindt. De server garandeert in dit geval de integriteit van de data. In deze variant hebben de verschillende clients geen weet van elkaar, wat met name zaken als toegang tot data en het gelijktijdig wijzigen ervan bemoeilijkt.

In de jaren '90 kwamen de zogenaamde object-request brokers (ORB) in opkomst, ook wel aangeduid met de meer algemene term *middleware* [09]. Voorbeelden hiervan zijn DCOM en CORBA. In wezen zijn dit mechanismen die het mogelijk maken dat processen overal in een netwerk kunnen worden gedraaid.



figuur 3-2 Object-request broker

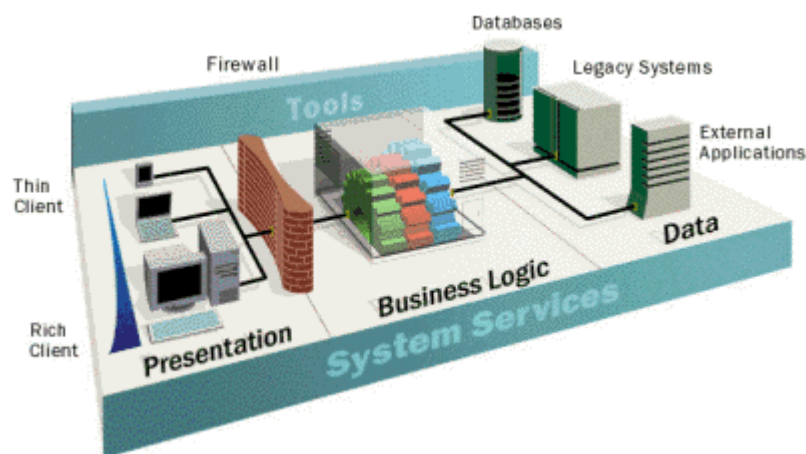
De ORBs maakten het mogelijk om verschillende delen van het informatiesystemen te herstructureren en te herverdelen. In de praktijk betekende dit de opkomst van het fenomeen *application server*, oftewel de mogelijkheid tot het onderbrengen van de applicatielogica op een aparte server. Dit concept is verder uitgewerkt en staat bekend als de 3-tier architectuur. Het is gebaseerd op het concept dat elke oplossing kan worden onderverdeeld in drie verschillende functionele domeinen:

- Data;
- Business;
- Presentatie.

De data laag bevat hierbij de centrale databaseopslag. Dit wordt meestal verzorgd door een DBMS, zoals bijvoorbeeld Microsoft SQL Server. Toegangsbeheer en integriteit van de gegevens zijn hierbij twee cruciale factoren.

De businesslaag, ook wel applicatielaag geheten, bevat de business rules van de applicatie. Het kan ook een workflow engine bevatten, welke de verschillende processen reguleert. Belangrijk hierbij is dat de meeste ORBs transactiegeoriënteerd werken, in analogie met de relationele database wereld. De wisselwerking met de database is ook een van de grootste uitdagingen, mede door de aansluiting van objecten op de veelal relationele databaseopslag.

De presentatielaag is de eigenlijke front-end voor de gebruiker, en kan op meerdere manieren worden geïmplementeerd. Juist doordat veel van de applicatielogica (zo niet alle) zich op een server bevindt, zijn de eisen aan de client meestal vrij bescheiden.



figuur 3-3 Windows DNA

In figuur 3-3 wordt een voorbeeld gegeven van Microsoft's visie op de 3-tier architecture, iets wat tot een ruim een jaar geleden werd aangeduid als Windows DNA (distributed network application).



## 3.2 Voordelen van de 3-tier architectuur

Met het gebruik van de 3-tier architectuur, worden een aantal zaken gerealiseerd met betrekking tot informatiesystemen. Allereerst zorgt de opdeling in drie verschillende functionele domeinen voor een meer flexibel systeem. Door de scheiding tussen gegevens, applicatielogica en presentatie heeft een wijziging in een van de domeinen minder invloed op de rest van het systeem. Door daarnaast binnen de domeinen zelf ook gebruik te maken van afzonderlijke delen, ontstaat een erg flexibel systeem, dat snel aan te passen en uit te breiden is.

Ook de schaalbaarheid heeft baat bij deze architectuur. In theorie kunnen alle drie de lagen zich op één en hetzelfde systeem bevinden, maar ook kan een laag door meerdere machines worden verzorgd. Inzet van machines op de plek waar het nodig is, verbetert de doeltreffendheid en hiermee de schaalbaarheid.

De systeemontwikkeling in teamverband kan tevens voordeel hebben van deze opzet. Allereerst wordt het eenvoudiger om afzonderlijke onderdelen onafhankelijk te ontwikkelen. Daarnaast vindt er een formalisatieslag plaats, doordat er meer eisen worden gesteld aan de wisselwerking tussen de verschillende lagen.

## 3.3 Architectuur voor web applicaties

Vanuit het perspectief van het 3-lagen model (zie paragraaf 3.1) veranderen er vanwege de introductie van het web niet veel zaken aan de implementatie. In dit geval is het enige dat wezenlijk verandert de implementatie van de presentatielaag. In plaats van een traditionele applicatie, zoals die bijvoorbeeld veel zijn te zien in Microsoft Windows, wordt nu gebruik gemaakt van een web-browser. De benodigde HTML code, de taal die universeel wordt gebruikt bij het aanbieden van websites, kan dan worden gegenereerd op een webserver, die gekoppeld is aan de applicatielaag van de oplossing. Een voorbeeld hiervan is Microsoft Internet Information Server, die middels zogenaamde Active Server Pages (ASP) de benodigde code kan creëren.

## 3.4 ASP model

Naast de 3-tier architectuur is recentelijk een nieuwe beweging in gang gezet, die wordt aangeduid als het Application Service Provider model. Volgens [03] komt het neer op het volgende: 'Een ASP [...] is een onafhankelijke dienstverlener, welke op afstand een software applicatie installeert, configureert en beheert middels centraal gelocaliseerde servers op basis van een huur- of lease-overeenkomst.'<sup>5</sup> Deze definitie is echter niet volledig dekkend wat betreft de technische realisaties die op dit gebied tot stand zijn gekomen.

---

<sup>5</sup> Zie [03] (p5, pp1).

In analogie met het three-tier model, zou een architectuurrealisatie van een ASP een zogenaamd *service-model* kunnen heten. Bepaalde diensten worden ontwikkeld als zelfstandige applicatie, die vervolgens door andere applicaties weer kunnen worden aangesproken. Dit idee wordt ook wel aangehaald als de 4-tier architectuur, waarbij de diensten worden aangeboden aan de applicatielaag. Het is niet helemaal verschillend van het concept *component based development*, waarbij een applicatie in meerdere, logisch samenhangende en vrijwel zelfstandige softwarecomponenten wordt opgedeeld.

### 3.5 Huidige ontwikkelingen

Met betrekking tot de architectuur van enterprise/web applicaties, zijn er diverse ontwikkelingen op het implementatieplatform gaande. Zo is er als voorbeeld het .NET platform van Microsoft, dat naar verwachting begin 2002 zijn intrede zal vinden. Het is een soort van dienstenraamwerk, die als een schil op het besturingsysteem draait. Vele taken die nu middels verschillende technieken op bijvoorbeeld de applicatielaag en front-end generatie tot nu toe moesten worden gerealiseerd, kunnen dan middels een eenduidige programmeerinterface worden verwezenlijkt. Ook Sun is bezig met een soortgelijk initiatief.

Een andere ontwikkeling is die van *multi-channeling*. Met name de opkomst van mobiele telefoons met toegang tot internet of en variant daarvan, heeft een impuls gegeven aan de opkomst van informatiesystemen die op meerdere manieren benaderd kunnen worden. Niet langer is het toegankelijk maken van een bestaand informatiesysteem via internet meer voldoende, ook via de telefoon moet het te benaderen zijn. Het besef dat deze systemen op meerdere manieren benaderd moeten kunnen worden, heeft geleid tot de multi-channeling gedachte. Hierbij wordt middels specifieke interface-applicaties, gericht op een bepaald kanaal, toegang gerealiseerd tot het informatiesysteem.

## 4 Specifieke karakteristieken WIS

In hoofdstuk 2 en 3 is een beschrijving van web-based IS gegeven ten opzichte van traditionele *enterprise applications*. Hierbij is met name vanuit de architectuurkant gekeken naar een mogelijke realisatie op dit gebied, wat suggereert dat beide systemen, WIS en IS, hierbij niet sterk van elkaar verschillen. Toch zijn er enkele specifieke zaken aan te wijzen op het gebied van WIS, die hier zullen worden besproken.

### 4.1 Marktuitedagingen

Vanwege het karakter van het internet, worden met name door potentiële klanten van WIS nogal wat verwachtingen geplaatst bij nieuwe diensten. Met name enkele jaren geleden, bij de grootschalige acceptatie van het internet, was het voor veel bedrijven zaak om zo snel mogelijk met hun informatiesysteem op het internet bereikbaar te zijn, of nieuwe applicaties specifiek voor deze doelgroep te ontwikkelen. In veel gevallen was het belangrijker dat er in ieder geval een dienst was, in plaats van de kwaliteit er van. Recentelijk is hier enigszins een kentering ingekomen, maar toch blijft de *time-to-market* een prominente rol spelen bij de ontwikkeling van een WIS. Om aan de kwaliteitseisen te voldoen, wordt nu vaak gekozen voor een gefaseerde introductie. Het project m-bankieren van de Postbank is hier een voorbeeld van [02].

Een tweede aspect dat om de hoek komt kijken is het uiteindelijke klantenbereik. Het is vooraf vaak moeilijk om een precieze inschatting te maken van het aantal mensen dat een WIS wil gaan gebruiken. Ook komt het voor dat een bepaalde dienst opeens explosief in gebruik toeneemt, wat kan veroorzaken dat de infrastructuur dit niet meer kan verwerken. De introductie van een online cd-winkel van de Free Record Shop in 1998 is hier een typisch voorbeeld van [04].

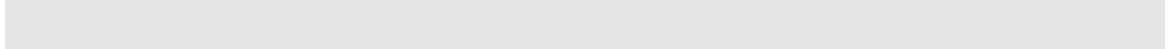
### 4.2 Technische uitdagingen

Als gevolg van de twee specifieke marktuitedagingen die in paragraaf 4.1 zijn besproken, is een aantal technische uitdagingen te onderkennen. Het dient niet onuitgesproken dat dergelijke aspecten ook bij de implementatie van traditionele informatiesystemen gelden, maar met name bij WIS spelen ze een veel belangrijkere rol.

Allereerst is flexibiliteit van het systeem een belangrijke uitdaging. Vanwege de korte doorlooptijden en het snel kunnen inspelen op de markt, is het gewenst dat het systeem snel uit te breiden en aan te passen is. Een mogelijke oplossing is het opdelen van een applicatie in een aantal componenten. Door interne koppelingen via vaste *interfaces* te doen verlopen, is de onderlinge afhankelijkheid vaak al een stuk minder groot. Hierdoor kan bijvoorbeeld eenvoudiger de functionaliteit worden uitgebreid, of code worden aangepast zonder dat dit gevolgen heeft voor de rest van het systeem.

Een tweede belangrijke gewenste karakteristiek is de beschikbaarheid en betrouwbaarheid van het systeem. Dit kan verder worden uitgesplitst in een tweetal verschillende indicatoren. Doordat er moet kunnen worden omgegaan met onverwachte groei, is het van belang dat het systeem schaalbaar is. Er zijn twee verschillende vormen van schaalbaarheid, die bekend staan als scale-up or scale-out. Scale-up is het simpelweg verzwaren van de hardware, zoals bijvoorbeeld het plaatsen van extra geheugen of processoren. Dit vormt meestal geen extra implicatie voor de software-implementatie. Het is echter meestal een onvolledige oplossing, omdat het inzetten van zwaardere hardware op een gegeven moment begrensd wordt. Scale-out is het distribueren van de *server-load* over meerdere machines. Door het bijplaatsen van extra servers wordt de individuele last per server verminderd. De software-implementatie moet dan wel gedistribueerd opgezet van, daarnaast is er specifieke software nodig die taken verdeeld over de meerdere servers.

Naast de schaalbaarheid van het systeem, is met name toegangsbeveiliging bij WIS een belangrijk onderwerp. Doordat de applicaties voor een veel groter publiek toegankelijk zijn, ontstaat er een ook een groter risico voor misbruik of sabotage. Met name nu veel processen binnen een bedrijf grotendeels afhankelijk zijn van informatiesystemen is dit een belangrijk gegeven. Ook een onderwerp als bedrijfsspionage kan hierbij een rol spelen.



## 5 Conclusie

Web-based information systems (WIS) is nog een vrij nieuwe ontwikkeling binnen de IT-wereld, maar mag zich nu al verheugen op een grote belangstelling. Veel van de problemen die zich bij de ontwikkeling en invoer van dergelijke systemen voordoen, lijken zich hierbij op het eerste gezicht pas voor het eerst voor te doen. Het is echter de vraag of ze een geheel nieuwe aanpak vereisen, of gelijkenissen vertonen met die van traditionele informatiesystemen.

Op het vlak van de moderne informatiesysteemontwikkeling is een belangrijke ontwikkeling die van de three-tier architectuur. Door de conceptuele scheiding tussen gegevens, bedrijfslogica en presentatie is het uiteindelijke systeem flexibeler van opzet en is het beter schaalbaar. Met de komst van WIS worden alleen andere eisen gesteld aan de presentatie, maar verandert het algemene concept niet.

Twee markttuitdagingen springen bij WIS in het oog: de *time-to-market* en onvoorspelbare omvang. Het eerste begrip, het zo snel mogelijk invoeren van een WIS, is moeilijk te bevatten. Gefaseerde invoer is hierbij een mogelijke oplossing, die de kwaliteit van de dienst waarborgt. Dit noodzaakt een flexibele opzet. De onvoorspelbare omvang van gebruikers dient van begin af aan in ogenschouw genomen te worden, waar schaalbaarheid van het systeem de bepalende factor is.

Met als basis de three-tier architectuur, gelden voor WIS weinig verschillen met de ontwikkeling van traditionele informatiesystemen. Door een andere marktsituatie zijn twee kenmerken, flexibiliteit en schaalbaarheid, echter wel meer bepalend voor een succesvolle invoer. WIS legt dan ook met name meer druk op al bestaande eisen aan informatiesystemen.

## Referenties

- [01] ALTER, S., 1996, *Information systems. A management perspective*, The Benjamin/Cummings Publishing Company, Inc., Menlo Park.
- [02] BEZEM, J., 'Giroblauw wapt voor jou', *Computable*, 18 mei 2001, nr 20, p30. [Online] Zie: <http://www.computable.nl/artikels/archief1/d20ra103.htm>.
- [03] CHERRY TREE & CO., 1999, "*Application Service Providers*" (ASP). *Spotlight Report*. [Online] Zie <http://www.asp.com/pdf/CherryTree1.pdf>.
- [04] COMPUTABLE, 'Virtuele winkel Free Record Shop lijkt voor Oracle te hoog gegrepen', *Computable*, 23 januari 1998, nr 4, p7. [Online] Zie <http://www.computable.nl/artikels/archief8/d04hb8pf.htm>.
- [05] GELLERSEN, H., WICKE, R. & GAEDKE, M., 1997, *WebComposition: An object-oriented support system for the web engineering lifecycle*, Proceedings of the Sixth International World Wide Web Conference. [Online]. Zie: <http://www6.nttlabs.com/HyperNews/get/PAPER232.html>.
- [06] HENGST, M., 2001, *Analyse van Bedrijfssystemen*, Faculteit Techniek, Bestuur en Management, TU Delft.
- [07] TAKAHASHI, K. & LIANG, E, 1997, *Analysis and design of web-based information systems*, Proceedings of the Sixth International World Wide Web Conference. [Online]. Zie: <http://www6.nttlabs.com/HyperNews/get/PAPER245.html>.
- [08] TANENBAUM, A.S., 1996, *Computer Networks*, 3<sup>rd</sup> edition, Prentice-Hall, Inc., New Jersey.
- [09] THE COMPUTER LANGUAGE COMPANY, *TechEncyclopedia*. [Online]. Zie: [www.techencyclopedia.com](http://www.techencyclopedia.com).