

CASE, een introductie

Drs. E.G.F.M. Dortmans

Informatica Adviseur
Directie Organisatie & Efficiency
Ministerie van Landbouw, Natuurbeheer en Visserij
Postbus 20401, 2500 EK 's-Gravenhage
Telefoon 070-3792676, telefax 070-3479097

Ir. E. ten Harkel

Consultant
ISES International
Postbus 8255, 3503 RG Utrecht
Telefoon 030-944141, telefax 030-949830

agro informatica 6(5) / december 1993

Inleiding

De laatste jaren wordt er zowel binnen als buiten de agrarische sector volop aandacht besteed aan de mogelijkheden voor het gebruik van CASE-technologie bij het realiseren van informatiesystemen. In dit artikel willen wij een aantal vragen rondom het gebruik van CASE-technologie beantwoorden.

Als eerste gaan we in het kort in op het 'waarom' van het gebruik van CASE-tools. Daarna wordt een model gepresenteerd voor de classificatie van de geboden functionaliteit van CASE-tools. Vervolgens worden de ontwikkeling en de verschillende verschijningsvormen van CASE beschreven. Dan wordt ingegaan op de selectie en implementatie van CASE-technologie in organisaties, waarna tenslotte kort enkele gevolgen van het ge-

bruik van CASE-technologie worden beschreven.

Waarom CASE? Het CASE dilemma

Als we naar de huidige manier van systeemontwikkeling binnen veel organisaties kijken dan wordt een aantal problemen zichtbaar:

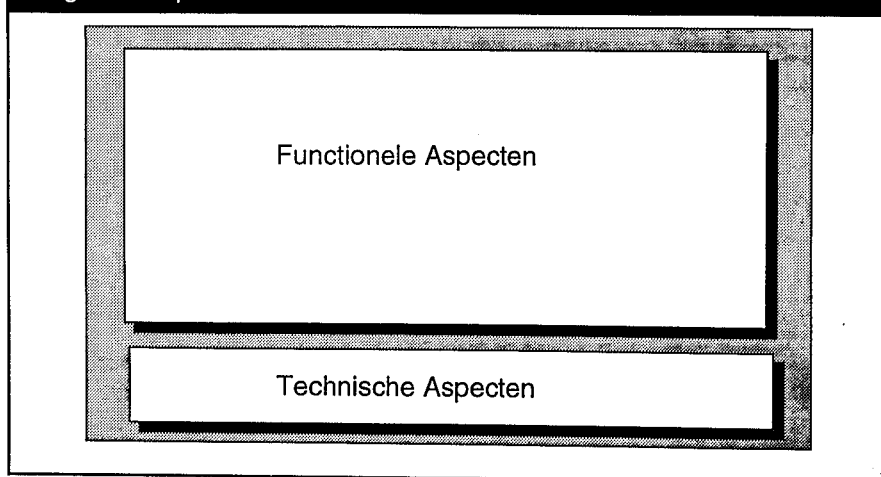
- De omvang systemen en complexiteit van projecten worden steeds groter;
- Bedrijfsbehoeften wisselen sneller waardoor een discrepantie tussen eisen en de beschikbare middelen ontstaat. Met andere woorden, men is niet voldoende in staat snel genoeg systemen op te leveren, en streeft naar doorlooptijdverkorting;
- Er is een toenemende opkomst van technologische "wondermiddelen" als client-server architecturen. Die lijken

de oplossing voor vele kwalen maar kunnen, zegt men, alleen met behulp van CASE-technologie beheersbaar ontwikkeld worden. Daardoor ontstaat een toenemende behoefte aan het gebruik van methoden en technieken (zie onder andere de Landbouw-Informatica-Aanpak (LIA). Als deze methoden en technieken echter niet door hulpmiddelen ondersteund worden, zal het gebruik al snel verminderen. Er bestaat dus een steeds grotere druk om met behulp van CASE systemen te gaan ontwikkelen.

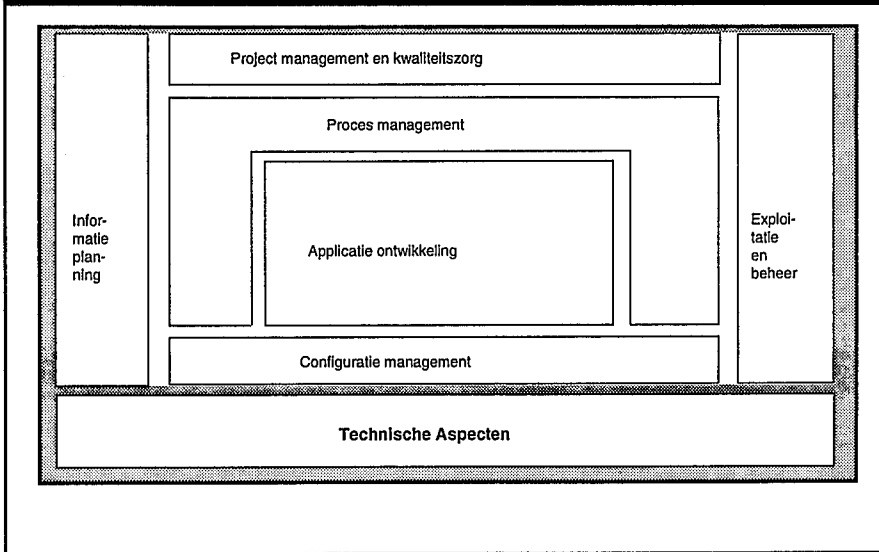
Daar tegenover bestaan er problemen bij de invoering en het gebruik van CASE-technologie:

- Een uniforme en breed geaccepteerde definitie van het begrip CASE begint nu pas te ontstaan. De vlag CASE dekt zeer verschillende ladingen;
- Doelstellingen voor gebruik van CASE-tools verschillen enorm. De Nederlandse Gebruikersgroep Gestructureerde Ontwikkelmethoden (NGGO) heeft in 1992 een onderzoek verricht naar het gebruik van CASE-tools in Nederland. Hieruit kwamen onder meer zeer verschillende doelstellingen voor CASE-gebruik naar voren (Kusters, 1992).
- Bovendien blijkt er een grote discrepantie tussen doelstellingen in een CASE-implementatie-traject en realisatie daarvan. Ervaring leert dat de door de leveranciers voorgespiegelde

Figuur 1 - Aspecten van CASE



Figuur 2 - Invulling functionele aspecten



produktiviteits- en kwaliteitsverbeteringen slechts mondjesmaat en dan nog sterk vertraagd worden gerealiseerd.

Dit maakt de keuze voor CASE een dilemma; enerzijds is de behoefte groot, anderzijds roept men door de invoering van CASE-technologie nieuwe en onvoorspelbare problemen over zich af. In dit artikel willen we proberen in dit dilemma enig licht te brengen.

Wat is CASE?

De term CASE staat voor Computer Aided Systems Engineering: het met behulp van geautomatiseerde hulpmiddelen ontwikkelen van informatiesystemen. Deze hulpmiddelen bieden de ontwikkelaar ondersteuning bij het uitvoeren van zijn of haar werkzaamheden. Wanneer in dit artikel over CASE wordt gesproken dan wordt hiermee bedoeld:

Een set van geautomatiseerde hulpmiddelen die de ontwerp-, modellerings- en bouwactiviteiten van het systeemontwikkelingstraject met meerdere technieken geïntegreerd ondersteunt, gebruik makend van een (de-)centrale database van specificatiegegevens.

Synoniem aan CASE zijn de termen Hulpmiddel, Tool en Workbench. We zullen deze termen in dit artikel dan ook door elkaar gebruiken. In deze paragraaf zullen wij een model presenteren dat het moge-

lijk maakt om de verschillende op de markt te verkrijgen CASE-tools ten opzichte van elkaar te classificeren. Daarbij maken we allereerst onderscheid tussen de geboden functionele ondersteuning en de technische aspecten van tools (zie figuur 1).

De functionele ondersteuning

De beschrijving van de functionele ondersteuning gebeurt aan de hand van een model voor de inrichting van de ontwikkelinfrastructuur. Doel van dit model is het weergeven van alle activiteiten die men gedurende een traject van het ontwikkelen van een informatiesysteem onderscheidt. Methoden en CASE-technologie bieden immers aan één of meerdere van

deze activiteiten ondersteuning (Harkel & Dortmans, 1993).

De volgende activiteiten worden onderscheiden (figuur 2):

Informatieplanning

Het verzamelen van en nemen van besluiten over de uitgangspunten van de informatievoorziening, doelstellingen van de organisatie, standaards die gevolgd moeten worden, leveranciers waarvan apparatuur of diensten betrokken moeten worden, randvoorwaarden voor de komende veranderingen, etc..

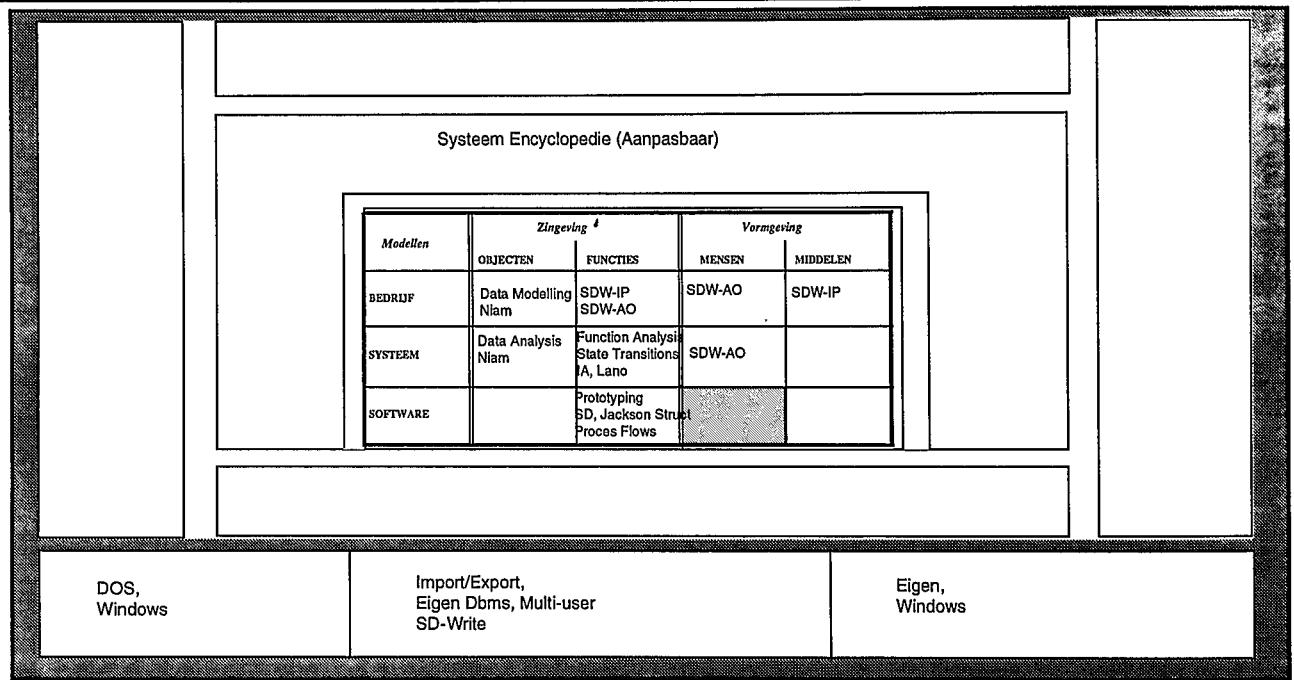
Applicatie-ontwikkeling

Hier worden de door methoden en tools geboden ondersteuning opgenomen voor het opstellen van de specificaties en het ontwerp van de informatiesystemen. Met specificaties wordt hier de verzameling modellen bedoeld die gezamenlijk de functionaliteit van een informatiesysteem beschrijven. Om de methoden en tools binnen dit gebied te kunnen beschrijven wordt het ISSES-raamwerk gebruikt (Laagland, 1992), zie figuur 3. In dit raamwerk worden drie abstractie-niveaus onderscheidend, te weten: Bedrijf, Systeem en Software. Ieder niveau heeft zijn eigen doelstelling en een daarvoor geëigende beschrijvingstaal, bestaande uit een aantal methoden en/of technieken. Verder wordt een onderscheid gemaakt tussen de zinge-

Figuur 3 - ISSES-Raamwerk

Modellen	Zingeving		Vormgeving	
	OBJECTEN	FUNCTIES	MENSEN	MIDDELEN
BEDRIJF	Informatie-model	Activiteiten model	Organisatiestructuur	Technische infrastructuur Informatiesystemen
SYSTEEM	Gegevens-model	Proces-model	Administratieve organisatie	Hardware/software configuratie
SOFTWARE	Database-structuur	Programma-structuur		Software bouwstenen

Figuur 4 - ISES-Raamwerk voor SDW



agro informatica 6(5) / december 1993

ving (de specificatie) en vormgeving (de configuratie).

Projectmanagement en kwaliteitsborging

Tot deze categorie behoren alle activiteiten die erop zijn gericht te bepalen welke acties nodig zijn voor het beheersen van de uitvoering van een project en het borgen van de kwaliteit van de werkzaamheden en de opgeleverde producten.

Exploitatie en beheer

Dit onderdeel van het raamwerk beschrijft alle activiteiten die kunnen worden gebruikt bij het exploiteren en het beheren van de ontwikkelde applicaties.

Procesmanagement

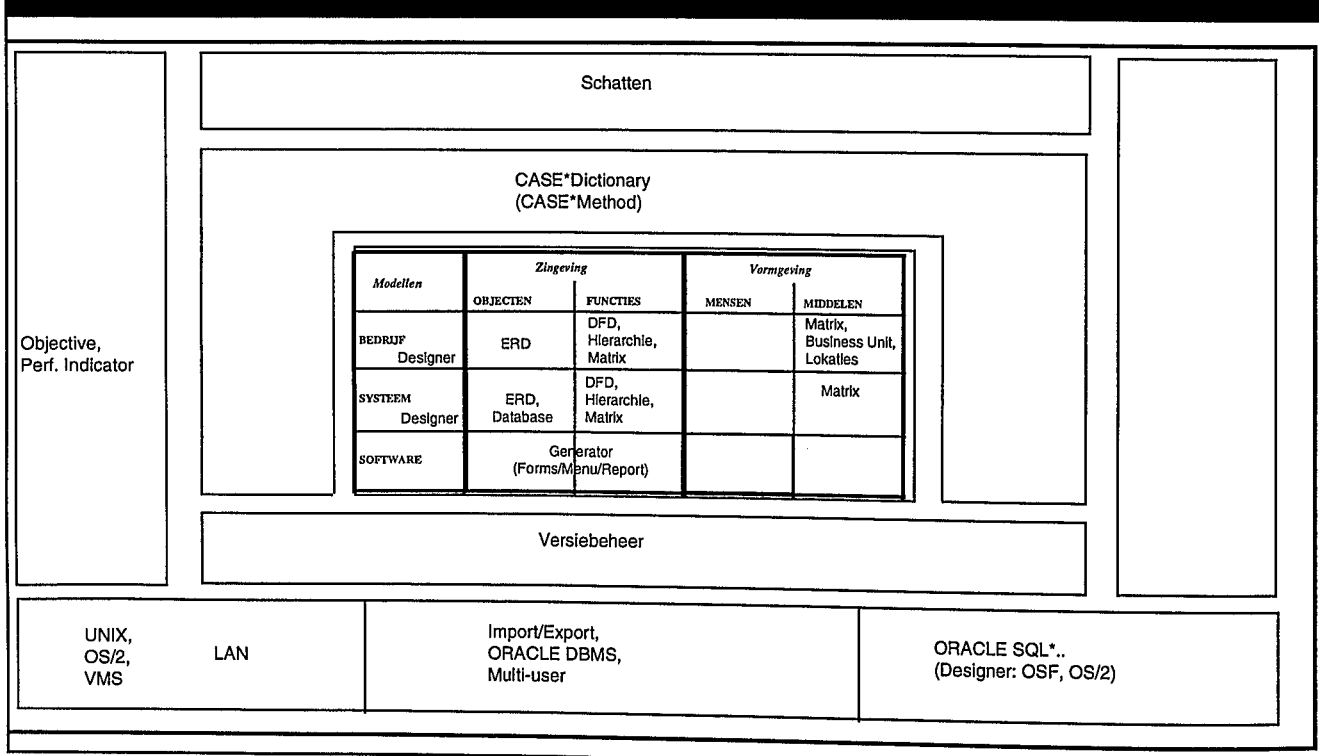
Activiteiten die hiertoe behoren hebben de planning, uitvoering en beheersing van de andere activiteiten tot doel. Dus het omzetten van de algemene standards en uit-

gangspunten beschreven bij informatieplanning, projectmanagement en exploitatie en beheer, in modellen van het bedrijf, het systeem en software. Procesmanagement beheert de volgorde van de activiteiten en de consistentie tussen de resultaten van de activiteiten.

Configuratie-management

In deze categorie vallen alle activiteiten, die het beheren van de tussen- en eindpro-

Figuur 5 - ISES-Raamwerk voor Oracle*CASE



ducten van het ontwikkeltraject tot doel hebben.

De technische aspecten

Van de technische aspecten worden de volgende aspectgebieden onderkend.

Mens-machine-interface

Het aspectgebied mens-machine interface beschrijft een aantal kenmerken van het user-interface, al dan niet grafisch, dat door het tool wordt gebruikt bij invoeren en representeren van de ontwikkelgegevens.

Repository

Het aspectgebied repository (=database) beschrijft een aantal kenmerken van de repository, die door het tool wordt gebruikt om de ontwikkelgegevens op te slaan. Opgenomen zijn onder meer:

- beveiliging van gegevens;
- aanpasbaarheid van het meta-model van de repository;
- consistentiecontrole tussen de objecten in de repository;
- interfaces tussen verschillende producten, etc.

Hardware/netwerk

In dit aspectgebied wordt zowel het tool als de doelomgeving van het tool beschreven, voor welke hardware en netwerken het tool geschikt is. Opgenomen zijn:

- welke database-managementsystemen (DBMS) kunnen worden gebruikt;
- voor welke besturingsplatformen kan het tool code genereren;
- op welke besturingsplatformen het tool beschikbaar is.

Als voorbeeld hebben we enkele hulpmiddelen in het ISES-raamwerk ondergebracht (ISES CASEBASE, 1993) in de figuren 4 en 5.

Daarnaast kunnen in het model natuurlijk ook methoden en technieken voor systeemontwikkeling worden geclassificeerd, zoals in figuren 6 en 7 is weergegeven.

Tevens is het mogelijk om in dit model de gewenste functionele ondersteuning en technische kenmerken van een te kiezen

tool vast te leggen en te gebruiken als uitgangspunt voor de selectie van een tool. Het selectie- en implementatietraject wordt verderop nader uitgewerkt.

Ontwikkeling en verschijningsvormen van CASE-tools

Ontwikkeling

Terugkijkend op de ontwikkeling van de verschillende CASE-producten zien we dat in het begin van de jaren '80 vooral hulpmiddelen op de markt waren die het tekenen van diagrammen ondersteunden, ter vervanging van de template en het "pen en papier". Hier ging het primair om ondersteuning bij de documentatie. Daarna begonnen hulpmiddelen op te komen waarin enkele van de consistentieregels bij de diagrammen waren opgenomen. Niet alleen kon men een vierkant tekenen, maar men kon het ook als entiteitstype benoemen en de uniekheid ervan laten controleren. Het hulpmiddel biedt dan vooral verificatie-ondersteuning. Als laatste stap in de ontwikkeling komen er hulpmiddelen op de markt die naast documentatie en verificatie ook mogelijkheden bieden om van het ene type diagram over te gaan naar een ander type diagram. Het gaat hierbij om een transformatie van ontwerp-specificaties van een niveau in ons model naar het onder- of naastliggende niveau. Zo kunnen tools tegenwoordig vanuit een logisch gegevensmodel een relationele database-structuur genereren.

Verschijningsvormen

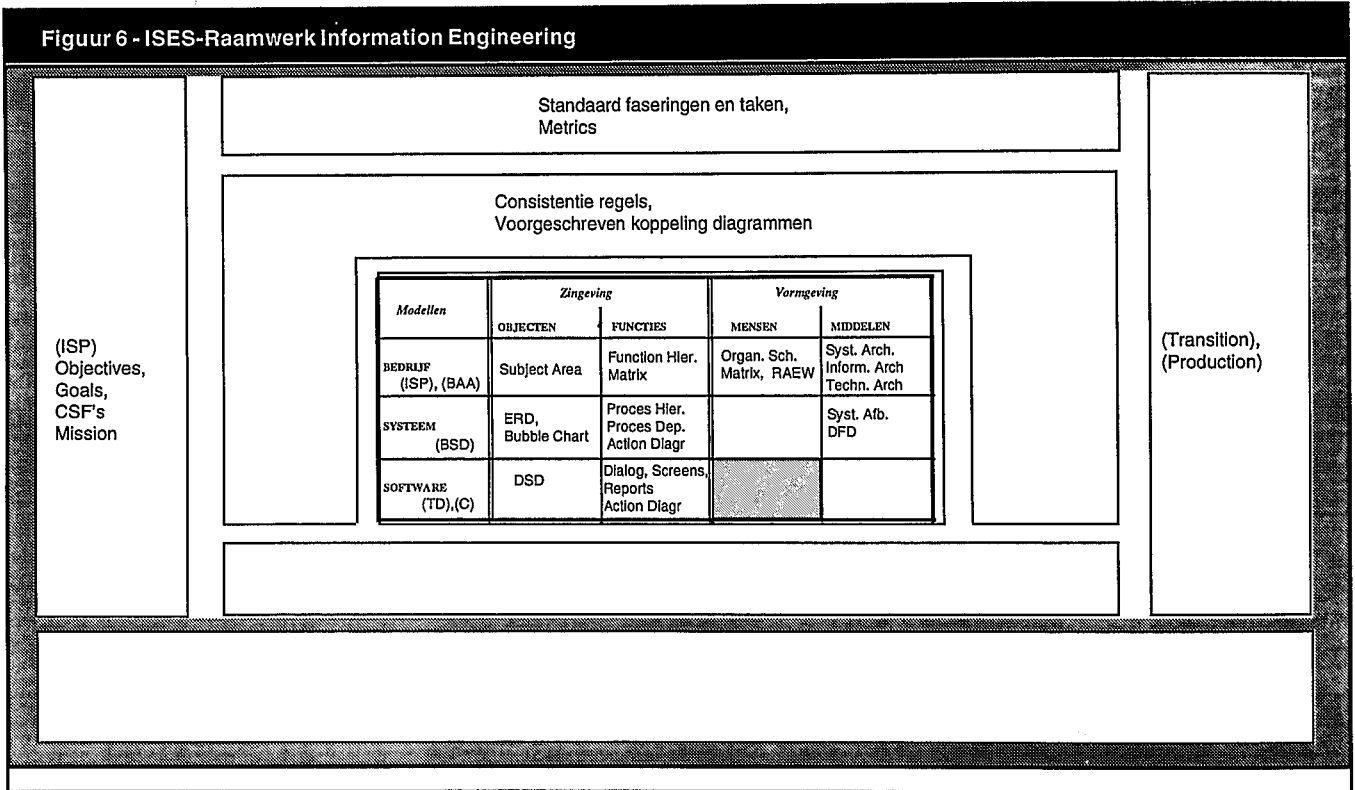
Wanneer we op de markt verkrijgbare producten analyseren dan kunnen wij deze onderverdelen in de volgende categorieën:

- *Upper CASE*: Dit zijn hulpmiddelen die één of meerdere technieken ondersteunen die worden gebruikt voor het specificeren en ontwerpen van informatiesystemen. Deze tools bieden derhalve ondersteuning op bedrijfs- en systeem-niveau uit ons model. Kenmerk van deze tools is het maken van diagrammen, zoals Entity-Relationship Diagrams, Data Flow-Diagrams en Matrices, op PC-werkstations. Er is een enorm aanbod van Upper-CASE-

tools op de markt, sterk variërend in prijs en prestatie. Belangrijkste producten zijn o.a. SDW van Cap Volmac en IEW/ADW van Knowledgeware.

- *Lower CASE*: Producten die tot deze categorie behoren ondersteunen de bouw en het eventuele testen en onderhouden van systemen door middel van bijvoorbeeld codegeneratie, testen, debuggen. Ze ondersteunen derhalve het software-niveau uit ons model. Producten die tot deze categorie behoren zijn o.a. CSP van IBM en TOP*CASE van Comtecno.
- *Integrated CASE (ICASE)*: Gereedschappen die samen de gehele levenscyclus van een informatiesysteem ondersteunen, vanaf de analyse van de specificaties tot en met het genereren van derde generatie-taal (3 GL) of 4 GL-code, testen en eventueel onderhoud. ICASE-producten bevatten een centrale repository, waarin alle systeemspecificaties en overige gegevens integraal worden opgenomen. Voorbeelden van ICASE-producten zijn de IEF van Texas Instruments en Oracle*-CASE.
- *IPSE*: Wanneer een hulpmiddel alle functionaliteit van een ICASE biedt en daarnaast ook nog ondersteuning op het terrein van project management dan spreken we van een IPSE, oftewel Integrated Project Support Environment. Complete IPSE's zijn nog niet op de markt, maar Maestro-II van Softlab komt daar dicht bij in de buurt doordat het een goed Upper-CASE-tool combineert met project management mogelijkheden.
- *Component CASE*: Component CASE bestaat uit een combinatie van meerdere CASE-producten (vaak upper- en lowercase producten), die al dan niet via een centrale repository gegevens uitwisselen. De producten ondersteunen hierdoor een of meerdere fasen van de levenscyclus van een informatiesysteem, maar zijn niet integraal gekoppeld. Veelal is slechts een koppeling op het gebied van de data-modellen aangebracht. Voorbeeld is de Intersolv-productenlijn (Excelerator en APS) en de vele koppelingen tussen SDW en 4GL-hulpmiddelen.

Figuur 6 - ISES-Raamwerk Information Engineering



agro informatica 6(5) / december 1993

■ **Integratieraamwerken/meta-CASE:** Integratieraamwerken bestaan meestal uit een centrale repository, al dan niet met gereedschappen om de gegevens in de repository te beheren en te verspreiden over de verschillende ontwikkelaars, faciliteiten voor project- en/of procesmanagement en voor het beheren van de verschillende eind- en tussenproducten van het ontwikkelproces. Vaak worden front-end en/of back-end producten van andere leve-

ranciers gebruikt om tot een volledige CASE-omgeving te komen. Voorbeelden hiervan zijn o.a. AD/Cycle van IBM, Cohesion van Dec.

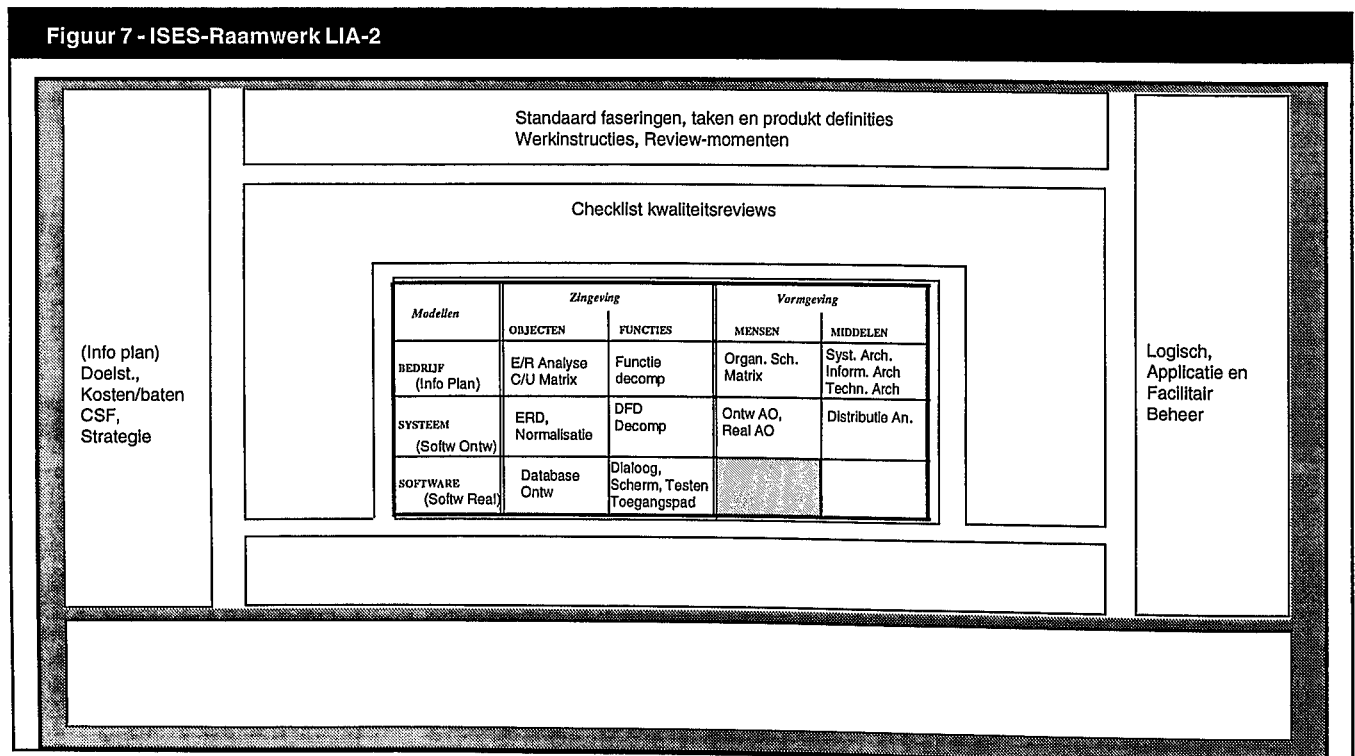
■ **Reverse engineering:** Dit is een categorie hulpmiddelen die ondersteuning bieden bij het analyseren van bestaande code- en database-structuren. Vooral reverse engineering van data-structuren wordt ondersteund en is terug te vinden in de vele Upper-CASE tools. Daarbij gaat het om het inlezen van be-

staande data-structuren in de bij het tool behorende repository, waarna deze gegevens in een (al dan niet conceptueel) datamodel worden weergegeven.

Vormen van integratie

Er zijn verschillende soorten integratie tussen (onderdelen van) tools te onderkennen. Wij onderscheiden de volgende drie soorten (zie figuur 8):

Figuur 7 - ISES-Raamwerk LIA-2



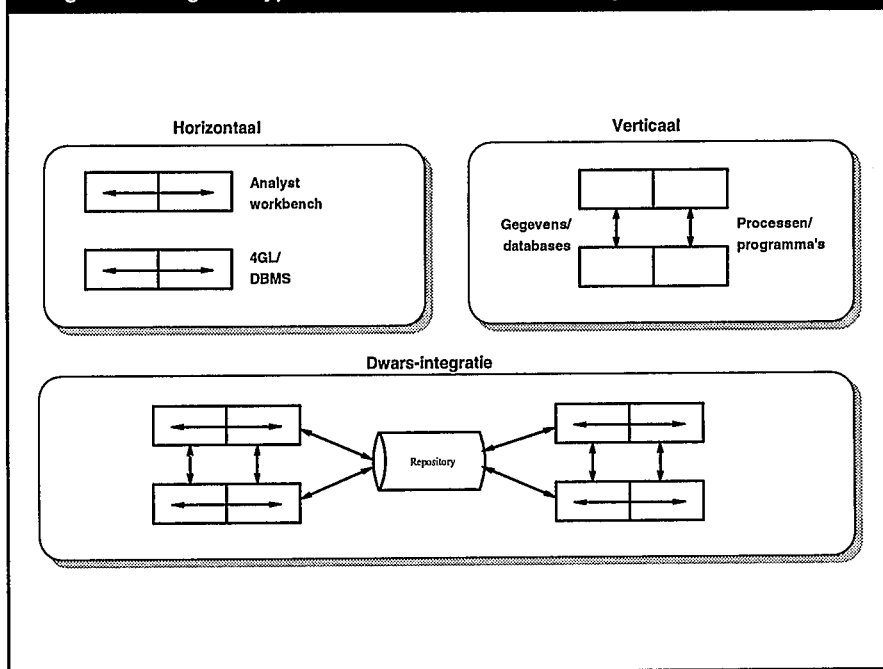
- 1 *Horizontale integratie*: Integratie van technieken binnen een modelleringsniveau (bijvoorbeeld het koppelen van gegevensmodellen aan procesmodellen);
- 2 *Vertikale integratie*: Integratie van technieken over meerdere modelleringsniveau's (bijvoorbeeld op basis van een gegevensmodel genereren van een database structuur);
- 3 *Dwars-integratie*: Integratie van producten van verschillende werkplekken (bijvoorbeeld het op twee werkplekken werken aan het zelfde gegevensmodel, oftewel de mate waarin multi-user-faciliteiten door het tool worden geboden).

Vormen van generatoren

Binnen de lower-CASE producten kunnen we ten aanzien van de generatoren een tweetal typen onderscheiden:

- *Proces Driven-generatoren*: Dit zijn generatoren die als input een proces-specificatie behoeven. Deze generatoren expanderen de specificatie, opgesteld in een formele, procedurele taal, veelal pseudo-code, tot voor compilers leesbare code. Voorbeeld van zo'n code-generator zijn CSP van IBM en APS van Intersolv.
- *Data Driven-generatoren*: Deze generatoren gaan uit van een logisch data-model om alle "standaard" operaties op de entiteiten, zoals invoeren, wijzigen, verwijderen, etc, te genereren. In tegenstelling tot de proces driven-gen-

Figuur 8 - Integratie-typen tussen verschillende CASE-producten



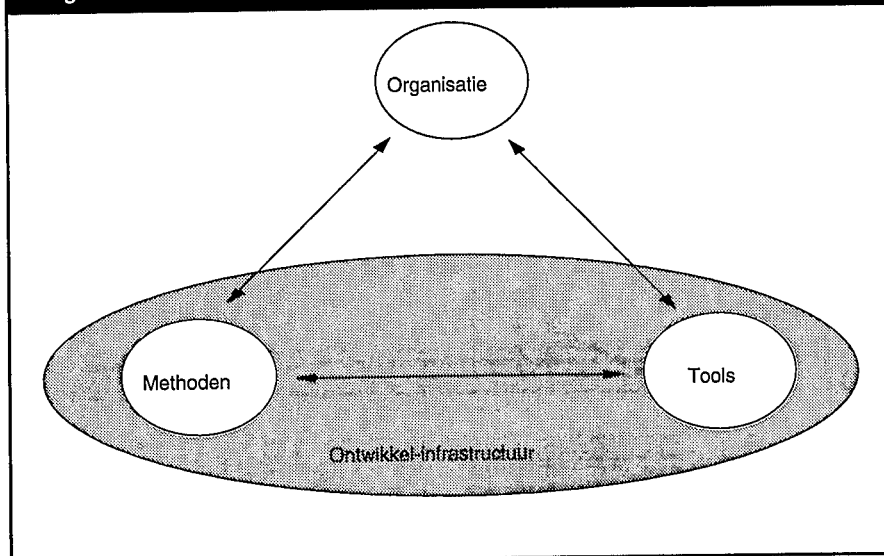
eratoren is de applicatie die gegenereerd wordt voorzien van een standaard user-interface. De benodigde invoer is echter veel minder. Voorbeelden van zo'n applicatie-generator zijn Oracle*Generator en TOP*CASE.

Implementatie van CASE

Naast aandacht voor de te gebruiken methoden en tools is er naar onze mening een derde belangrijke factor die het succes van de invoering bepaalt: de inbedding in de organisatie. Deze organisatie-component wordt expliciet gemaakt vanwege een aantal redenen:

- Veelal is de start van het keuzeproces voor CASE-tools een tegenvallende performance van de ontwikkelorganisatie. Het is dus van belang de organisatie te kennen vooraleer een keuze voor een tool gemaakt wordt;
- Het type systemen dat voor de gebruikersorganisatie moet worden gerealiseerd heeft directe invloed op de keuze van de methoden, technieken en tools. Het is dus zaak de karakteristieken van deze systemen te onderkennen en expliciet te maken;
- De invoering van CASE is pas geslaagd te noemen als de doelgroep ook daadwerkelijk de nieuwe hulpmiddelen gebruikt (de inbedding in de organisatie).

Figuur 9 - De drie dimensies van het implementeren van CASE



Met de factor 'organisatie' wordt dus beschreven wat de karakteristieken zijn van de organisatie die de methoden en hulpmiddelen zal gaan gebruiken (de systeemontwikkelorganisatie) maar ook ten behoeve waarvan de systemen ontwikkeld gaan worden (de gebruikersorganisatie).

Samenvattend kunnen we stellen dat aan veranderingen als gevolg van het implementeren van CASE-technologie een drietal dimensies kunnen worden onderkend. Deze drie dimensies zijn weergegeven in figuur 9.

Doel van een implementatie van CASE-technologie is niet het realiseren van een adequate methode en toolset, maar het structureel verbeteren van de wijze van systeemontwikkeling. Een effectieve en efficiënte ontwikkelinfrastructuur kan immers geen doel op zich zijn, maar moet ten doel staan aan de ontwikkel-organisatie.

Om een éénduidige beschrijving van de ontwikkelomgeving te verkrijgen worden de inhoud van de bovengenoemde dimensies en hun onderlinge relaties schematisch weergegeven. Dit schema noemen we het "ontwikkelpateau". In figuur 10 is het ontwikkelplateau weergegeven.

Naast het model van de ontwikkelinfrastructuur uit de paragraaf "Wat is CASE?" is dus in het ontwikkelplateau een categorie organisatorische aspecten opgenomen, die een groot aantal onderdelen van de dimensie organisatie beschrijft (zoals structuur van de ontwikkelingsorganisatie, type applicaties dat gebouwd moet worden, besteding van beschikbare mensen en geld en technische en emotionele performance van de ontwikkelomgeving). Een aantal technische aspecten van de organisatie, zoals de hardware en software, die aanwezig

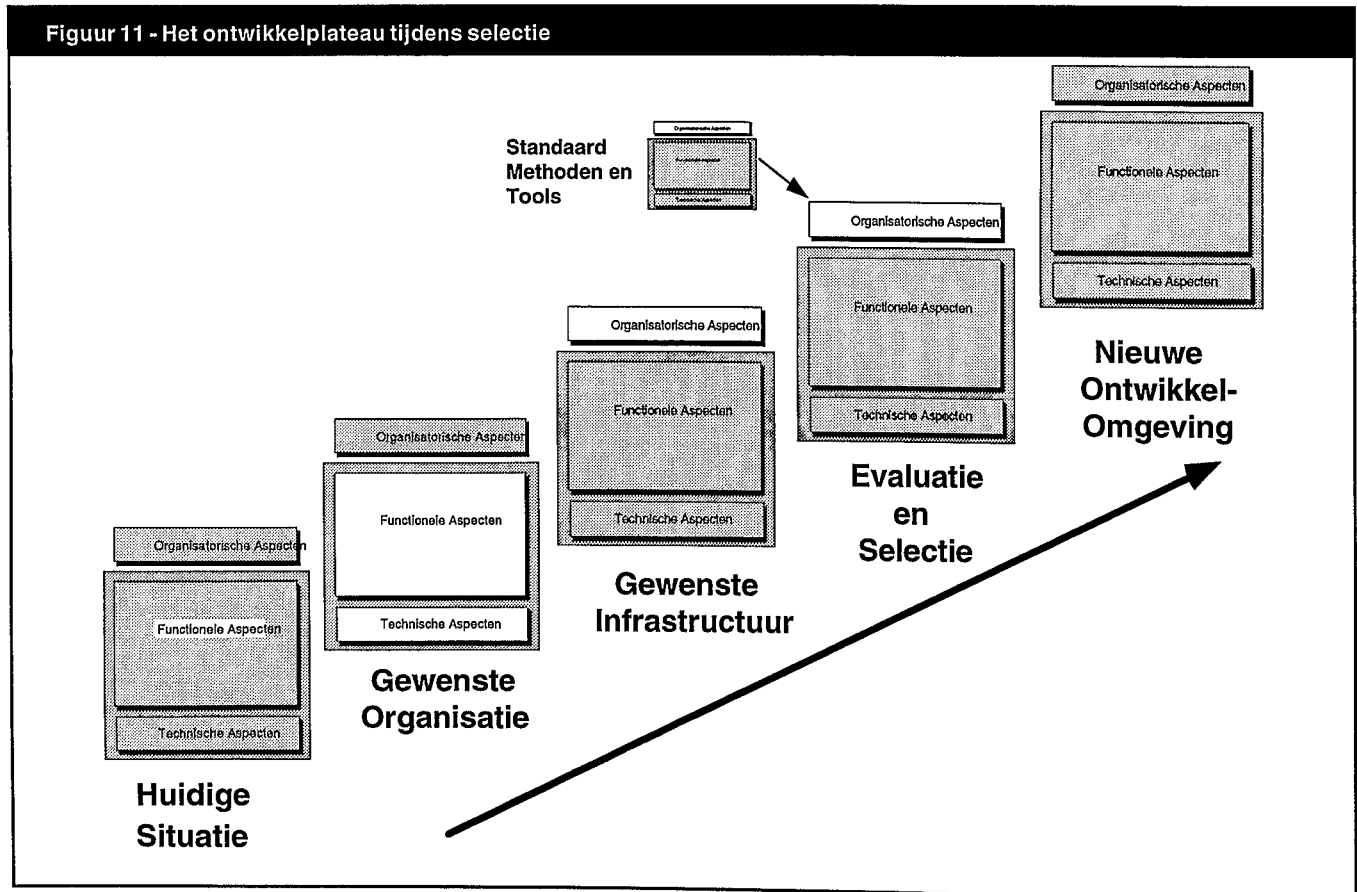
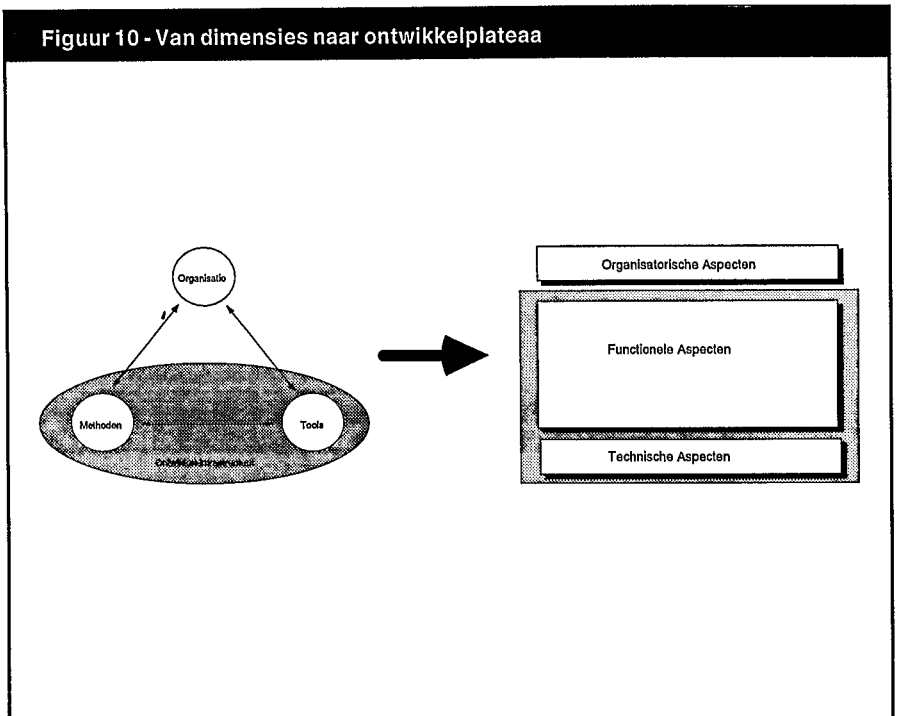
is, kan eveneens in het blok voor de technische aspecten beschreven worden (Harkel & Dortmans, 1993).

Het selectie-traject

Het ontwikkelplateau wordt gebruikt voor het beschrijven van de organisatie en haar ontwikkelinfrastructuur op een bepaald tijdstip. Deze beschrijving vormt het uitgangspunt bij het onderken-

nen en invoeren van een nieuwe ontwikkelomgeving. Hierbij worden een aantal selectiestappen gebruikt die zijn weergegeven in figuur 11, te weten:

- *Evaluatie huidige ontwikkel-organisatie*; De huidige aanwezige methoden en tools worden in kaart gebracht alsmede de huidige ontwikkel-organisatie;

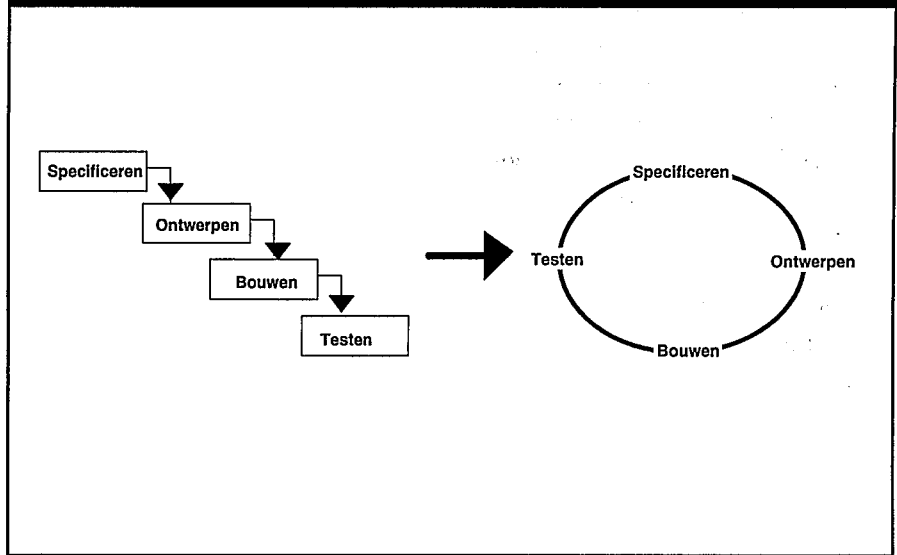


- *Opstellen scenario voor gewenste organisatie*; Er wordt een scenario van de gewenste organisatorische situatie opgesteld, uitgaande van de strategie van de onderneming;
- *Bepaling gewenste infrastructuur*; De gewenste functionele en technische kenmerken van de nieuwe ontwikkelomgeving worden bepaald;
- *Evaluatie en selectie van methode en tools*; Op basis van de op de markt verkrijgbare methoden en tools vindt een selectie- en evaluatieproces plaats. Hierbij worden de gewenste invulling van het ontwikkelplateau vergeleken met de invulling die methoden, technieken en tools kunnen bieden;
- *Definitie nieuwe ontwikkelomgeving*. De te realiseren ontwikkelomgeving wordt in deze stap in z'n totaliteit gedefinieerd, waarbij het denkbaar is dat de gewenste situatie niet gerealiseerd kan worden met de beschikbare methoden en hulpmiddelen, en er dus in ambitieniveau een stapje terug gedaan moet worden.

Het invoeringstraject

Voorwaarde voor een succesvolle implementatie van nieuwe technologieën is de acceptatie door de gebruikers van deze technologie. Dat betekent dat de in de vorige paragraaf gedefiniëerde ontwikkelomgeving niet alleen moet worden "aangekocht", maar dat deze moet worden ingebed in de organisatie. De mate van succes van een verandering, ofwel de kwaliteit

Figuur 12 - Een nieuwe manier van ontwikkelen



van de verbetering, is afhankelijk van twee factoren (Bemelmans, 1987):

- *De kwaliteit van het produkt*. Hiermee wordt in ons model de inhoud van de verandering bedoeld. Dit wordt weergegeven in het ontwikkelplateau zoals dat hiervoor beschreven is.
- *De kwaliteit van de implementatie van het produkt*. Hiermee wordt de manier bedoeld, waarop het produkt als verandering wordt geïmplementeerd in de organisatie.

Dit betekent dat de invoering een beheerst proces moet zijn, waarbij de juiste verwachtingen op de juiste momenten worden waargemaakt. De praktijk van implementatie leert dat men hiervoor het best eerst de nieuwe werkwijze van de automatiseerders test op een bestaand, niet vitaal

geautomatiseerd project, dat met de nieuwe ontwikkelomgeving wordt nagebouwd. Dit project, met als primaire doelstelling het leren kennen van de ontwikkelomgeving, wordt door ons het "pathfinder project" genoemd. Vervolgens zal de nieuwe werkwijze moeten worden gedemonstreerd in de bestaande organisatie door middel van een voorbeeldproject. Doelstelling van dit project is de organisatie te tonen de nieuwe ontwikkelomgeving ook daadwerkelijk beter is. Dit project noemen we het "eyecatcher"-project. Na de eyecatcher volgt de fase van verspreiding van de nieuwe ontwikkelomgeving in de organisatie. Ook dit dient beheerst, door middel van vooraf gedefiniëerde ontwikkelplateaus, te gebeuren.

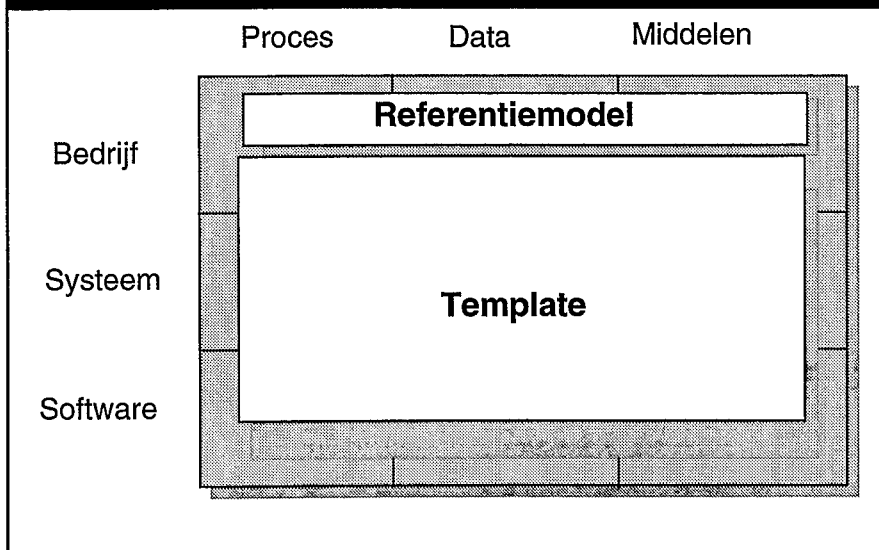
Afsluiting

CASE wordt wel eens omschreven als het automatiseren van het gehele proces van systeemontwikkeling. Echter, het verschilt van produkt tot produkt welke bijdrage wordt geleverd aan het ontwikkelproces. Echt gebruik van CASE-technologie heeft belangrijke gevolgen voor de fasering van het ontwikkelproces.

Bijvoorbeeld:

- Gebruik van ICASE-tools en generatoren maakt het mogelijk om via een iteratieve methode te gaan ontwikkelen. Vanuit de specificaties uit de analyse wordt het systeem ontworpen op basis waarvan door middel van een genera-

Figuur 13 - Templates



tor de applicatie wordt gegenereerd. Deze code kan dan weer worden getest om te controleren of de specificaties juist en voldoende zijn geweest. Vervolgens kan op basis van de gewijzigde specificaties het systeem opnieuw worden gegenereerd. Deze nieuwe werkwijze is weergegeven in figuur 12.

- Een ander nieuw fenomeen dat optreedt door het gebruik van ICASE-tools zijn de "templates". Dit zijn specificaties van complete applicatie-pakketten in de vorm van een vulling van een ICASE-repository op bedrijfs-, systeem- en software-niveau. Aanpassingen kunnen op de specificaties aangebracht worden waarna opnieuw gegenereerd kan worden. Zo ontstaan "tailormade standaardpakketten". Templates zijn wezenlijk verschillend van de reeds lang in de landbouw wereld bekende referentie-informatiemodellen, waarin alleen een model op bedrijfs- of systeemniveau aanwezig is (zie figuur 13).

- Invoering van CASE heeft gevolgen voor de personele samenstelling van de ontwikkelorganisatie. Methoden zullen meer worden afgedwongen, waardoor creativiteit in het produkt gestopt kan worden in plaats van in het proces. Programmeurs zullen steeds meer de functie krijgen van "system engineers" waardoor ze, meer dan voorheen, met eindgebruikers moeten gaan werken.

CASE is automatisering van de automatisering. Als CASE dus in chaos wordt ingevoerd, krijgt men last van het GIGO-effect. Garbage In, Garbage Out. Wanneer men chaos automatiseert, maakt men er in korte tijd, volgens de belofte van de leveranciers, een nog grotere hoeveelheid chaos bij. Echter, indien CASE-technologie weloverwogen en beheerst wordt geïmplementeerd zullen de problemen geschetst in de paragraaf over het CASE-dilemma, volgens ons best eens verminderd kunnen worden.

Literatuur:

- Laagland, P.T.M. & S. Argelo (1992)
De opdrachtgever werkt mee, Praktijkgids
De controller & informatie management
- Harkel, E. ten & E. Dortmans (1993)
Het implementeren van CASE; de
organisatie als vergeten dimensie,
Informatie
- Bemelmans, T.M.A. (1987)
Bestuurlijke Informatiesystemen en
automatisering, Stenfert Kroese, Leiden.
- Kusters, R.J. & G.M. Wijers (1992)
Het gebruik van CASE-Tools in
Nederland, Kluwer Bedrijfswetenschappen
- ISES CASEBASE (1993)
Een database met een classificatie van 70
verschillende CASE-Tools, ISES
International
- Database Gids '92-'93
Database Magazine, december 1992