



KUNSTMATIGE INTELLIGENTIE

Voorbeeld van een toepassing van kennistechnologie: een expertsysteem dat bij de reparatiedeling van een fabriek voor consumentenelektronica ondersteunt bij het herstellen van afgekeurde apparatuur.

bleemoplosgedrag van een ervaren reparateur en ondersteunt, onafhankelijk van een specifiek elektronisch schema, de reparateur bij het lokaliseren van defecten. Het kan om informatie vragen en tests adviseren voor het toetsen van veronderstellingen. Door middel van een uitlegfaciliteit kan de gebruiker de toegepaste redeneringen volgen en aanleren. Het systeem registreert gevonden storingen en stelt de oplosstrategie bij. Statistische informatie over de voorgekomen defecten wordt teruggekoppeld naar de ontwerp- en fabricageafdelingen, wat de kwaliteit van de productie als geheel positief beïnvloedt.

KENNISREPRESENTATIE

Ontwerp en functionaliteit van kennisystemen hebben een aantal specifieke kenmerken die worden uitgedrukt door onderscheid te maken tussen *kennistechnologie (KT)* en *informatietechnologie (IT)*. Het verschil met een klassiek informatiesysteem is dat het proces van probleemoplossen niet gemakkelijk kan worden beschreven als een rechtlijnige, eenduidige procedure. Daarvoor is het proces te dynamisch, te complex en vormt het te zeer een geheel om programmastromen met datatypen en beslissingsmomenten te kunnen bepalen.

Dit gegeven heeft lange tijd de ontwikkeling van bruikbare expertsystemen gefrustreerd. Door de ontwikkeling van nieuwe technieken voor kennisrepresentatie, zijn er echter tegenwoordig efficiëntere methoden beschikbaar, die inmiddels hun toepasbaarheid hebben bewezen. De softwarestructuur van een expertsysteem is hierbij niet zozeer een gerichte graaf met vertakkingen maar eerder een bidirectioneel netwerk met talloze in- en uitvoerpunten. Het gaat hier, voor de duidelijkheid, om re-

Beroep kennistechnoloog

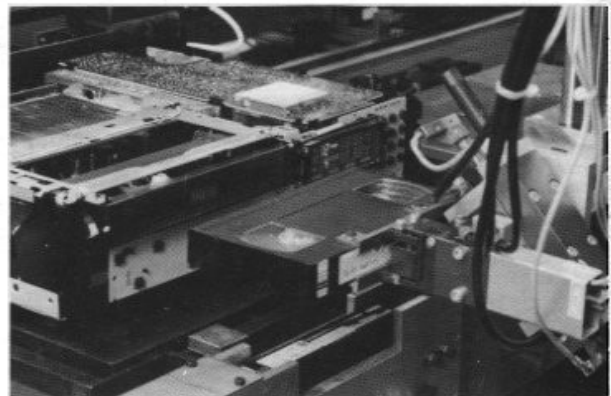
Cursus SKE vormt introductie in jong vakgebied

Kennissystemen zijn in de eerste plaats bedoeld om oplossingen van problemen te zoeken en de resultaten daarvan te tonen. Daarmee wijken ze fundamenteel af van de traditionele informatiesystemen, die uitsluitend zijn bestemd om procedures te implementeren. Een context van algoritmen en datastructuren is dan ook niet erg geschikt voor de verwerking van expertise. Dit blijkt veel beter te beschrijven binnen een concept van oplosstrategieën en kennismodellen. Inmiddels is het mogelijk om dergelijke strategieën en modellen te expliciteren, althans sommige daarvan, en daarmee is een nieuwe deeldiscipline binnen de informatica ontstaan: de kennistechnologie. Het is dit vakgebied waarop het bedrijf Bolesian zich voornamelijk heeft toegelegd. Een belangrijk deel van de activiteiten wordt gevormd door het organiseren van cursussen. Door middel van een stelsel van op elkaar aansluitende modules kunnen deelnemers worden opgeleid tot kennistechnoloog. De eerste module, SKE, wordt in dit artikel beschreven. Deze cursus behandelt de methode 'Structured Knowledge Engineering', een uitgebreide en gedetailleerde leidraad voor het ontwerpen van kennissystemen. Uiteindelijk is het, in een notedop, de taak van de kennistechnoloog om de expertise van de deskundige te 'vertalen' naar geschikte representaties, zonder inhoudelijk iets te veranderen. De expert is koning in de kennistechnologie.

Kennissystemen staan nog aan het begin van hun ontwikkeling maar worden toch meer en meer daadwerkelijk in praktijk situaties geïmplementeerd en toegepast. Een voorbeeld daarvan, ter verduidelijking maar zonder man en paard te noemen, is het geval dat speelt bij een concern dat onder meer op elektronica gebaseerde consumentenartikelen produceert. Apparaten die bij de eindcontrole worden afgekeurd, gaan naar een reparatiedeling. Hier bestaat echter een tekort aan ervaren reparateurs. Het produkt is complex en wordt ook regelmatig qua ontwerp iets gewijzigd. De ervaren mensen hebben hier veel minder last van dan de nieuwelingen. Onervaren reparateurs lopen vaker vast en hebben bijna altijd meer tijd nodig voor een reparatie.

Het binnen deze onderneming gerealiseerd kennissysteem simuleert nu het pro-

Bij de eindcontrole wordt alleen een fout geconstateerd, het is aan de reparatiedeling om de oorzaak te vinden. Hiertoe simuleert het kennis-systeem het probleemoplosgedrag van een ervaren reparateur.



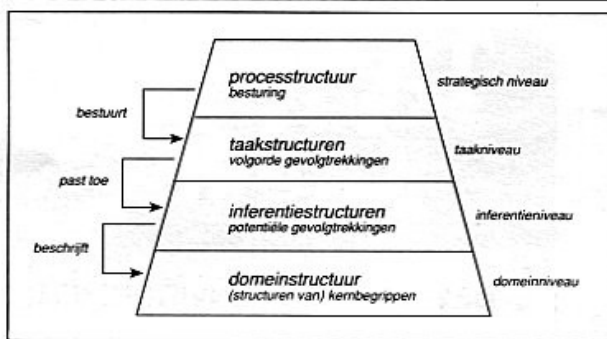


Fig. 1. Het vierlagen-model van KADS.

▷ presentaties op hoger niveau. Op machinenniveau is er natuurlijk geen verschil tussen op kennis gebaseerde programmatuur en andere soorten software.

LAGENMODEL

Aan de Universiteit van Amsterdam is onder de naam *Knowledge Acquisition and Documentation Structuring (KADS)* een techniek ontwikkeld die is gebaseerd op een vierlagenmodel (fig. 1). Dit concept berust geheel op het werken met strategieën voor oplossingen en met kennismodellen in plaats van in procedures ondergebrachte algoritmen en datastructuren.

Met KADS als basis heeft het bedrijf Bole-sian een meer omvattende ontwerpmethodode voor expertsystemen ontwikkeld. Hierin zijn ook functies met betrekking tot het organisatorische traject opgenomen. Dat is gebeurd vanuit het uitgangspunt dat ook moet worden voorzien in het beheren van de projectvoortgang, vanaf het eerste contact met de opdrachtgever tot en met het onderhoud van het kennisstelsel (zie figuur 2).

De methode, *Structured Knowledge Engineering (SKE)* genaamd, is voor derden

beschikbaar door middel van een drie-daagse cursus. Hoewel deze een op zichzelf staand geheel vormt, maakt hij deel uit van een door het bedrijf opgezette modulaire opleiding tot *kennistechnoloog*. Het totale pakket bestaat uit zeven modules, waarvan de eerste de in dit artikel beschreven cursus SKE is en het laatste blok bestaat uit een praktijkopdracht.

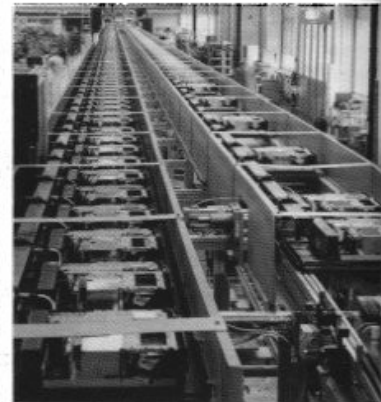
ELEMENTAIR

KADS is mede ontwikkeld als deelproject in het kader van het Europese onderzoeksprogramma *Esprit*. De methode komt uit de hoek van de cognitieve psychologie. J.A. Breuker en B.J. Wielinga van de Universiteit van Amsterdam zijn de grondleggers. Uitgangspunt is dat het proces van probleemoplossen compositioneel is, dat wil zeggen dat het kan worden opgevat als een samenwerking van elementaire taken. Een taak is een redenering die een mogelijke oplossing geeft, waarbij de (deel)oplossing het doel is van de taak.

Volgens KADS werkt een taak altijd op een bepaald *afleidingsmodel*, een zogeheten *inferentiestructuur*. Er zijn gevolgtrekkingsstructuren ontwikkeld voor een aantal algemene probleemttypen, zoals bewaking,

klassificatie, geschiktheidsbeoordeling, remedie en anderen. Voor *diagnose*, gedefinieerd als het lokaliseren van een probleem of storing, zijn twee algemene inferentiestructuren te onderscheiden. Deze zijn te gekarakteriseerd als *systematisch* en *heuristisch*.

Vuistregels die op het eerste gezicht vaak willekeurig lijken maar die bijna altijd een systematische verklaring hebben, vallen in de categorie heuristische gevolgtrekkingen. Op grond van ervaring wordt het pad



Er zijn verschillende soorten inferentiestructuren voor uiteenlopende probleemttypen, zoals de klassificatie van een probleem of fout.

van gevolgtrekkingen op een aantal punten als het ware afgesneden. In het genoemde voorbeeld van de reparatieafdeling is er, voorafgaand aan de reparatie zelf (remedie), zeker sprake van diagnose.

SYSTEMATISCH

Voortbordurend op het thema reparatie, zou de systematische variant als volgt kunnen verlopen:

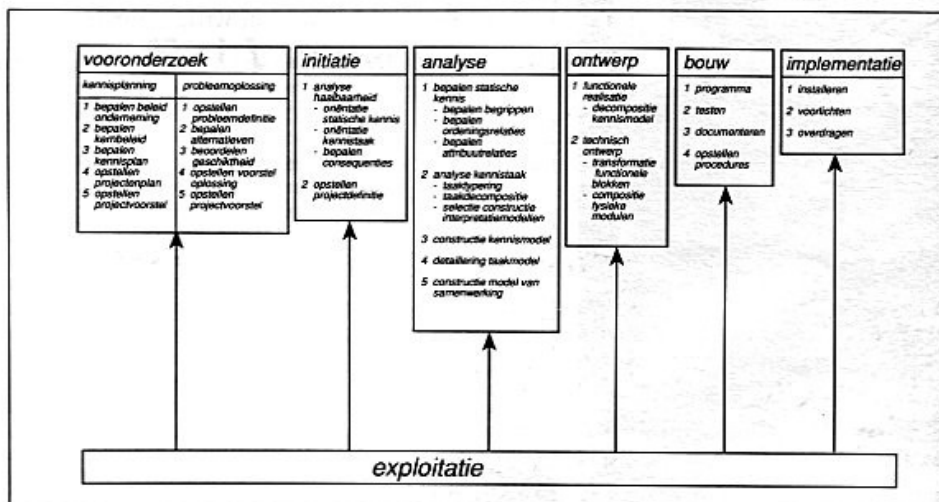
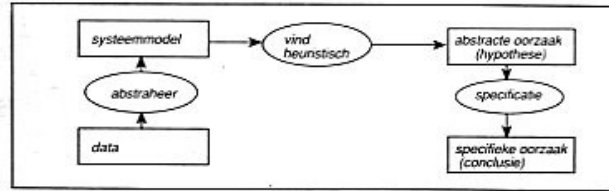


Fig. 2. De ontwerpmethodode SKE omvat niet alleen de kennisverwerving maar ook de beheersing van het hele ontwerptraject: de verschillende stadia en de diverse aspecten daarvan.

Een reparateur die 'volgens het boekje' werkt, zal naar aanleiding van de klacht en naar wat hij of zij weet van het totale apparaat, direct een beeld hebben welke secties in elk geval wel functioneren en zich beperken tot de overige onderdelen van het produkt. De storing is in elk geval in een, of wellicht meer, van die modules of compo-

Fig. 5. Een andere mogelijkheid is heuristische diagnose, waarvan het afleidingsmodel er zo uit kan zien.



Een reparateur die systematisch werkt, zal direct een aantal foutbronnen kunnen uitsluiten.

STRUCTUREN

In figuur 5 staat het inferentiemodel afgebeeld en de figuren 6 en 7 geven de twee taakstructuren. Te zien is dat deze procesvormen niet eenduidig zijn bepaald door het afleidingschema maar dat ze in het algemeen altijd op meer manieren kunnen worden doorlopen. Op basis van de diagnose volgt een remedie.

nenten te lokaliseren en daarmee dus in principe gegeven. Het probleem is nu om te bepalen welke van de mogelijkheden de juiste is.

Vervolgens wordt als tweede stap een van die mogelijke foutoorzaken als hypothese genomen. Op basis daarvan selecteert de reparateur een waarneming of test waarvan de uitkomsten kunnen worden vergeleken met een norm. Als de meetwaarden niet afwijken van deze norm, gebeurt hetzelfde bij een volgende veronderstelling en bijbehorende test. Als de waarde wel afwijkt, is het probleem in een kleinere sectie gelokaliseerd en volgt een nieuwe hypothese, net zo lang tot er geen mogelijkheden meer zijn.

Het systematisch doormeten van een schakeling (een geselecteerde module) is een voorbeeld van deze taakstructuur. In de ab-

stracte notatie van figuur 3 zijn de parameters met '-' ervoor uitvoer (de doelen) en die met '+' ervoor invoer. De taak werkt op het afleidingsmodel (de inferentiestructuur) dat in figuur 4 is weergegeven.

```

HEURISTISCHE - DIAGNOSE(voorwaarts - redeneren)
begin
vind( - specifieke - oorzaak)
begin
haak( - data);
abstraher(+ data, - abstracte - data);
vind - heuristisch( + abstracte - data, - abstracte - oorzaak);
specificeer( + abstracte - oorzaak, - specifieke - oorzaak)
end
end.
    
```

Fig. 6. Taakstructuur voor heuristische diagnose bij voorwaarts redeneren.

Fig. 3. Interpretatiemodel voor systematische diagnose.

```

SYSTEMATISCHE - DIAGNOSE(causaal traceren en lokalisatie)
begin
vind( - oorzaak)
begin
selecte(+ klacht, + systeemmodel, - sub - systeemmodel);
WHILE(geen - conclusie) DO
begin
ontleed(+ sub - systeemmodel, - hypothese, - conclusie);
WHILE (aantal - hypothesen > 1) DO
begin
selecte(+ universum - van - waarnemingen,
+ hypothese, - variabele - waarden);
specificeer(+ hypothese, - norm);
vergelijk(+ variabele - waarden, + norm, - verschil)
end
end
end
end.
    
```

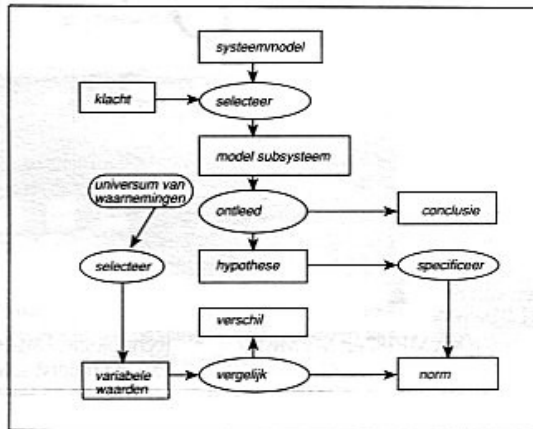


Fig. 4. Mogelijke inferentiestructuur voor systematische diagnose.

De reparateur hoeft echter niet te beginnen met een systematische diagnose. Hij kan ook meteen naar een mogelijke oorzaak springen, nagaan welk algemeen effect dat zou hebben, het effect met een groep waarnemingen in verband brengen en dan kijken of een ervan ook optreedt. Dit is een heuristische diagnose met terugredentie. De andere mogelijkheid is dat de reparateur meteen iets opvalt dat hij op grond van zijn ervaring uit vorige gevallen met een oorzaak kan verbinden. Dit heet voorwaarts redeneren.

Fig. 7. Als figuur 6 maar nu voor terugredeneren.

```

HEURISTISCHE - DIAGNOSE(terugwaarts - redeneren)
begin
test( + specifieke - oorzaak)
begin
specificeer( - abstracte - oorzaak, + specifieke - oorzaak)
vind - heuristisch( - abstracte - data, + abstracte - oorzaak);
abstraher( - data, + abstracte - data);
haak( + data);
end
end.
    
```


▷ De grondgedachte van KADS is nu dat het hele reparatieproces kan worden beschreven als een stelsel van samenwerkende taken. Bovenop dit geheel komt dus een overkoepelend niveau met de strategie, ook wel *processtructuur* genoemd. Hierin worden in dit geval de verbanden tussen de taken voor de diagnose en de remedie vastgelegd.

De afleidingsmodellen die een rol spelen, moeten natuurlijk op hun beurt worden verbonden met de specifieke vakkennis. Op dit *domeinniveau* moet bij de reparatie waarschijnlijk worden gedacht aan functionele subsystemen van het apparaat, aan de opbouw in borden en kaarten en aan de diverse chips met hun verbindingen en dergelijke. Het gaat echter ook om de *abstracte oorzaken (hypothesen)* en tests, met mogelijke waarden en normen en om de heuristische regels die de ervaren reparateur uitprobeert. Het gaat ook om de connectie tussen een klacht en de eerste selectie van het probleemgebied.

INTERACTIE

Het gehele proces van probleemoplossen zoals hier boven beschreven, past nu in het vierlagenmodel uit figuur 1. Er is echter nog een belangrijk aspect dat hierbij speelt. Het idee van samenwerkende taken veronderstelt een hoge mate van interactie waarin ook de menselijke gebruiker een rol heeft. Elke procedure krijgt dan ook precies één 'eigenaar' toegewezen die met andere actoren kan communiceren in een proces van geven en nemen.

Als er bijvoorbeeld gegevens nodig zijn, kan dat in de abstracte syntaxis worden weergegeven als *haal(-data)*. Maar wie is de eigenaar van die taak en wie neemt het initiatief? De eigenaar zal hier waarschijnlijk het proces zijn dat de data nodig heeft maar de bron van de gegevens kan een ander proces zijn of de gebruiker ervan.

Het maakt ook verschil uit of er een vraag naar informatie uitgaat van het systeem dan wel of er een aanbod komt op commando van de gebruiker. Dit betekent dat wat een taak is niet alleen wordt bepaald vanuit de probleemoplossing maar ook door de communicatieaspecten. Als er gegevens van twee verschillende actoren kunnen komen, moeten er ook twee taken worden gedefinieerd.

De communicatieve aspecten vallen in SKE onder de noemer *modaliteit*. Deze aspecten zijn zo belangrijk dat er in de analyse fase naast het kennismodel ook een model van samenwerking wordt geconstrueerd.

Een derde aspect wordt gevormd door externe beperkingen en voorwaarden, bijvoorbeeld dat het uiteindelijke systeem onder DOS zal werken, een natuurlijke taal in-



Een voorbeeld van synthese is als een arts te maken krijgt met een nieuwe, nog onbekende aandoening.

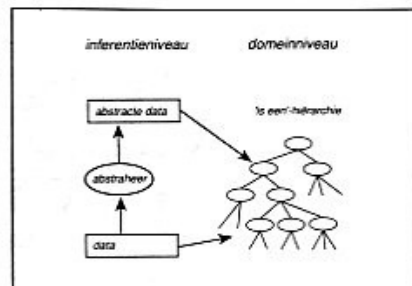
terface zal bezitten en dergelijke. Dit zijn de *eisen*. Probleem oplossen, modaliteit en eisen sturen gezamenlijk het ontwerpen binnen 'Structured Knowledge Engineering'.

DIAGNOSE

Er wordt onderscheid gemaakt tussen drie hoofdtypen van problemen. Bij een *analysetype*, zoals de bepaling van storingen, is de oplossing eigenlijk altijd gegeven. Het probleem schuilt hier in het maken van een keuze uit een eindig, hoewel misschien zeer groot, aantal mogelijke oplossingen. De tegenhanger hiervan is de *synthese*. Bij dit type is het niet mogelijk om een remedie te kiezen maar moet deze in zijn geheel worden gegenereerd omdat hij niet vooraf bekend is. Een arts die een diagnose stelt, zal meestal een bekende ziekte vaststellen (analyse) maar het is altijd mogelijk dat hij met een nieuwe, nog onbekende ziekte heeft te maken. In het laatste geval is het probleemtype synthetisch.

Een tussenvorm is *modificatie*. Hierbij kan wel een remedie worden geselecteerd maar zal deze niet altijd volledig voldoen zodat er aanpassingen nodig kunnen zijn. Per definitie is het meest bekend over analyse, maar binnen SKE bestaat een biblio-

Fig. 8. De kennisbron abstraheer geeft de hiërarchie van metaklassen aan. Hoe de boom feitelijk wordt ingevuld, behoort tot de domeinstructuur.



theek met standaard inferentiestructuren voor meer dan tien subtypen, waaronder verschillende synthetische.

KENNISBRON

Een schema voor een afleidingsmodel bestaat altijd uit blokken en ovalen die respectievelijk *metaklassen* en *kennisbronnen* voorstellen. Een metaklasse is een type gegevens, gedefinieerd in de algemene termen die bij het probleemtype passen. Een *kennisbron* is een relatie tussen metaklassen die het logische verband aangeeft. De kennisbron *abstractie* uit figuur 8 geeft bijvoorbeeld aan dat de abstracte gegevens zich in een hiërarchische boom op een hoger niveau bevinden dan de data.

De invulling van de boom, waarbij wordt aangegeven welke technische begrippen die betreffende relatie bezitten, behoort tot de domeinstructuur. Deze bestaat in het algemeen uit een lexicon van trefwoorden die geordend zijn in classificatiebomen, tijdvolgorderelaties, causale relaties en subject/predikaat-relaties maar er kunnen ook kwantitatieve gegevens in zitten. De pijlen in een inferentiestructuur geven alleen de richting van de relatie aan en niet de richting van de gevolgtrekking.

Wat de richting van gevolgtrekking is, wordt bepaald in de taak die van de afleidingsstructuur gebruik maakt (zie figuur 6 en 7). De ontledingskennisbron uit figuur 4 zal bijvoorbeeld als een hiërarchische boom kunnen worden geïnterpreteerd. Hier wordt uit het geselecteerde subsysteemmodel — waarin de storing moet zijn te lokaliseren — een hypothese genomen. Op afleidingsniveau houdt dit in dat de taak een hiërarchische boom van het subsysteem afzoekt, waarbij hij van boven naar beneden niveau voor niveau in de breedte afwerkt. Een bevestigde hypothese kan dan leiden tot een nadere abstracte oorzaak die zich lager in de domeinboom bevindt, net zo lang totdat een *specifieke oorzaak*, de *conclusie*, is bereikt. Dit hoeft echter niet altijd zo te gaan en de taakstructuur uit figuur 3 is zeker niet de enig mogelijke.

AANPASSING

Het kan voorkomen dat het subsysteemmodel niet correct is gekozen, zodat een nieuwe selectie nodig is. Dit gebeurt dan op grond van aanvullende informatie van de gebruiker. Een heuristische gevolgtrekking zal misschien slechts een andere module kunnen aanwijzen waarin misschien wel de oorzaak van het probleem schuilt. Dat maakt dat er nog een systematische test- en hypothese-taakstructuur moet komen. Deze zou vervolgens maar een gedeelte van de inferentiestructuur hoeven te gebruiken.

Aanpassing van die afleidingsmodellen is

▷ altijd mogelijk en de kennistechnoloog moet vaststellen volgens welke patronen de redeneringen van de deskundige verlopen en in hoeverre die bij standaard inferentiestructuren passen. Hij moet ook proberen vast te leggen of en, zo ja, hoe de deskundige heuristische en systematische diagnose combineert en of er sprake is van een te beredeneren strategie.

REMEDIE

Hierna volgt er in principe een remedie. Aangezien het meestal mogelijk is dat er ook daarna nog steeds klachten zijn of dat nieuwe storingen optreden, moet er een terugkoppeling zijn naar de diagnosefase. Op dit procesniveau bestaat niet meer dan één structuur want in deze laag wordt de hele probleemoplossing vastgelegd. Ook op domeinniveau wordt geen nader onderscheid gemaakt. Domeinkennis en de opbouw daarvan zijn algemeen toegankelijk. Het is duidelijk dat de modaliteit, het communicatieaspect tussen taken onderling en de gebruiker, parallel loopt aan de analyse van het probleemoplossen. In de analysefase van SKE wordt dan ook, zoals gezegd, naast een kennismodel en een gedetailleerde taakstructurering tevens een model van samenwerking geconstrueerd. Behalve de actoren, de richting van de overdracht en de initiatiefnemers, komt hierin ook te staan welke informatie nodig is voor een succesvol verloop van de communicatie. In het uiteindelijke expertsysteem worden zo onder andere de dialogen bestuurd. Ook het samenwerkingsmodel kijkt naar de inferentiestructuren maar wederom niet alsof dit de stroomschema's zijn voor de programmaverwerking. Het is heel goed denkbaar dat bij een twijfelachtige test het inferentieproces de gebruiker alleen informatie uit de metaklasse *klacht* oplevert, ter bevestiging dan wel ontkenning van de hypothese.

ONTWERPFASEN

Voordat een kennisstelsel kan worden geïmplementeerd, moeten verschillende ontwerpstadia worden uitgevoerd (zie figuur 2). De uiteindelijke implementatie gebeurt meestal in een ontwikkelomgeving en met een ondersteunende programmeerschil. Voorbeelden daarvan zijn ADS (Aion Development System) en KBMS (Knowledge Based Management System). Ook IBM is actief op deze markt met TIRS (The Integrated Reasoning Shell). Verder zijn de programmeertalen Prolog en Lisp, respectievelijk gebaseerd op predikatenlogica en symboolverwerking, behoorlijk populair als ontwikkelomgeving. Allereerst moet de kennistechnoloog echter de expertise zodanig expliciteren en presenteren dat die voor verdere bewer-



Lisp is al vele jaren een veelgebruikte ontwikkelomgeving binnen de kennistechnologie en bij andere sectoren van de kunstmatige intelligentie.

king geschikt wordt. In SKE is het ontwerptraject zo minutieus mogelijk gefaseerd, zoals figuur 2 toont, en elke fase heeft weer onderdelen met controlelijsten van activiteiten en controlepunten. Ontwerpen is als probleemtype natuurlijk een typisch voorbeeld van synthese.

DOMEIN

Kennisverwerving, of *kenniselicatie*, wordt in SKE zo veel mogelijk gestuurd vanuit interpretatiemodellen. Daarom zal al in het vooronderzoek worden geprobeerd om vast te stellen of en, zo ja, welke bekende interpretatiemethoden een rol spelen. Tevens gaat het in dit stadium erom alvast een schatting te maken van de omvang van het domein en om een indruk te krijgen hoe dit is opgebouwd.

Een klein domein heeft, om de gedachten te bepalen, tussen de honderd en een paar honderd kernbegrippen maar daartussen kunnen zeer complexe relaties bestaan. De volledigheid van de onderlinge koppelingen tussen begrippen is van meer belang dan de grootte van het domein als geheel.

Het is heel goed mogelijk dat in een later stadium blijkt dat de interpretatiestructuur moet worden veranderd. Ook zal het voorkomen dat het probleemtype helemaal niet in een van de bekende categorieën valt onder te brengen. Bij voorkeur moet dit al in het vooronderzoek blijken omdat in dit geval een nieuw afleidingsmodel moet worden ontwikkeld, met alle consequenties voor de duur en de kosten van het ontwerp.

KENNISVERWERVING

Als de eisen zijn vastgesteld en de haalbaarheidsanalyse tot een positief resultaat heeft geleid, volgt in de analysefase het kennisverwervingsproces. Dit heeft eigenlijk tot doel om de vier lagen te beschrijven voor dit speciale geval van probleemoplossing. Het KADS-model is een raamwerk dat voldoende flexibiliteit bezit om alle overige structuren bij geïdentificeerde inferentiemodellen te laten aansluiten.

In het algemeen wordt bij elicitering een combinatie van technieken gebruikt. Literatuurstudie; breedte-interviews; zeer gerichte en nauwkeurig voorbereide vraaggesprekken met terugkoppeling naar de expert door schriftelijke rapportage; observatie; rapportage door de expert; sessies waarbij via hardop denken wordt geprobeerd een gedachtengang te volgen; het bestuderen van typische gevallen; het zijn allemaal technieken met een eigen plaats en een eigen toepassingsgebied. Een aparte vermelding verdient de zogenaamde 'mock up'-studie. Hierbij speelt in een onderlinge dialoog, die uiteraard op afstand plaatsvindt, de expert het systeem en de kennistechnoloog de toekomstige gebruiker.

VERANDERINGEN

Het verwerven van kennis, wat een belangrijke plaats inneemt in SKE en hier enigszins onderbelicht blijft, is een gecompliceerd en bewerkelijk proces. Vraaggesprekken worden bijvoorbeeld niet alleen opgenomen en vervolgens op schrift gesteld maar ook gesplitst in onderdelen die elementaire informatie geven. Die informatie moet vervolgens in een metaklasse worden geplaatst. Kernbegrippen worden geïdentificeerd en in een mogelijke domeinstructuur geordend.

De kennistechnoloog heeft gelukkig wel twee punten van houvast: de modellen en de expert. Het is immers de bedoeling de kennis van de laatste te 'vertalen' en niet om die zelf te bedenken, laat staan te corrigeren. De deskundige heeft altijd het laatste woord. Informatie kan echter vervolgens wel veranderen en met name verouderen maar in een volgens SKE gestructureerd ontwerp is het domeinraamwerk op zich vrij gemakkelijk te onderhouden. Veranderingen in methoden van probleemoplossen op een vakgebied, in de loop der tijd, vereisen ingrijpendere aanpassingen. Zulke veranderingen komen in de praktijk slechts sporadisch voor. □

Inl.: Bolesian Systems Europe, Steenovenweg 19, 5708 HN Helmond (04920) 23455, fax (04920) 33985.