

Kreatief met KADS

ir. M.E. Uildriks

Agora adviessystemen
v. Riebeeckstraat 10-bis
3531 EJ Utrecht
Telefoon 030-937349

Referaat

CommonKADS is een methode voor het ontwikkelen van kennisystemen. In dit artikel wordt in vogelvlucht beschreven hoe kennismodellering met CommonKADS verloopt. Aan de hand van een case-study worden enkele karakteristieke producten getoond. Het artikel eindigt met enige conclusies over de bruikbaarheid van de methode.

Trefwoorden: CommonKADS, kennisystemen, ontwikkelingsmethode, evaluatie, case-study

Inleiding

In dit artikel presenteer ik tussenresultaten van een case-study naar de bruikbaarheid van de CommonKADS-methode voor het ontwikkelen van kennisystemen (in het vervolg kortweg KADS genoemd). In de case-study wordt een herontwerp gemaakt van een bestaand kennisysteem, het gewasbeschermingsadviesysteem CAPP. Dit systeem (een eerste prototype) is ontwikkeld in de periode 1989 - 1992 door de afdeling Bedrijfssynthese van het Proefstation voor Tuinbouw onder Glas voor de advisering van paprikatuinders bij de gewasbescherming. Momenteel wordt door Priva en Koppert Biological Systems, op basis van het prototype CAPP een systeem ontwikkeld (binnen de bedrijfsregistratie-software) voor ondersteuning van de geïntegreerde bestrijding bij meerdere gewassen.

CommonKADS is een software-ontwikkelingsmethode voor kennisystemen. Vanwege de beperkte beschikbare ruimte, beperk ik me tot een bespreking van KADS als methode om de kennis van een expert te modelleren.

Na een korte inleiding over de aard van de kennis van een expert geef ik aan hoe KADS deze kennis in een aantal stappen modelleert. Ik illustreer de verschillende stappen aan de hand van een selectie van de ontwerpdocumenten betreffende de taakdecompositie en kennismodellering. Ik besluit met enkele conclusies over de

bruikbaarheid van deze onderdelen van de methode.

Kennismodellering met KADS

In de eerste generatie kennisystemen wordt uitgegaan van het idee dat de kennis van een expert bestaat uit feiten en regels. Het redeneren van een expert komt dan neer op het achtereenvolgens toepassen van regels op een gegeven situatie. Consequente toepassing van de regels levert uiteindelijk een conclusie. In deze opvatting is het de taak van de kennisingenieur om te achterhalen welke regels en uitzonderingen de expert precies gebruikt. Deze manier van werken is effectief om tot de juiste conclusies te komen maar veel minder om de conclusie te onderbouwen. Als een gebruiker probeert te traceren waarom het systeem een bepaald antwoord geeft, dan zal hij geconfronteerd worden met een verzameling toegepaste regels waarin hij weinig samenhang kan ontdekken.

In de tweede generatie kennisystemen staat de ordening van het redeneren centraal. KADS is hiervan een exponent. KADS gaat er van uit dat een expert bij het redeneren gebruik maakt van domein-onafhankelijke redeneerstrategieën. Deze worden probleemoplosmethode (PSM) genoemd. Voorbeelden van PSM's zijn:

■ *beoordelen:*

het bepalen of een gegeven geval tot een bepaalde klasse behoort;

- **diagnose:**
het vinden van een oorzaak voor een probleem;
- **ontwerp:**
het transformeren van een eisenpakket naar een ontwerp;
- **planning:**
het vinden van een volgorde die aan gestelde eisen voldoet.

Afhankelijk van de kenmerken van het probleem kan de uitwerking van een PSM verrijkt worden. Een PSM maakt gebruik van een redeneerstructuur. De redeneerstructuur geeft de samenhang tussen redeneerstappen. Redeneerstappen bewerkstelligen veranderingen in de toestand van het probleemoplosproces, bijvoorbeeld het toevoegen of verwijderen van een hypothese.

Kennismodellering volgens KADS gebeurt in achtereenvolgende stappen:

- 1 splitsing van een taak in deeltaken tot deze voldoende afgebakend zijn voor een PSM;
- 2 selectie van een geschikte PSM uit de KADS-bibliotheek;

- 3 aanpassing van de redeneerstructuur voor additionele eisen;
- 4 koppeling van de redeneerstappen aan de domeinkennis;
- 5 maken van een ontwerp-algoritme voor PSM.

Deze werkwijze wordt geïllustreerd in de volgende paragraaf.

Case CAPP: herontwerp met CommonKADS

In deze paragraaf laat ik voor opeenvolgende stappen zien welke ontwerpproducten ontstaan als bij het ontwerp van het gewasbeschermingsadviesysteem de KADS-methode gebruikt wordt.

1. splitsing taak in deeltaken

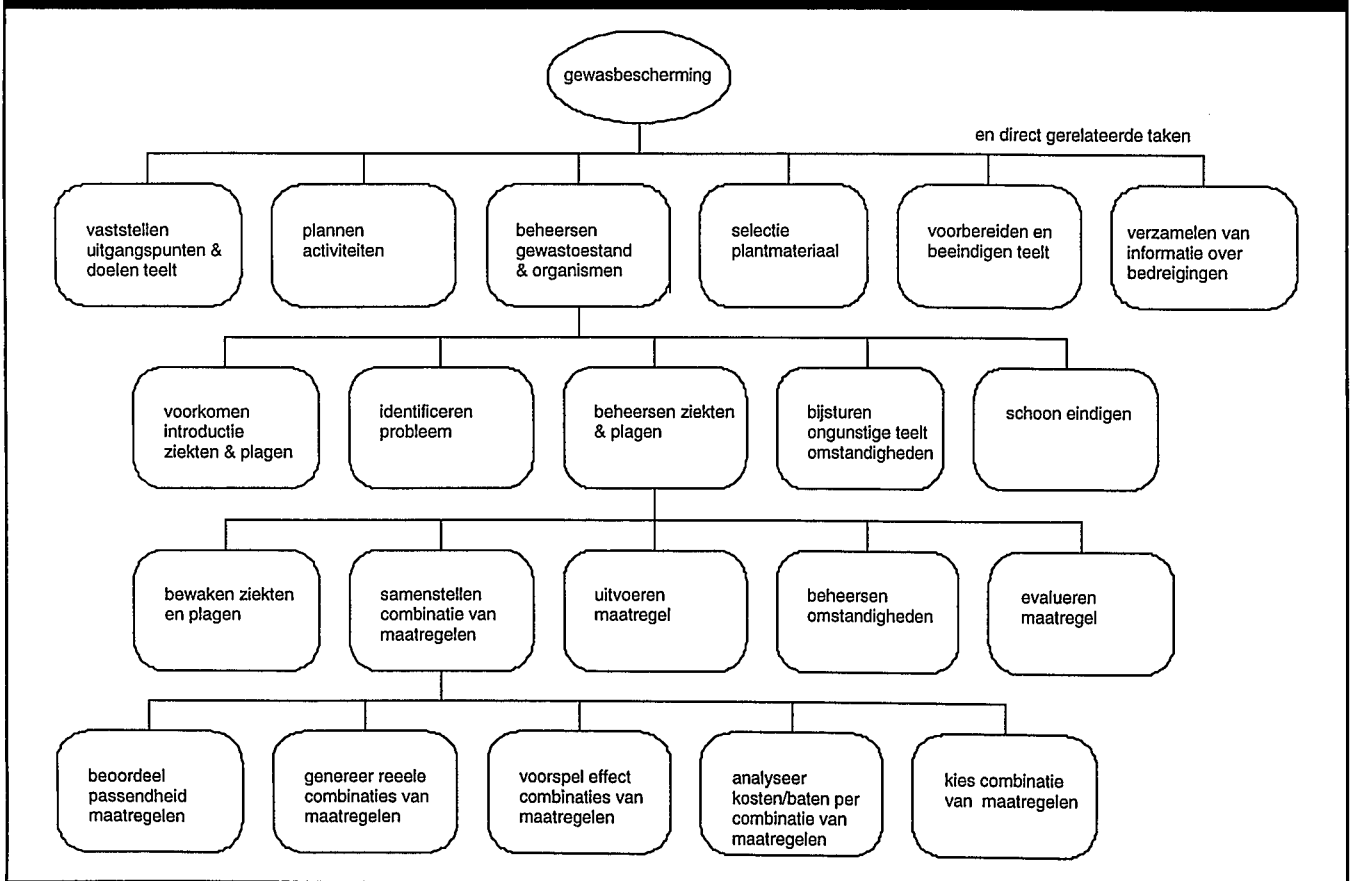
Figuur 1 laat zien hoe ver de decompositie uitgewerkt moet worden voordat een taak voldoende afgebakend is om in een probleemoplosmethode uit te kunnen werken. De functie gewasbescherming raakt nogal wat aspecten van de bedrijfsvoering. In de eerste twee lagen wordt het probleem afgebakend tot 'be-

heersen ziekten en plagen'. Deze taak wordt vervolgens uitgewerkt in diverse taken zoals bewaken van ziekten en plagen, advisering over combinaties van maatregelen en uitvoeren en evalueren van een maatregel. De motivatie om een taak 'samenstellen combinatie van maatregelen' toe te voegen is dat maatregelen niet geïsoleerd beoordeeld kunnen worden: Het chemisch bestrijden van de ene plaag heeft gevolgen voor het biologisch bestrijden van een andere. De taak 'samenstellen combinatie van maatregelen' bestaat uit een opeenvolging van goed afgebakende problemen.

2. selectie geschikte PSM

De eerste taak binnen 'samenstellen combinatie van maatregelen' is 'beoordeel passendheid maatregelen'. Deze taak heeft tot doel het aantal maatregelen dat overwogen wordt, te beperken. Dit is een taak van het PSM-type 'beoordelen' omdat per combinatie van ziekte en maatregel de huidige situatie vergeleken wordt met de algemene regels die aangeven wanneer de maatregel overwogen dient te worden.

Figuur 1 - Vergaande decompositie van taken is nodig voor voldoende afbakening voor PSM



3. aanpassing redeneerstructuur

Bij het uitwerken van deze taak maak ik gebruik van de KADS-bibliotheek. Een taak van het PSM-type 'beoordelen' kent drie hoofdcomponenten: een abstractiecomponent voor de gevalsbeschrijving, een specificatiecomponent voor het meetsysteem en een vergelijkingscomponent.

Uitgaande van het basispatroon kunnen de componenten, en daarmee de redeneerstructuur van deze taak, geëxpandeerd worden (zie Figuur 2). Daarbij wordt een beslisboom gebruikt. De eerste expansie voegt bijvoorbeeld een abstractie-redeneerstap toe aan de basisstructuur. Deze redeneerstap is nodig omdat de gevalsbeschrijving meer informatie bevat dan gebruikt wordt in de vergelijkingscomponent.

Na een aantal van deze stappen uitgevoerd te hebben ontstaat de totale redeneerstructuur die past bij de taak 'beoordeel passendheid maatregelen'.

4. koppeling redeneerstappen aan domeinkennis.

Iedere redeneerstap maakt gebruik van een specifiek gezichtspunt op de domeinkennis. Bijvoorbeeld in het geval van de 'abstraheer geval'-redeneerstap wordt de gestructureerde gevalsbeschrijving op basis van het meetsysteem teruggebracht tot een beschrijving welke uitsluitend uit oogpunt van de vergelijking relevante kenmerken bevat.

Bijvoorbeeld:
neem 'korte gevalsbeschrijving':
 $aantal(trips) = 65$ $aantal(Orius) = 1$.
(Orius is een natuurlijke vijand van de trips). Stel dat de domeinkennis zegt:
mogelijk-passende-maatregel(dichloorvos, true) IF aantal(trips) > 50

Dit is het soort kennis waaruit het meetsysteem is opgebouwd. De redeneerstap 'abstraheer-geval' gebruikt deze regel om te concluderen dat $aantal(trips)$ hier dus blijkbaar relevant is voor de gevalsbeschrijving en $aantal(Orius)$ blijkbaar niet.

De 'geabstraheerde gevalsbeschrijving' wordt dan: $aantal(trips) = 65$. In de redeneerstap 'vergelijking' wordt de geabstraheerde gevalsbeschrijving vergeleken met het 'meetsysteem'. Dit levert als resultaat: *mogelijk-passende-maatregel(dichloorvos, true)*.

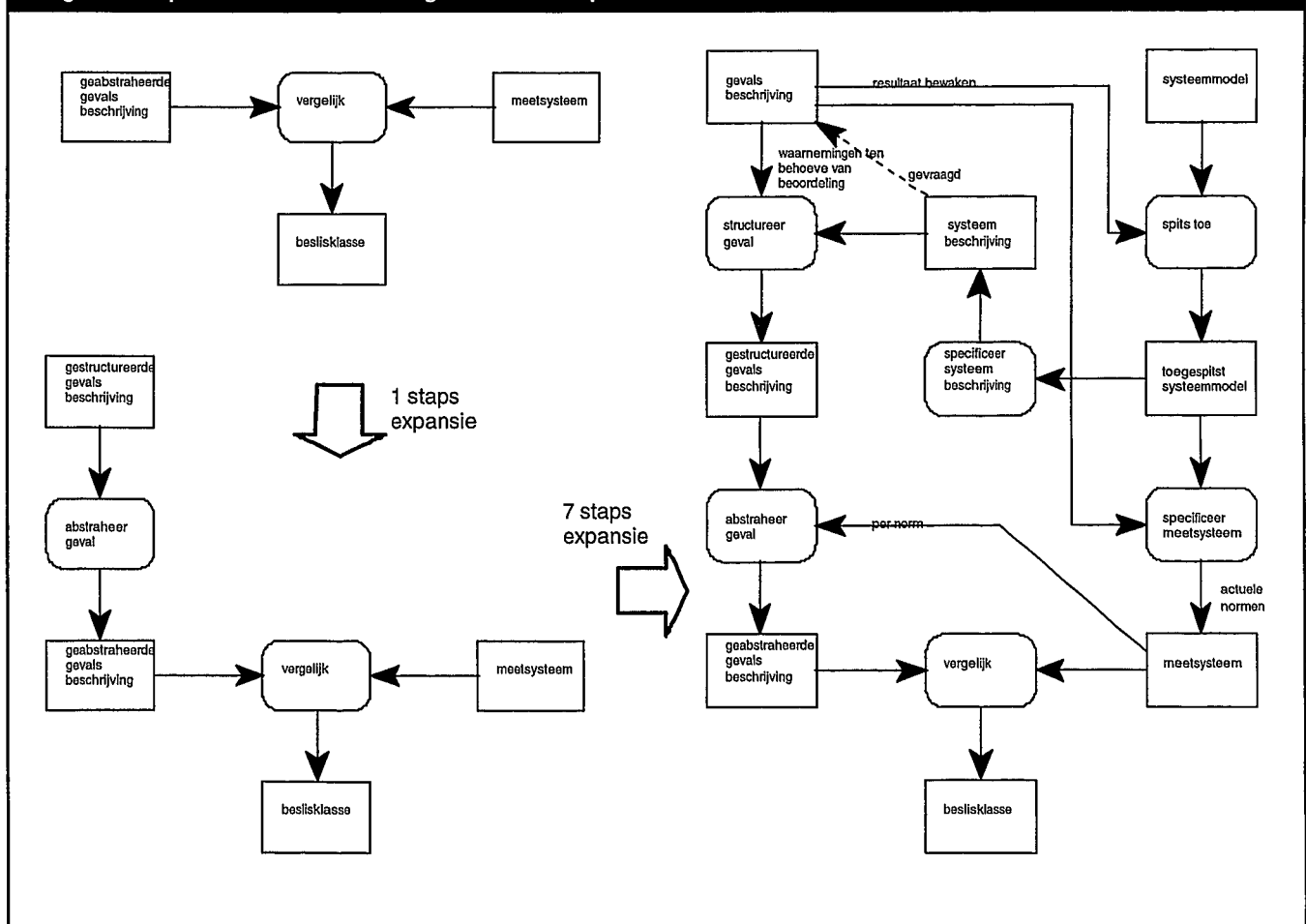
5. ontwerp PSM-algoritme

De redeneerstructuur uit stap 3 laat zich eenvoudig vertalen naar het algoritme voor de probleem-oplosmethode. In het algoritme wordt er van uitgegaan dat de redeneerstappen in stap 4 aangeroepen kunnen worden. Een deel hiervan is te zien in de taakbeschrijving (zie Figuur 3). De taakbeschrijving wordt ook gebruikt om de ontwerpbeslissingen bij de expansie te documenteren (aannamen, QA1 = Question on Abstraction branch 1).

KADS, wat heb je eraan?

Ik wil nog eens benadrukken dat het voorafgaande geen volledig beeld geeft van de KADS-methode. CommonKADS is een

Figuur 2 - Expansie van de beoordelingstaak naar complete redeneerstructuur.



zeer complete en ver uitgewerkte methode gericht op de modellering van kennis en het ontwikkelen van kennissystemen. In dit artikel is alleen de kennismodellering aan bod gekomen.

CommonKADS is een theorie over kennis en een visie op het bouwen van kennissystemen. De theorie over kennis veronderstelt het bestaan van typologieën voor problemen en voor redeneerstappen. Dit is te vergelijken met de benadering in de toegepaste wiskunde waar men ook klassen van problemen met verschillende technieken benadert. Voor veel gebruikers is de typologie van probleemoplosmethoden (PSM) een bron van inspiratie. In de loop der jaren is gestaag aan de uitbreiding van de bibliotheek gewerkt. Op basis van studies van door anderen gerealiseerde systemen, is de systematiek ten behoeve van hergebruik van PSM verder uitgewerkt.

KADS als visie op het bouwen van kennissystemen gaat uit van een topdown benadering van het decomponeren van het pro-

bleemgebied. Uit het herontwerp is gebleken dat je de taakdecompositie ver uit moet werken voordat er sprake is van helder afgebakende problemen. Pas als dit het geval is, kan begonnen worden met het selecteren van een PSM. Hoewel de KADS-bibliotheek zeker nog niet volledig is, is er toch al sprake van belangrijke ondersteuning bij de kennisanalyse en het uitwerken van de PSM.

Is de methode bruikbaar? Lange tijd is het zo geweest dat de methode meer problemen stelde dan dat ze oploste. De richting was wel duidelijk, maar de bibliotheek en systematiek waren nog zó onvolledig en weinig uitgekristalliseerd dat ze eerder een belemmering vormden dan een hulp. Hier begint gelukkig verbetering in te komen, maar de methode is nog steeds niet probleemloos. In de vorige paragraaf is de taak 'beoordelen' uitgewerkt, waarbij de KADS-bibliotheek goede ondersteuning heeft geleverd.

Een belemmering voor de bruikbaarheid is de beschikbare documentatie. Deze is zeer uitgebreid en voornamelijk gericht op de ontwikkelaars van KADS zelf. Daardoor is deze documentatie moeilijk toegankelijk voor derden. Er vinden nog steeds verschuivingen plaats in de gebruikte en nogal KADS-specifieke terminologie, wat het er allemaal niet makkelijker op maakt. Gelukkig wordt momenteel een gebruikershandleiding ontwikkeld.

Het onderzoek waarop dit artikel is gebaseerd is nog niet ver genoeg gevorderd om echte conclusies over de bruikbaarheid te trekken. Mijn persoonlijke ervaring tot nu toe is toch vooral: inspirerend maar taai. Maar geldt dit niet altijd voor het ontwikkelen van kennissystemen?

Literatuur

Schreiber, G., Wielinga, B., Breuker, J. (Eds.) (1993)

KADS, a principled approach to knowledge-based system development, Academic Press.

Figuur 3 - Taakbeschrijving, algoritme en aannamen 'beoordeel-passendheid-maatregel'.

Taak: **beoordeel-passendheid-maatregel**

taak-definitie

doel: beoordelen of een maatregel mogelijk passend is;

invoer: maatregel, gevalsbeschrijving (initieel en uitgebreid); waarnemingen;

uitvoer: waarheidswaarde: boolean die aangeeft of maatregel wel/niet passend is;

taak-inhoud

deeltaken: spits-toe, specificeer meetsysteem, specificeer systeem beschrijving, structureer geval, abstraheer geval, vergelijk;

additionele-rollen: toegespitst systeemmodel, meetsysteem, systeem beschrijving, gestructureerde gevalsbeschrijving, geabstraheerde gevalsbeschrijving;

besturing:

beoordeel-passendheid-maatregel(m:maatregel + g:gevalsbeschrijving ->b:waarheidswaarde) =

spits-toe(g':initieel-gevalsbeschrijving + m:maatregel + s:systeemmodel ->s':toegespitst-systeemmodel) ...

REPEAT

abstraheer-geval(x:norm + g':gestructureerde-gevalsbeschrijving ->g'':geabstraheerde-gevalsbeschrijving)

UNTIL vergelijk(x:norm + g'':geabstraheerde-gevalsbeschrijving ->>true)

aannames:

Abstractie component:

QA1: De gevalsbeschrijving is niet abstract genoeg om direct te kunnen vergelijken

QA2: De gevalsbeschrijving moet eerst gestructureerd worden

QA3: Er is maar één soort geval dat geanalyseerd hoeft te worden

Specificatie component:...

motivatie:...

3. Er is een vaste verzameling fundamentele criteria die een rol kunnen spelen: schade, populatiegroei, omvang, infectiedruk, verspreiding (inclusief natuurlijke vijanden). Alle ziekten kunnen in deze termen beschreven worden