

Het gebruik van de Classify Workbench

Een hulpmiddel bij het opzetten van complexe gegevensstructuren

R.F.I. van Himste
P.C. de Jong

ATC-SIVAK
Postbus 1032
8200 BA Lelystad
Telefoon 03200-34900, telefax 03200-33824

Referaat

Er zijn tegenwoordig voldoende software-pakketten beschikbaar voor het ontwikkelen van allerlei registratieve systemen. Echter, wat te doen met systemen die voortdurend aan veranderingen onderhevig zijn? De aandacht dient zich dan meer te concentreren op de analyse-fase en minder op de technische implementatie. In dit artikel wordt een Workbench met achterliggende methodiek besproken om snel systemen te ontwikkelen.

Trefwoorden: Workbench, CASE-tool, Teeltbegeleidingssysteem, Classify

Inleiding

Vanaf 1988 wordt door ATC-SIVAK, in samenwerking met andere organisaties, gewerkt aan de ontwikkeling van teeltbegeleidingssystemen (TBS). Een teeltbegeleidingssysteem is een kennisintensief systeem voor telers, dat ondersteuning biedt bij het nemen van teeltbeslissingen. Hierbij kan gedacht worden aan bijvoorbeeld de hoeveelheid stikstof op een bepaald tijdstip gedurende het groeiseizoen, het wel of niet toepassen van gewasbeschermingsmiddel, welk ras het meest geschikt is in de bedrijfseigen situatie, etc. Eén van de kenmerken van deze systemen is dat er diverse complexe relaties tussen de gegevens bestaan. Op dit moment wordt de laatste hand gelegd aan de technische opwaardering van BETA (suikerbieten) en CERA (granen). Dit omvat het verbeteren van de

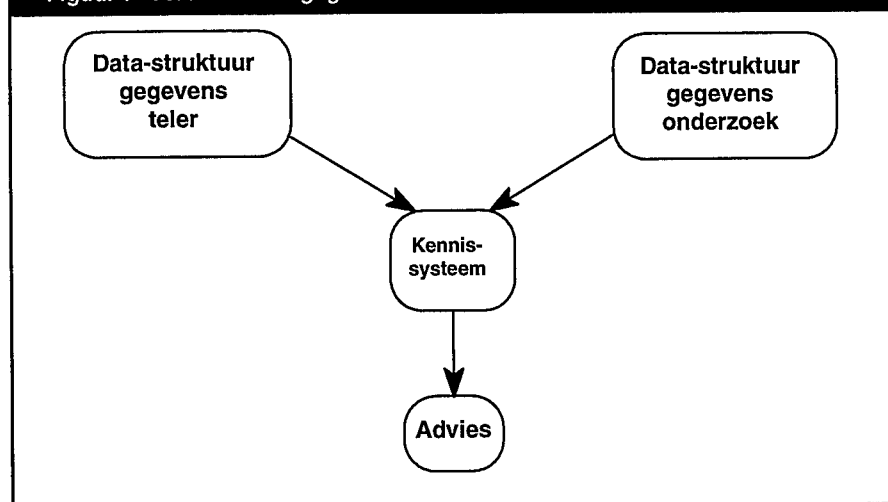
onderhoudbaarheid van programmatuur, functionele en technische documentatie en gebruikersvriendelijkheid en het opzetten van een test-omgeving. Daarnaast wordt gewerkt aan de realisatie van een teeltbegeleidingssysteem voor spruitkool en bloemkool (KOBAS). Ten behoeve van het gegevensonderhoud is gebruik gemaakt van een Workbench die vanuit de geanalyseerde functionele eisen, de bestanden en programmatuur genereert. Over de mogelijkheden en ervaringen met de Classify Workbench en bijbehorende methodiek gaat dit artikel.

Data-onderhoud

Een belangrijk kenmerk van teeltbegeleidingssystemen is dat er naast een advies- of kennisgedeelte, gebruik wordt gemaakt van diverse gegevens. Deze gegevens zijn gedeeltelijk afkomstig van het eigen bedrijf, maar ook veel gegevens zijn afkomstig van derden. Veelal betreft dit onderzoeksgegevens, maar ook algemene gegevens over bijvoorbeeld rassen en merken van gewasbeschermingsmiddelen vallen in deze categorie (zie figuur 1).

De teeltbegeleidingssystemen BETA, CERA en KOBAS hebben een gemeenschappelijke gegevensstructuur als basis. Omdat de systemen verschillende gewassen ondersteunen, moet het onderhoud door verschillende personen en/of organisaties gebeuren. Dit betekent dat de pro-

Figuur 1 - Verschillende gegevensbronnen leiden tot een advies



opties zoals display; alle invoer omzetten in hoofdletters, etc;

- *referentiële-integriteit*; in de DD wordt geregeld dat wanneer een NAW-record wordt verwijderd, ook alle andere gegevens over dit record verwijderd worden; Bijvoorbeeld alle orders en orderregels van een klant worden verwijderd als klant verwijderd wordt;
- *entry- en exit-procedure op velden*; bepalen of een veld benaderd mag worden, afhankelijk van de waarde van een ander veld;
- *functionele relaties*; bv. het bijhouden van een redundant gegeven, zoals saldo van een bankrekeningnummer;
- *een vast gedefinieerd lijstje*; het kiezen van een waarde uit een lijstje; bv. "Ja of Nee" of "Blauw, Groen en/of Rood";
- *toevoegen*; bij een veld kan aangegeven worden of deze in de lijst toegevoegd moet worden. Wanneer een order wordt toegevoegd, moet een klant en artikel gekozen worden. Via een lijst kan een keuze gemaakt worden. De analist kan zelf aangegeven welke velden in deze lijst te zien zijn;
- *aangeven welke teksten op het scherm te zien zijn voor het veld*. Bij een lijstgeoriënteerd scherm kan een andere schermtekst opgegeven worden, dan bij een record-georiënteerd scherm.

Wanneer alle relevante gegevens voor de database-bestanden zijn gedefinieerd, kan begonnen worden met het definiëren van de data entry-programma's voor de eindgebruikers. Hierbij wordt gewerkt met de volgende hiërarchie (zie figuur 3):

- *applicaties*; een verzameling programma's ten behoeve van een bepaald project (gewas);
- *programma's*; een programma is een zelfstandige eenheid die één of meerdere views bevat. Het programma zal in eerste instantie vragen om een gebruikersidentificatie en daarna de action-bar activeren.
- *views*; een view is een functionele eenheid, die op één of meerdere bestanden betrekking kan hebben en bevat dus één of meerdere subsystemen. Een view wordt via de functie 'Pull-Down' geactiveerd.

- *subsystemen*; een subsysteem is een verzameling zooms, verantwoordelijk voor de data-entry op één bestand.
- *zooms*; alle gegevens worden weergegeven in zogenaamde zooms. De gegevens uit een bestand worden gegroepeerd in zooms die functioneel samenhangende velden combineren (bv. financiële gegevens, adresgegevens). Er zijn verschillende zoom-types, afhankelijk van de soort informatie (formulier, vrije tekst, checkbox -keuze, etc).

Alle gegevens, die worden ingevoerd via deze programma's, worden via de DD gevalideerd voordat deze daadwerkelijk worden vastgelegd in de database. De DD's verzorgen de consistentie van de database. Via het definiëren van programma's volgens bovenstaande hiërarchie, kunnen snel en op eenvoudige wijze programma's gemaakt worden. De meeste tijd wordt, door de analist, besteed aan de opzet van een bepaald bestand.

Wanneer wijzigingen in de DD's optreden, moeten alle relevante programma's opnieuw gecompileerd worden. Dit mechanisme wordt ook wel aangeduid als passieve Data Dictionary. Oracle7 heeft een zogenaamde actieve Data Dictionary en heeft als voordeel dat altijd direct met de recente DD gewerkt wordt. Binnen Data-

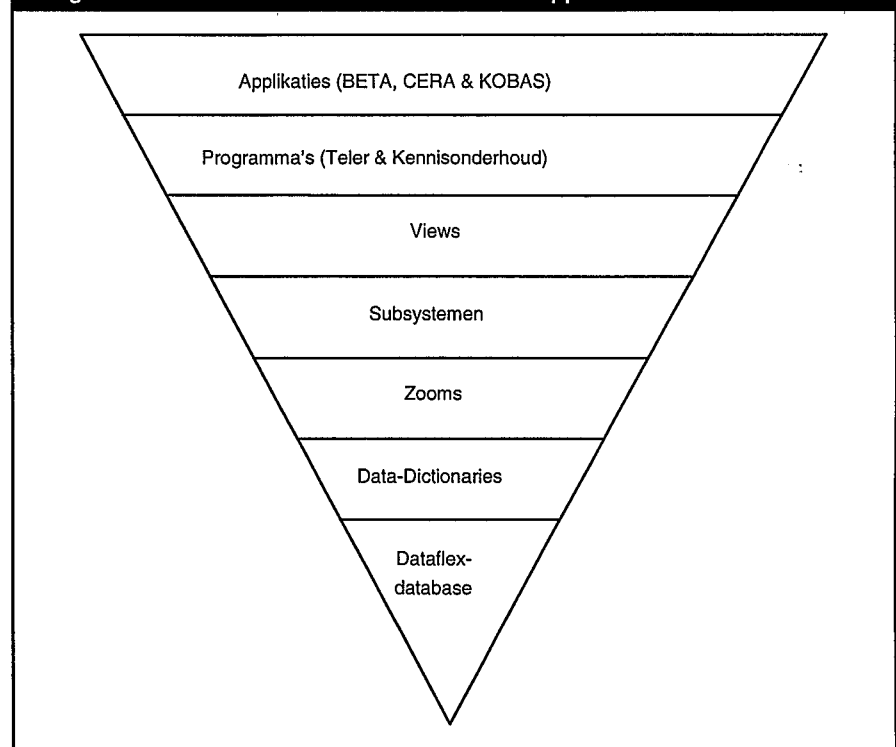
flex moeten gewijzigde DD's eerste opnieuw worden gecompileerd, willen deze actief worden in programma's.

De Classify Workbench

Wanneer met een project wordt begonnen, wordt veelal eerst een uitgebreid functioneel ontwerp gemaakt. Tegenwoordig echter, is er een groeiende behoefte om snel een prototype te maken van het programma in plaats van het bekende traject, functioneel ontwerp, technisch ontwerp en tenslotte de bouw van het programma. Boven genoemde Classify-methodiek biedt de mogelijkheid om alle bijbehorende bestanden voor programma's, views, subsystemen, etc. zelf te schrijven. Echter snel zal blijken dat soortgelijke source code-bestanden op elkaar zullen lijken. Er ontstaat dan al snel de gewoonte om bestanden te kopiëren en een aantal source code-regels aan te passen. Echter, wanneer aanvullingen en/of verbeteringen op de Data Dictionary worden geïmplementeerd, zullen alle bijbehorende bestanden aangepast moeten worden.

Met de Classify Workbench is het mogelijk om alle gegevens met betrekking tot DD, Programma, View, Subsysteem en Zooms te definiëren. Vervolgens kan de source code voor de applicatie worden ge-

Figuur 3 - De hiërarchie voor het definiëren van applicaties



genereerd. Na compilatie kunnen de diverse programma's worden getest en gebruikt. Het help-systeem wordt gegenereerd in het uiteindelijke systeem vanuit de gegevens die de analist heeft ingevoerd in de Workbench. Tevens kan vanuit de Workbench voor zowel de analist als voor de eindgebruiker een handleiding worden afgedrukt.

Ervaringen

Bij het ontwikkelen van de onderhoudsomgeving voor teeltbegeleidingssystemen is dankbaar gebruik gemaakt van de mogelijkheden die de methodiek Classify in combinatie met de Workbench biedt. In een omgeving met complexe gegevensrelaties wordt de basis van het systeem inclusief handleiding en help-systeem opgezet. Samenvattend zijn de volgende voor- en nadelen m.b.t. het gebruik van Classify en de Workbench te noemen.

Voordelen:

- Er wordt verhoudingsgewijs beduidend meer tijd besteed aan het analyseren van de gegevens, dan aan het programmeren. Het programmeren bestaat op dit moment nog uit het maken van nieuwe veld-specifieke entry-, exit- en validatie-procedures;
- De Classify Workbench is een goede aanvulling op de analyse-tool van ADW (opvolger van IEW). De modules van ADW voor het genereren van source code zijn vrij duur en genereren geen specifieke programma-code (COBOL, DB2), waardoor zelf aanvullende programmering nodig is voor het maken van het uiteindelijke program-

ma. Daarnaast zijn, samen met Calvin Consultancy, uitbreidingen op de Workbench gemaakt voor het data-onderhoud van teeltbegeleidingssystemen. Tevens moeten de uiteindelijke programma's op een PC kunnen draaien, die in de praktijk gangbaar mag worden geacht (PC met 386SX-processor);

- Het invullen van de database-structuur gaat vrij snel. Na het invullen van de overige gegevens (zoom, subsysteem, view en programma) kan in korte tijd een werkend programma worden gemaakt;
- Aangezien alle gegevens van de Workbench in een database zijn opgeslagen, kan met meerdere mensen tegelijkertijd worden gewerkt aan de analyse van de gegevensstructuur;
- Programma, help en documentatie zijn volledig op elkaar afgestemd. Ze worden immers uit dezelfde bron gegenereerd. Tevens is het onderhouden van help en documentatie eenvoudiger;
- Alle programma's zijn eenduidig gedefinieerd. Zij hebben uiterlijk allemaal dezelfde werking en zijn volgens dezelfde hiërarchie opgebouwd.

Nadelen:

- Niet alle programma's zijn volledig te genereren. Op dit moment wordt 80% van de programma's gegenereerd. Met name voor ingewikkelde toepassingen, zoals teeltplanning voor KOBAS, blijkt het niet mogelijk om gebruik te maken van de standaard source code-generatie vanuit de Workbench. Dit heeft echter meer te maken met het extra toevoegen van functionaliteit, dan een tekortkoming van de Workbench.

Zo is bijvoorbeeld voor het onderdeel 'Teeltplanning' van KOBAS voor het invullen van de teeltactiviteiten (oogsten, planten, etc.) een automatische invul-procedure ontwikkeld, zodat de gebruiker een hulpmiddel heeft bij het invullen van de begin- en eindperioden van deze teeltactiviteiten;

- Daar de Classify Workbench pas kort beschikbaar is, ontbreken nog diverse applicatiebeheer-opties zoals het aan-geven wanneer programma's opnieuw gegenereerd dienen te worden, wanneer opnieuw documentatie afgedrukt moet worden etc. Dit wordt momenteel opgelost door regelmatig na invoer van wijzigingen alles te genereren, te compileren en af te drukken.

Tot slot

Classify en de Classify Workbench zijn hulpmiddelen om het technisch onderhoud van het registratieve deel van teeltbegeleidingssystemen te vergemakkelijken. Wanneer men eenmaal de werkwijze onder de knie heeft, is het vrij eenvoudig om data-onderhoud te verrichten. Als zodanig is dit waarschijnlijk niets anders dan andere CASE-tools. Classify en de Classify Workbench zijn geënt op Dataflex-programmatuur. Als binnen de projecten een andere programmeertaal was gekozen, dan was waarschijnlijk ook voor een andere Workbenches gekozen. Het grote voordeel van het gebruik van tools is dat eenvoudig en snel flexibele systemen zijn te ontwikkelen, die zowel consistent als beheersbaar zijn. Daarnaast zijn help en handleidingen direct beschikbaar.