

BETA & CERA; DE USER-INTERFACE

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

Door SIVAK, PAGV en andere participanten, worden momenteel twee teeltbegeleidingssystemen ontwikkeld; één voor suikerbieten (BETA) en één voor Granen (CERA). In dit artikel wordt aangegeven hoe de user-interface voor deze twee systemen is opgezet.

Inleiding

In de bestuurlijke informatica heeft de user-interface een lange tijd een ondergeschikte rol gespeeld.

Onder user-interface (gebruikers-interactie) wordt in dit artikel verstaan alle mogelijke manieren / vormen van communicatie tussen de gebruiker en een systeem. De nadruk ligt met name op het gebruik van beeldscherm, toetsenbord en muis. Strikt gezien dienen ook print- of lijst lay-outs hieronder te worden verstaan.

De laatste jaren echter wordt steeds meer de nadruk gelegd op de gebruiker; deze is tenslotte degene die gebruik maakt van het ontwikkelde informatiesysteem. Ook in de agrarische sector, zijn met name door het onderzoek vele (deel-)systemen ontwikkeld, waarbij de inhoud van het programma meer op de voorgrond stond dan de presentatie naar de gebruiker.

De presentatie naar de gebruiker is echter net zo belangrijk als de inhoudelijke correctheid van het informatiesysteem. Het rendement van een informatiesysteem is immers het produkt van de kwaliteit van het informatiesysteem en de acceptatie van het informatiesysteem.

rendement informatiesysteem = kwaliteit x acceptatie

Bij de ontwikkeling van teeltbegeleidingssystemen door het SIVAK en het PAGV in samenwerking met andere organisaties staat de acceptatie door de gebruiker voorop. Deze acceptatie is bevorderd door het in een vroeg stadium betrekken van de toekomstige eindgebruikers bij de ontwikkeling van de teeltbegeleidingssystemen en anderzijds door het ontwikkelen van een eenvoudige, makkelijke én toegankelijke user-interface.

In dit artikel zal na een korte inleiding over teeltbegeleidingssystemen en de gebruikte terminologie en begrippen, de user-interface van BETA en CERA nader worden bekeken. Verder wordt aan de hand van algemene ontwerp-principes de nadere uitwerking hiervan binnen BETA en CERA toegelicht. Tenslotte wordt besloten met een aantal ervaringen zoals die binnen de projecten BETA en CERA zijn opgedaan.

Teeltbegeleiding

BETA en CERA zijn systemen die de akkerbouwers ondersteunen bij het nemen van beslissingen tijdens de

teelt van resp. suikerbieten en granen. Dit begint bij het kiezen van een ras tot het moment van oogsten van het gewas, met daartussen o.a. de bestrijding van onkruiden en parasieten, stikstofbemesting etc.. Het zijn systemen die niet dagelijks door de telers worden gebruikt en hebben dan ook een lage gebruiksfrequentie en -intensiteit. Deze systemen dienen dan ook zeer gebruikersvriendelijk te zijn, zodat ze gemakkelijk kunnen worden gebruikt. Er dient dan ook zeer veel aandacht worden besteed aan de user-interface; de communicatie tussen gebruiker en het computersysteem.

Bij de ontwikkeling van het eerste teeltbegeleidingssysteem (BETA) is tijdens het opstellen van het functioneel ontwerp een demonstratiemodel gemaakt om zo met de toekomstige gebruikers te kunnen discussiëren over hoe het systeem er uit diende te zien. Dit demonstratiemodel is gemaakt op een Apple Macintosh met Hypercard. De keuze viel op deze combinatie vanwege de gebruikersvriendelijkheid van de Apple Macintosh en de eenvoud van Hypercard. Met deze combinatie was het mogelijk in zeer korte tijd een demonstratiemodel te ontwikkelen. Na afloop van de demonstratieperiode konden onder andere de volgende conclusies worden getrokken:

- het gebruik van de Macintosh sloeg niet aan vanwege de geringe beschikbaarheid van agrarische software voor dit type PC; terwijl zulke software wel voor de MS-DOS PC's te krijgen is. Verder hadden een aantal van de klankbordgroepen al een MS-DOS PC;
- het gebruik van muis en de daarbij behorende interface bestaande uit scroll-boxen, menu's, etc. had wel succes. Met weinig kennis van zaken bleek men snel in staat om met zo'n systeem te werken.

Conclusie:

Teeltbegeleidingssystemen dienen te draaien onder MS-DOS waarbij het gebruik van een 'grafische' user-interface heel gewenst is. Daarnaast volgde uit het ontwerp dat het te gebruiken ontwikkelhulpmiddel database-faciliteiten diende te hebben. Voorts diende het geheel ook te werken op PC's van het type XT, daar in de doelgroep meerdere van deze machines aanwezig zijn.

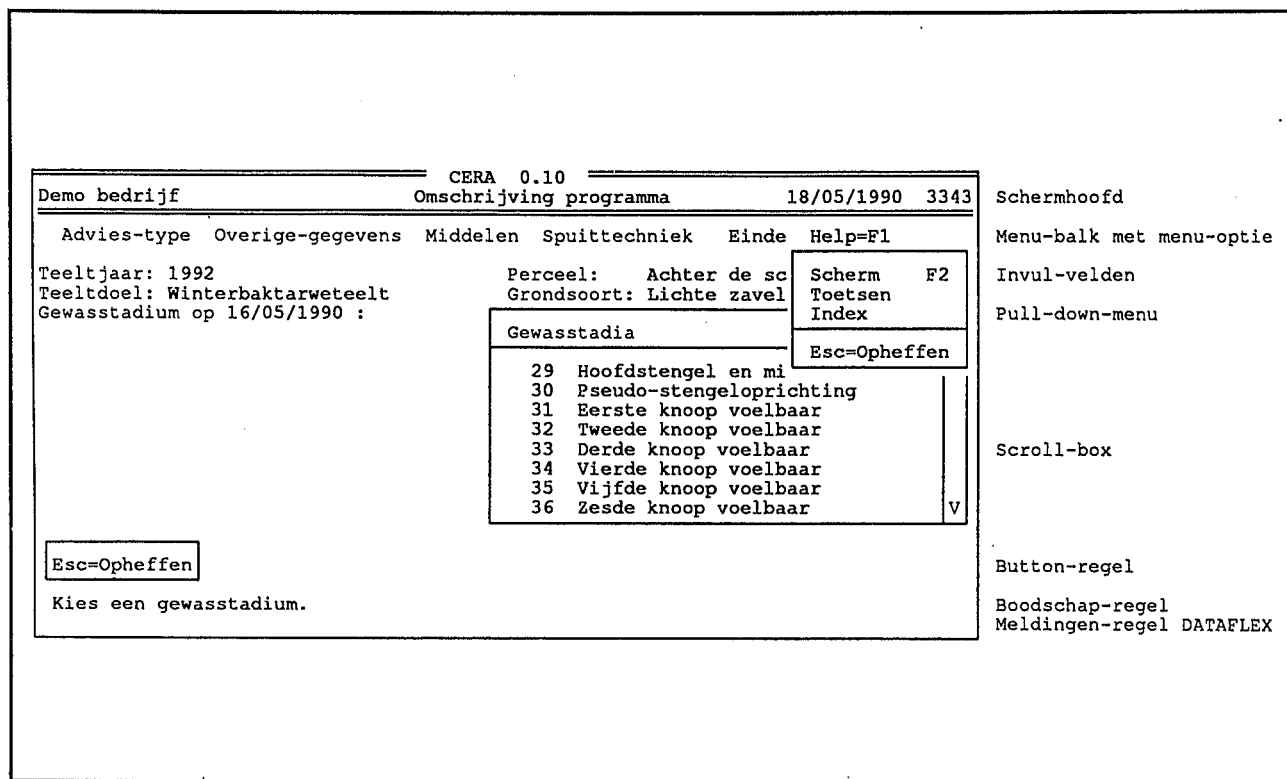
Na enig marktonderzoek is uiteindelijk gekozen voor DATAFLEX 3.0, een nog in ontwikkeling zijnde 4e-generatietaal, welke min of meer aan al de bovenstaande eisen voldoet.

Terminologie en begrippen

Aan de hand van een voorbeeld beeldscherm uit CERA, zullen een aantal begrippen worden uitgelegd en nader worden toegelicht.

Een voorbeeld van de opbouw van de user-interface van BETA en CERA is weergegeven in Figuur 1. Het scherm is opgebouwd uit de volgende onderdelen:

- het schermhoofd met daarin de naam van het teeltbegeleidingssysteem en versie-nummer, de naam van de



Figuur 1: Weergave van een scherm van CERA

- gebruiker, de omschrijving van de programmafunctie, de datum en een uniek schermnummer waaraan de schermhelp is gekoppeld (te activeren door toets F2);
 - de menubalk met een aantal menu-opties. Hierin zijn altijd de menu-opties 'Einde' en 'Help=F1' opgenomen;
 - pull-down-menu's van de menu-opties. Een menu-optie kan uit een aantal onderdelen zijn opgebouwd. Via de pull-down-menu wordt een keuze gemaakt;
 - het invul-scherm met een aantal invul-velden. Deze kunnen op een drie-tal wijzen worden ingevuld:
 - directe invoer van de waarde. Dit geldt bijvoorbeeld voor het invullen van de datum;
 - invoer via een scroll-box. Uit de scroll-box wordt één van de toegestane waarden gekozen en op het scherm getoond. Bijvoorbeeld het gewasstadium;
 - de antwoorden van het vraag-antwoord-spel. Bij een vraag dient de gebruiker te kiezen uit 'Ja' of 'Nee'. De betreffende keuze wordt als omschrijving getoond.
 - de adviezen worden meestal in een scroll-box getoond. Dit betreft veelal adviesteksten en geadviseerde gewasbeschermingsmiddelen;
 - de button-regel (regel 21, 22 en 23), waarbij een actie wordt geactiveerd met de muis of door de betreffende functietoets; of verder gaan met de volgende actie (binnen BETA & CERA is dit) of terug naar de vorige actie "ESC";
 - de boodschap-regel (regel 24), waarin diverse (fout-)meldingen worden aangegeven over water aan de hand is dan wel wat er wordt verwacht;
 - de laatste regel (regel 25) wordt gebruikt door DATA-FLEX om zgn 'run-time errors' te melden. Als het goed is zal een gebruiker deze nooit te zien krijgen.
- Naast de genoemde onderdelen zijn nog twee begrippen van toepassing, namelijk:
- de grafische interface:

Grafische interfaces maken gebruik van de grafische mogelijkheden van het beeldscherm. Binnen de teeltbegeleidingssystemen gebruiken we de standaard 'character-mode' van het beeldscherm, dit betekent dat de interface wordt ontwikkeld voor een scherm van 80 bij 25 posities. Bij grafische weergave is dit afhankelijk van de grafische kaart. Voorbeelden zijn programma's die onder MS-Windows werken.
 - het object-georiënteerde systeemontwerp:

De programma-componenten worden gezien als objecten en staan door middel van boodschappen met elkaar in verbinding (Sommerville, 1988). Ieder object kent meestal een schermrepresentatie en een aantal toegestane operaties, die ingrijpen op de eigenschappen van het object. BETA en CERA zijn volgens een object-georiënteerd systeemontwerp opgezet.

Daarnaast zijn de programma's ook modulair opgezet. Dit betekent dat een bepaalde module door verschillende programma's kan worden gebruikt; vergelijkbaar met subroutine-bibliotheken. Zo wordt het kiezen van een perceel als een standaard-routine binnen een programma gebruikt. Deze module is te zien als het object 'perceel'; met bijbehorende routines om met het object te manipuleren.

Object-georiënteerd systeemontwerp

Aan de hand van een voorbeeld zal worden toegelicht hoe het object-georiënteerde systeemontwerp binnen DATAFLEX vorm heeft gekregen.

Het object 'perceel' bestaat uit de volgende onderdelen dan wel kent de volgende eigenschappen:

- het object perceel wordt geactiveerd vanuit het object 'teeltjaar'. Dit betekent in BETA dat in de scroll-box percelen van het object 'percelen' alleen de percelen met suikerbieten worden getoond in dat teeltjaar;
- bestaat er geen perceel (waarde), dan verschijnt de lege scroll-box met een foutmelding;
- bestaat er één perceel, dan wordt deze automatisch geselecteerd. Bij meerdere percelen dient de gebruiker er één te kiezen door deze met de muis aan te klikken of deze met de toets "ENTER" te bevestigen;
- toetst men tijdens het kiezen van een perceel op de toets "ESC" of klikt men de betreffende button aan, dan verdwijnt de scroll-box en gaat men terug naar de vorige actie. Dit is het kiezen van een teeltjaar (Figuur 1);
- na een perceel gekozen te hebben verdwijnt de scroll-box en wordt het gekozen perceel in het invul-veld getoond. Daarna wordt het volgende object geactiveerd;
- wordt het invul-veld perceel met de muis benaderd, dan wordt het huidige object netjes afgehandeld en verschijnt de scroll-box met de percelen waarin het perceel uit het invul-veld als gekozen wordt aangegeven.

Kortom, object-georiënteerd programmeren biedt veel vrijheden voor de programmeur. Deze vrijheden dienen echter voor de programmeur te worden beperkt, zowel functioneel als technisch, zodat de gebruiker met een eenduidige applicatie kan werken. Het technisch onderhoud wordt op deze manier zowel op de korte als op de lange termijn ook vereenvoudigd.

Ontwerp-principes

Nu een aantal begrippen zijn toegelicht zal verder worden ingegaan op een aantal algemene ontwerp-principes en de toepassing daarvan binnen BETA en CERA. Het gaat om de volgende principes:

- de interface dient aan te sluiten op de gebruikersfrequentie en -intensiteit van de akkerbouwers. Veel programma's worden een enkele keer per jaar gebruikt. Daarom dient het systeem eenvoudig te leren en te bedienen zijn;
- consistent gebruik maken van objecten binnen de applicatie. Bijvoorbeeld: de 'grondsoort'-scroll-box wordt als één module (subroutine) aangeroepen. Hierdoor is de omschrijving van de box overal in de applicatie gelijk, de inhoud wordt op dezelfde wijze gesorteerd en er wordt dezelfde boodschap getoond;
- consistent gebruik maken van acties. Indien een waarde wordt gekozen uit een scroll-box, een button, een functietoets of een menu-optie wordt geactiveerd, dient overal op dezelfde wijze de actie in de applicatie afgehandeld te worden;
- er dienen hulp-faciliteiten aanwezig te zijn. Dit gebeurt in BETA en CERA in de vorm van een boodschap,

onderaan het scherm en help-faciliteiten per scherm. De help is per menu-optie uit de menu-balk ingevuld;

- er dienen uiteraard alleen die zaken op het scherm getoond te worden welke op dat moment ook relevant zijn. Bijvoorbeeld; van percelen is meer informatie in de database bekend. Het is mogelijk om alle informatie te tonen aan de gebruiker, maar niet alle informatie is op dat moment van belang om in het besluitvormingsproces mee te nemen. Zo is de ligging in het waterwingebied alleen van belang wanneer er gewasbeschermingsmiddelen geselecteerd worden; deze kunnen namelijk in een waterwingebied wel eens verboden zijn;
- de vrijheden die het muis-georiënteerd/object-georiënteerd programmeren biedt; dienen in het ontwerp te worden aangegeven.

Ervaringen

De ontwikkeling van BETA en CERA zal naar verwachting eind 1990 worden afgerond. Vanuit de klankbordgroepen, die de opgeleverde deelsystemen van BETA en CERA testen, worden reacties ontvangen. Van deze reacties en de ervaringen opgedaan bij het opzetten van de user-interface kan het volgende worden weergegeven:

- In het functioneel ontwerp worden de verschillende procedures (= processen, = teeltbeslissingen) beschreven. Hierin staan de functionele specificaties o.a. de programma-structuren en beeldscherm-layouts beschreven. Vanuit het functioneel ontwerp wordt het technisch ontwerp opgesteld. Hierin wordt rekening gehouden met het ontwikkelhulpmiddel; DATAFLEX 3.0. Het blijkt in de praktijk dat de rekenregels en de globale scherm-afhandeling goed beschreven staan in het functioneel ontwerp. De specifieke scherm-afhandeling dient uitgebreid beschreven te worden in het Technisch Ontwerp. Daar dienen ook de andere technische specificaties beschreven te zijn. Genoemd kunnen worden:
 - opbouw van de scherm-elementen; o.a. objecten;
 - routing van de objecten op het scherm aangeven inclusief de mogelijke acties. SAA biedt hiervoor een aantal richtlijnen;
 - te gebruiken bestanden, buttons, modules (packages).

Het blijkt dat in de technische ontwerpen veel dezelfde functionele zaken op verschillende wijzen technisch worden opgelost. Daarom is het noodzakelijk om standaards (richtlijnen) op te zetten voor de ontwikkeling van teeltbegeleidingssystemen. Deze standaards blijken nooit voldoende gedetailleerd te kunnen worden vastgelegd en zullen dan ook in de loop van beide projecten worden aangescherpt. Dit komt met name door het feit dat de gebruiker middels de muis vele vrijheden heeft. De diverse mogelijkheden dienen voor de programmering zoveel mogelijk vooraf te worden aangegeven. Dit geldt met name voor de volgende onderdelen:

- het schermhoofd;
- naamgeving scroll-boxen, buttons, variabelen en menu-opties;

- positionering van de verschillende objecten (schermen);
- afhandeling van acties (buttons en functietoetsen);
- opbouw programma.

Teeltbegeleidingssystemen worden gemaakt voor de individuele teler(tuinder). Deze systemen moeten dan ook zeer gebruikersvriendelijk zijn. Ze moeten geënt zijn op persoonlijk gebruik met een geringe intensiteit en frequentie. Dit wordt beoogd door de volgende middelen:

- minimaal invoer-werk, door het gebruik van objecten (scroll-boxen en menu's);
- door het gebruik van velerlei normen is een teeltbegeleidingssysteem eenvoudig in het gebruik, zodat 'lastige' invoer via toetsenbord zoveel mogelijk wordt vermeden;
- in kort tijdsbestek te leren, doordat het initiatief bij het computersysteem ligt. De gebruiker wordt aan de hand van de verschillende objecten door de applicatie geleid. Dit is met name gemakkelijk voor gebruikers die niet dagelijks met het programma werken.

Conclusies

Het opzetten van een consistente user-interface blijkt een lastige doch creatieve bezigheid te zijn. Dit komt met name doordat het ontwikkelteam bestaat uit een vijftal programmeurs. Elke programmeur blijkt functionele zaken op een verschillende technische wijze op te lossen. Het

blijkt dan ook zeer waardevol te zijn om een uitgebreid technisch ontwerp te schrijven, waarbij met name de afhandeling van het scherm wordt beschreven. Daarnaast is het zeer praktisch om naast de algemene standaards (zoals opgelegd door de methodieken van systeemontwikkeling; bv. LIA) ook standaards op te stellen voor de presentatie naar de gebruiker; de user-interface. Door een éénduidige user-interface is het succes van het programma niet afhankelijk van de user-interface, maar kunnen de gebruikers de teeltbegeleidingssystemen inhoudelijk beoordelen. Tevens zal, door het opstellen van richtlijnen voor de user-interface, het projectteam en de technische bouwleiding in staat zijn om de afstemming en éénduidigheid binnen de applicatie te waarborgen.

Literatuur

Sommerville, I., 1988, *Software Engineering; het bouwen van grote programma's*, 2e herziene druk, Academic Service, Schoonhoven.

ing. R.F.I. van Himste is werkzaam bij de sectie management-onderzoek van de afdeling bedrijfssynthese van het Proefstation voor de Akkerbouw en de Groenteteelt in de Vollegrond te Lelystad. ir P.C. de Jong is werkzaam als projectbegeleider teeltbegeleidingssystemen bij de Stichting Informatieverzorging Akkerbouw te Lelystad. □