

Zwemmen in een slootje of in de oceaan

Enkele persoonlijke ervaringen met user-interfaces in het HAO

ir. J.M.F.M. van Tartwijk

Docent veehouderij – Internationale Agrarische Hogeschool 'Larenstein'

Postbus 7, 7400 AA Deventer

telefoon: 0570-684800, telefax: 0570-684608

e-mail: j.m.f.m.van.tartwijk@iahdev.agro.nl

Referaat

Dit artikel beperkt zich tot de toegepaste veehouderij softwarepakketten die op de diverse Agrarische Hogescholen worden gebruikt. Voor een niet onaanzienlijk gedeelte van deze pakketten geldt dat het bewerkingen zijn van programma's die elders al gebruikt werden. De onderwijskundige doelstelling (wat wil ik bereiken met het vak en hoe faciliteert een softwarepakket daarin) is bij die ontwikkeling wel eens te veel op de achtergrond geraakt. Dit heeft in een aantal gevallen ertoe geleid dat programma's onvoldoende zijn toegesneden op gebruik in het onderwijs. Het artikel geeft aan dat het onderwijs een specifieke gebruikersgroep vormt die ook specifieke eisen stelt aan programmatuur. De user-interface bepaalt voor een aanzienlijk deel het succes van een programma binnen het HAO omdat bij de inrichting daarvan keuzes gemaakt moeten worden. Die keuzes dienen gemaakt te worden vanuit onderwijskundige doelstellingen.

Inleiding

De afgelopen jaren zijn voor het HAO al heel wat softwareproducten ontwikkeld. Als docent veehouderij gebruik je dan meestal de toegepaste pakketten als je ze zo mag noemen. Pakketten die ondersteunend zijn voor onderwijs in deelgebieden van het veehouderijvakgebied. Voorbeelden van dergelijke pakketten zijn o.a.: Koefok, Leren analyseren rundvee, Leren analyseren varkens, Leren analyseren pluimvee, Simulatiespel melkveehouderij en milieu, Voeding van het melkgevend rund, Technisch Model Varkensvoeding, Saldo- begrotings- en financieringsprogramma varkenshouderij, Saldo- begrotings- en financieringsprogramma melkveehouderij, Milieuzorgsystemen in de rundvee, en varkenshouderij. Het is geen volledige lijst maar een willekeurige greep uit een toch wel groot aantal pakketten van uiteenlopende aard. Niet alleen verschillend voor wat betreft inhoud (rekenpakketten of simulatiepakketten) maar ook voor wat betreft succes. Met succes bedoel ik dan de mate waarin de pakketten daadwerkelijk in het onderwijsprogramma gebruikt worden en wat ze te bieden hebben.

Succes

Het succes wordt naar mijn smaak voornamelijk bepaald door de meerwaarde die een softwarepakket biedt boven de meer traditionele overdrachtsvormen als mondelinge presentatie en schriftelijk

materiaal. Die meerwaarde zit natuurlijk in de volgende twee zaken:

- softwarepakketten: die als rekenmachines het domme rekenwerk overnemen van de student en zo meer tijd vrijmaken voor interpretatie van cijfers;
- softwarepakketten: die gevolgen van beslissingen doorrekenen (simulatiepakketten dus) en zo de student inzicht verschaffen in de samenhang van verschillende factoren, die op een bepaalde zaak inwerken.

Alle eerder genoemde softwarepakketten bieden deze meerwaarde maar toch zijn niet alle pakketten een succes geworden binnen het HAO.

Een mooi voorbeeld is misschien het pakket Koefok; een van de eerste simulatiepakketten die zijn intrede deed in het HAO. Koefok simuleert een aantal melkveebedrijven. Iedere student (of groep studenten) krijgt een bedrijf met het daarbij behorende jongvee en melkvee. Koefok kent van iedere koe haar genetische aanleg en de milieufactoren die die aanleg versluieren. Het programma presenteert de geschatte fokwaarden van iedere koe aan de gebruiker die op basis daarvan de koeien selecteert voor het volgende jaar (de volgende run van het programma). Een prachtig programma zou je zeggen waar je ook nog een competitieelement in kunt bouwen zodat het extra aantrekkelijk wordt voor studenten. Toch zijn we in Deventer na een aantal semesters gestopt met het gebruik van dit programma bij de lessen populatiegenetica.

Hiervoor waren meerdere redenen maar de belangrijkste was toch wel dat het aantal runs wat binnen het tijdsbestek van een lesperiode gemaakt kon worden te klein was. Fokkerij is, zoals bekend, een zaak van lange adem. De effecten van verschillende selectiestrategieën zie je pas na een groot aantal jaren (een groot aantal runs). Het rekengedeelte van het programma was dermate realistisch dat dit effect niet zichtbaar werd binnen een onderwijsperiode. Wat is het gevolg? Op een gegeven moment merk je dat je als docent meer uitleg aan het geven bent over de werking en rekenachtergronden van het programma dan over populatiegenetica. Natuurlijk zouden de resultaten van het programma centraal moeten staan, deze moeten de wetmatigheden van de populatiegenetica illustreren. Hiermee kan aan de lessen een meerwaarde worden gegeven.

Wat wil ik duidelijk maken met dit verhaal? De onderwijskundige doelstelling staat binnen het HAO altijd centraal. Ieder softwarepakket moet ten dienste staan van die doelstelling en vervolgens een meerwaarde bieden anders heb je er meer last van dan gemak.

Onderwijskundige doelstelling

Bij de ontwikkeling van softwarepakketten werd meestal nogal snel over die onderwijskundige doelstelling heengestapt. Bij Koefok was het bijvoorbeeld zo dat er "een geweldig mooi rekenmodel beschikbaar was dat ook prima binnen het HAO gebruikt zou kunnen worden". Dit was de belangrijkste motivatie om de aanpassing (!) van dit rekenmodel ter hand te nemen. Het resultaat: een redelijk bruikbare schil om een wetenschappelijk rekenmodel. Over vragen zoals; welke genetische wetmatigheden moet het programma kunnen illustreren is nauwelijks nagedacht. Gelukkig is het nadenken over dit soort vragen sterk verbeterd. Het gebeurt o.a. in de vergaderingen van de paraplugroep Zoötechniek. Om verder te gaan; wat voor de inhoud van het programma geldt, geldt ook voor de schil: de user-interface. De onderwijskundige doelstelling van ieder vak of vakonderdeel moet ook hier bepalend zijn.

Vakken verdringen elkaar

Landbouwopleidingen zijn per definitie brede opleidingen. Een direct gevolg is dat de vakken zich verdringen om een plaatsje te verkrijgen binnen het opleidingsprogramma. De bezuinigingen met als gevolg het verminderde aantal contacturen dat beschikbaar is, versterkt dit alleen nog maar. Een uitgelezen kans dus voor softwarepakketten om de zelfwerkzaamheid van studenten te vergroten en zo de aanspraak op de kleine berg contacturen te ontlasten. Helemaal waar, maar ook dit onderstreept nog eens de noodzaak van pakketten die zijn 'toegesneden' op het onderwijsprogramma, zowel voor wat betreft inhoud, als voor wat betreft de user-interface. Want veel tijd om pakketten toe te lichten is er niet.

De user-interface

Voordat nagedacht kan worden over de user-interface moeten eerst twee vragen worden beantwoord, die betrekking hebben op wat ik maar de kern van het programma noem.

- welke faciliteiten biedt het programma;
- welke eindresultaten worden per faciliteit verkregen.

Deze twee zaken moeten glashelder zijn en geformuleerd in het licht van de doelstellingen van het betreffende onderwijs-element. Dus: welke doelstellingen heeft het betreffende onderwijs-element in het onderwijsprogramma van een Agrarische Hogeschool en hoe kan een softwarepakket hierin faciliteren.

Als dit duidelijk is dan kom je bij de user-interface. Hieraan kun je naar mijn smaak de volgende eisen stellen. De gebruiker:

1. kan in één oogopslag zien welke faciliteiten het programma biedt;
2. kan eenvoudig door het programma heen wandelen;
3. weet op elk moment waar hij/zij zich in het programma bevindt;
4. weet op elk moment wat hij/zij doet en waarom;
5. krijgt gerichte informatie voorgeschoteld;
6. kan gericht informatie op papier laten uitdraaien;
7. kan zonder handleiding met een korte

toelichting direct met het programma 'aan de slag'.

Een voorbeeld: TMV

Aan de hand van het programma Technisch Model Varkensvoeding (TMV) wil ik het een en ander toelichten. Dit programma is evenals Koefok een wetenschappelijk rekenmodel waar een schil, geschikt voor gebruik in het HAO, omheen gebouwd is. TMV is in mijn ogen daarentegen wel een succes.

De faciliteiten in één oogopslag

Een sterk punt van TMV dat het gehele programma in zijn volle structuur op één scherm staat weergegeven. Op de volgende pagina is dat scherm te zien.

De bovenste helft van het scherm heeft te maken met de aansturing, de onderste helft voor de uitvoergegevens. De bovenste helft is logisch opgebouwd: aan de linkerkant een aantal knoppen waar je uitganggegevens in kunt vullen, aan de rechterkant de knoppen waarmee je keuzen ten aanzien van de simulatie kunt invullen. Tussen deze beide rijen knoppen worden de belangrijkste keuzen t.a.v. de simulatie weergegeven. Het sterke van dit scherm is dat alles steeds in beeld is, terwijl toch het scherm overzichtelijk is gebleven en niet te vol is geworden. Als een berekening gemaakt is, kun je bij het lezen van de resultaten (in de onderste helft) steeds zien bij welke simulatiekeuze deze uitvoer hoort (het middengedeelte van de bovenste helft).

Een detail t.a.v. dit scherm: de pijltjes aan de linkerkant horen bij het bladeren in de uitvoer. Hun plaats op het scherm had iets lager kunnen zijn zodat dit duidelijker was geweest.

Bedieningsgemak

Dit gemak is belangrijk vanwege het genoemde tijdsaspect. Bediening door middel van het aanklikken van knoppen met de muis is tegenwoordig vrij algemeen en erg gemakkelijk. In TMV zijn de knoppen overzichtelijk aangebracht en maakt de tekst bij iedere knop de gebruiker voldoende duidelijk waar hij/zij terecht komt.

Waar bevind ik me in het programma?

Het programma TMV gaat steeds slechts één niveau diep zodat alles erg overzichtelijk blijft. De onderliggende pagina komt bij aanklikken zodanig over het hoofdscherm heen dat je deze laatste steeds ziet liggen. Verlaat je het onderliggende scherm dan zit je direct weer in het eerder afgebeelde hoofdscherm.

Bij sommige programma's is dit laatste niet altijd mogelijk. Een goed ander alternatief is dan om het programma te doorlopen als de pagina's uit een boek. Onder op het scherm kan dan steeds een aanduiding zichtbaar zijn op welke pagina van het grotere geheel je zit. Een goed voorbeeld van een programma's waar dit laatste wordt geïmplementeerd zijn de pakketten SBF-varken en SBF-rund (Saldo, Begrotings- en Financieringsprogramma). Voor wat betreft onderliggende schermen moet een duidelijke beperking gelden: één niveau diep is diep genoeg. Zou je als ontwikkelaar nog dieper willen dan heb je mijns inziens in onvoldoende mate gekozen tussen wat je wel of niet wilt met het programma.

Waar ben ik mee bezig?

Deze vraag hangt samen met de vorige maar bevat ook een aantal nieuwe aspecten.

In het HAO worden nog steeds een aantal pakketten gebruikt die afkomstig zijn uit de onderzoeksweld. Ik denk dan vooral aan de economisch getinte pakketten. Het is niet ongebruikelijk dat een gebruiker plotseling een scherm voor zich krijgt waar 'even nog wat aanvullende gegevens moeten worden ingevuld' omdat anders de berekeningen niet gedaan kunnen worden. Dit moet vermeden worden. Deze zaken zijn er ondermeer de oorzaak van dat je als docent meer over het programma moet uitleggen dan over de eigenlijke materie. Ook TMV is niet helemaal verschoond van zulke 'verrassingen'. Wanneer gekozen wordt voor STARTEN wordt de gebruiker plotseling gevraagd welke gegevens bij de uitvoer van grafieken op de betreffende assen geplaatst moeten worden.

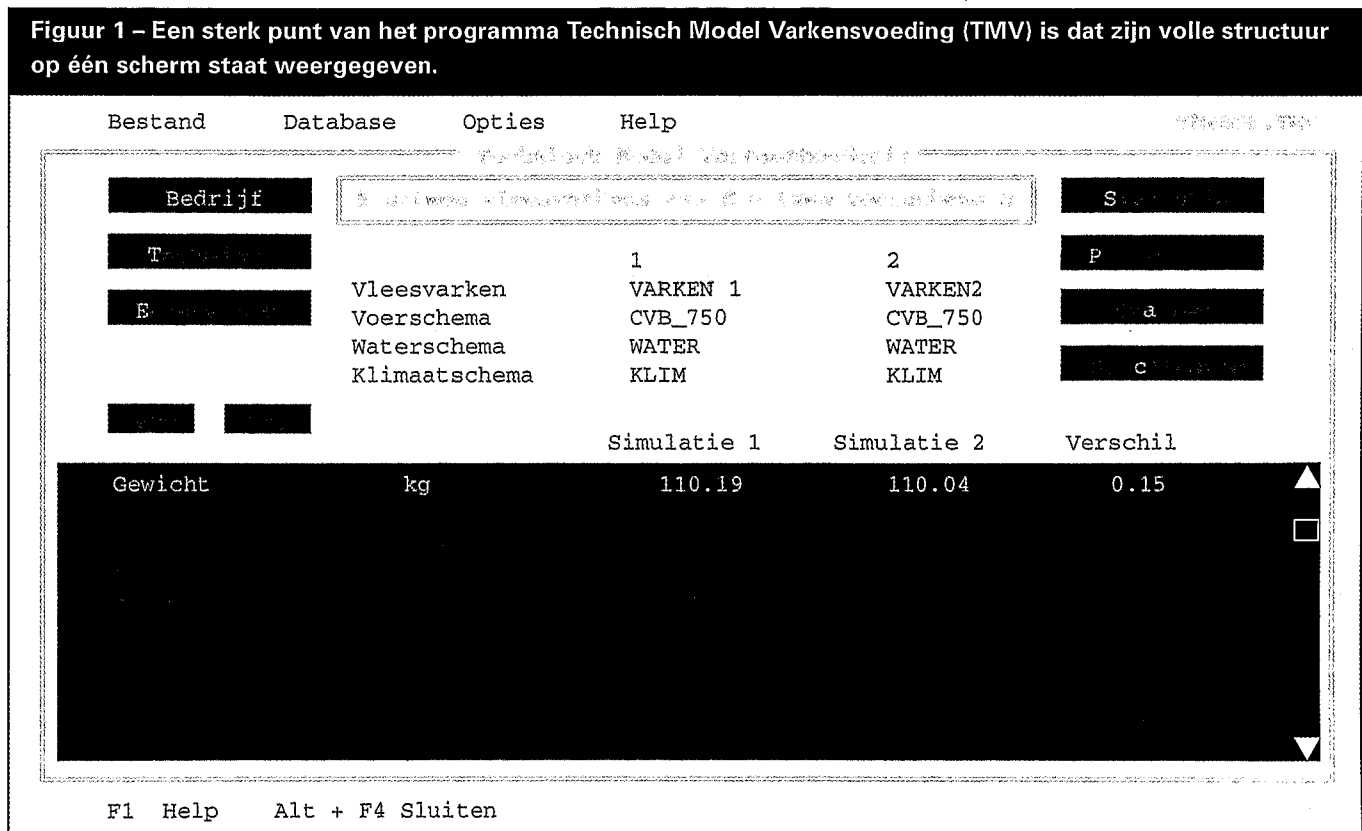
Eigenlijk is het ontstaan van dit scherm wel een stereotiep voorbeeld te noemen. Het wetenschappelijke rekenmodel van TMV produceert enorm veel uitvoer, één van de wijzen van presenteren zijn dan uiteraard grafieken. "Het is toch hartstikke leuk dat je in een grafiekje kunt zien in welke periode van het mesttraject lysine beperkend wordt" hoor ik een van de werkgroepleden nog zeggen (ik was ook werkgroep lid). "De gegevens zijn

voorhanden dus laten we het maar inbouwen". Achteraf bezien hadden we dit beter niet kunnen doen. De keuze voor grafieken is nu meegenomen in het 'grote pad' van het programma. Het pad dat je moet doorlopen van invoer tot uitvoer. Grafieken zijn een aardigheidje, ze kunnen erbij maar hadden meer naar 'de achtergrond' verplaatst moeten worden. Het verschijnen van dit scherm ontlokt nu de reactie: hé, wat is dit ?

Al bijna standaard is verder dat overal in het programma de HELP knop kan worden geraadpleegd. Een goede zaak maar als deze ook echt veel geraadpleegd worden geeft de gebruiker daarmee aan dat het programma niet deugt. Natuurlijk is dat wat zwart-wit gesteld en gaat dat in mindere mate op voor grote programma's die gebruikers op allerlei gebruikersniveau's tevreden moeten stellen.

Gerichte informatie voorschotelen

Een open deur lijkt het wel maar dit heeft weer alles te maken met de onderwijsdoelstelling. Ook hier is TMV een aardig voorbeeld omdat de outputstroom vrijwel onbeperkt leek. Kiezen dus. Mijns inziens is dit aardig opgelost door de gegevensstroom te rubriceren en per rubriek de meest passende (onderwijsdoelstelling)



gegevens te presenteren. Voorbeelden van rubrieken: groeigegevens, benuttingsgevens, mest- en mineralengegevens. Elke rubriek heeft maar één pagina (de helft van het hoofdscherm) en je kunt eenvoudig bladeren in de onderste helft.

Informatie op papier

Deze moet naar mijn smaak zoveel mogelijk gelijk zijn aan datgene wat je op het scherm krijgt. Uiteraard kan op papier iets meer gepresenteerd worden dan op het scherm maar het mag m.i. niet zo zijn dat de papier-output plotseling twee keer zo groot blijkt te zijn als dat wat je verwachtte. Als je in het programma bewust kiest tussen wat je wel en niet aanbiedt aan de gebruiker dan kan die lijn naar het papier worden doorgetrokken.

Direct aan de slag

De programmeeromgeving is tegenwoordig zodanig dat gebruikers vrijwel direct met een pakket aan de slag kunnen. Zoals duidelijk moge zijn geworden moet het programma m.i. zo helder opgebouwd zijn dat hij/zij niet alleen direct aan de slag kan maar ook de weg niet kwijtraakt. TMV voldoet hier wat mij betreft voldoende aan. Een uitleg om nieuwe gebruikers in TMV wegwijs te maken kan in 15 minuten afdoende worden afgehandeld. Dat kwartiertje zou een mooi richtgetal kunnen zijn al realiseer ik me dat dat getal voor pakketten met uitgebreide mogelijkheden een scherp gestelde limiet is.

Jeugd is niet computer-freak, docenten ook niet

Onlangs zag ik een kop in de krant met bovenstaande strekking. Voor mij geldt min of meer hetzelfde en ik meen te mogen zeggen ook voor een groot

gedeelte van mijn collega's. Ze zitten nagenoeg allemaal regelmatig achter hun PC's maar het zijn docenten die hun vak doceren, al klinkt dat misschien wat ouderwets in deze tijd waar het onderwijsleerproces van de student centraal staat (geheel terecht overigens). Docenten zijn, vaak noodgedwongen, niet geneigd om allerlei aardige opties in een programma te verkennen; de grote lijn, daar gaat het om. Softwarepakketten moeten snel, zonder al te veel uitleg, ingepast kunnen worden in een lesprogramma en het belangrijkste: je moet er wat aan hebben.

Zwemmen in een slootje of in de oceaan

Het is een bekende kreet 'alles hangt met alles samen'. Ik hoop dat duidelijk is geworden dat hierin een gevaar schuilt bij het ontwikkelen van softwarepakketten. Vaak bestaat de neiging om te veel mogelijkheden in één en hetzelfde programma te stoppen. Beter is het om vooraf duidelijk te krijgen wat je nu precies wilt met het betreffende pakket in het onderwijs. Bij dat kiezen moet de onderwijsdoelstelling van het betreffende onderwijselement zoveel mogelijk centraal worden gesteld.

Het is bij de ontwikkeling van toegepaste softwarepakketten vaak 'mis' gegaan omdat veel van die pakketten bewerkingen zijn van wetenschappelijke modellen of in het onderzoek gebruikte pakketten. Het lijkt dan al gauw 'zonde' om veel van de mogelijkheden die dergelijke modellen bieden niet te gebruiken. Wat mij betreft daarom niet al te veel 'tab-bladen', 'ikonen-balken' en andere drukknoppen in het scherm, die van alles en nog wat mogelijk maken.

Uiteraard is er ook een keerzijde. Een heel strak gestructureerd programma is ook niet wenselijk. Er ligt ergens een optimum en waar dat optimum ligt wordt bepaald door wat je met het pakket wilt. Met de titel van dit stuk heb ik dit proberen aan te geven: wil je zwemmen in een slootje of in de oceaan? Het optimum zou misschien een rustig binnenmeertje kunnen zijn.

De één punt nul versie

Eén ding wil ik graag nog kwijt, ondanks dat het geen directe relatie heeft met user-interfaces. In het Hoger Agrarisch Onderwijs wordt voor wat betreft de toegepaste pakketten vrijwel uitsluitend gewerkt met de zogenoemde 1.0-versie. Softwareproducten worden op projectbasis ontwikkeld. Evaluatie van gebruikerservaringen vindt nagenoeg niet plaats. Aanpassing van suboptimale of verkeerde zaken in een pakket dus ook niet. Het resultaat is dat sommige softwareproducten al vrij snel uit de onderwijsprogramma's verdwijnen ondanks dat ze in potentie veel te bieden hebben. Er is dus nog veel werk aan de winkel. @