

GEbruikersvriendelijkheid

Druk op de muis of kat in de zak?

G.J. Hofstede

Tot tien jaar geleden waren computers het domein van specialisten. Gebruikers waren van een andere wereld. Van deze houding getuigt bijvoorbeeld de ironische verzuchting van T.P. Moran (1981): "A system does not, alas, terminate at its terminals; users are attached". Behalve het woord terminal toont ook het woord eindgebruiker dat gebruikers een eind van de systemen afstonden.

De komst van de PC en van veelzijdige, kneedbare programma's als tekstverwerkers, spreadsheets en databasepakketten hebben dit beeld gedurende het laatste decennium ingrijpend veranderd. Dit artikel gaat in op de huidige status van de computergebruiker en op het belang van gebruikersvriendelijkheid van computers. Dit belang is weliswaar groot maar kan overschat of misbruikt worden.

Gebruikers in beeld

Het belang van gebruikersvriendelijkheid wordt vandaag de dag wijd en zijd erkend in onderzoek, onderwijs en bij softwareproducenten. Ook VIAS gunt gebruikers een rol, getuige bijvoorbeeld dit themanummer maar ook het gastoptreden van de ergonoom Ted White op het VIAS-symposium 1990.

Onderzoek

Onderzoekers schrijven stapels artikelen en boeken over gebruikers. Pioniers waren bijvoorbeeld Ben Shneiderman, bekend geworden met zijn boek "Software Psychology" in 1980, en de hierboven geciteerde T.P. Moran met zijn voorstel voor het scheppen van een "Applied psychology of the user" in 1981. Sinds die tijd zijn complete onderzoeksafdelingen als paddestoelen uit de grond geschoten.

Cognitief psychologen voeren experimenten uit om de gevoeligheid en waardering van gebruikers voor allerlei interface-aspekten te bepalen. Om de proefopzet beheersbaar te houden gaat het hierbij doorgaans om laboratoriumexperimenten waarbij proefpersonen eenvoudige taken uitvoeren.

AI-specialisten modelleren menselijk denken. Sinds de tijd van de General Problem Solver van Newell en Simon (1972) zijn ze hierin een stuk minder ambitieus geworden. Communicatie op niveau tussen kennissysteem en gebruiker blijft een van de moeilijke punten. Stel bijvoorbeeld dat de gebruiker van een expertsysteem een toelichting wil bij het advies van het systeem. Juist het geven van begrijpelijke uitleg over het redeneerproces is voor een expertsysteem een moeilijke opgave.

OR-specialisten erkennen dat het zoeken en geschikt maken van slimme algoritmen niet de enige voorwaarde is

voor een effectieve beslissingsondersteuning. In de vorige Agro-Informatica schreef Hendriks (1990)

"De gebruikersvriendelijkheid van het user interface bepaalt, uiteraard naast de voorgestelde beslissingsvarianten en -adviezen, of het DSS zal worden geaccepteerd en toegepast."

Onderwijs

Bevindingen uit ontwerp praktijk en onderzoek vinden op grote schaal hun weg naar de leerboeken. Overigens is het onderscheid tussen onderzoek en onderwijs vaak klein. Gebruikersaspecten zijn een onderwerp van uiteenlopende informaticaboeken. Enkele voorbeelden binnen ons taalgebied:

Bots et al. (1990) behandelen in hun boek over BIK de deelname van gebruikers in automatiseringsprojecten. Bemelmanns (1987) geeft in zijn boek over BIK een checklist met aandachtspunten voor gebruikersvriendelijkheid. Hofstede (1990) geeft een soortgelijke lijst, maar met andere punten, in zijn boek over databases. Daarnaast zijn er aparte leerboeken aan gebruikersinterfaces gewijd, bijvoorbeeld dat van Willemse en Lindijer (1988). Deze auteurs noemen hun boek "Software Ergonomie" en geven daarmee aan dat dit aandachtsgebied volwaardig onderwerp van ontwerpinspanningen moet zijn. Ze presenteren, in begrijpelijke taal en ondersteund door vele diagrammen, een methode voor het ontwerpen van gebruikersinterfaces. Internationaal zijn de boeken van Shneiderman vermaard. Het laatste, "Designing the User Interface" (1987) bevat talrijke nuttige voorbeelden en adviezen.

In checklists vindt men een veelheid aan richtlijnen. Enkele daarvan keren overal terug, zoals:

- consistentie;
- eenvoud;
- feed-back;
- 'undo'-mogelijkheid.

Bedrijfsleven

Tenslotte, en dit is wat het meeste gewicht in de schaal legt, is software een wereldmarkt van enorme omvang geworden waarin gebruikersvriendelijkheid een belangrijk verkoopargument is. In reclamecampagnes vindt men dikwijls korte aanleertijd, efficiënt werken of plezierig werken als aangeprezen eigenschappen van hard- en software.

Het belang van gebruikersinterfaces blijkt ook uit het feit dat softwarebedrijven steeds vaker functionarissen aanstellen die speciaal zijn belast met gebruikersinterfaces.

Problemen met ontwerprichtlijnen

Hoewel er, zoals gemeld, al sinds enige jaren nogal wat handboeken en checklists verschijnen voor het ontwerpen

van gebruikersinterfaces, is de theoretische basis van veel richtlijnen nog omstreden. Ik zal bij wijze van voorbeeld enkele problemen belichten.

Taak en interface

De theorie achter interface-ontwerp is met name aangevraagd door cognitief psychologen. Een voorbeeld van een goed cognitief psychologisch experiment is dat van Gerstendörfer en Rohr (1987). Zij lieten gebruikers met verschillende cognitieve stijl (visueel, verbaal of formeel ingesteld) verschillende taken uitvoeren in verschillende weergaven. Het bleek dat de taak, en niet de cognitieve stijl van de gebruiker, bepaalde welke weergave geschikt was. Wanneer de weergave niet paste bij de taak liep het mis. Bijvoorbeeld zelfs een visueel ingesteld iemand kookte beter met een geschreven recept dan aan de hand van

taak	weergave		
	plaatjes	tekst	tabel
tafel dekken	++		
eten koken		++	
inkopen doen			++

foto's. Zie figuur 1 voor een indruk van dit experiment.

Figuur 1. Combinaties van taak en weergave bij Gerstendörfer (1987). ++: Geschikte combinaties van taak en weergave

Helaas spreken de vele honderden experimenten van dit type elkaar nogal tegen. Er zijn ook experimenten die juist wel een effect van cognitieve stijl vinden. Er zijn experimenten die gebruik van kleuren ondersteunen, er zijn er ook die het onbelangrijk achten. Er zijn experimenten die grafieken waarderen boven tabellen en andersom (zie bijvoorbeeld Dickson et al., 1977, Davis & Olson, 1985, Bullinger & Shackel, 1987, MCI 1988). Klaarblijkelijk spelen de experimentele taak en de keuze van proefpersonen een onvoorziene rol waardoor de resultaten van een experiment beïnvloed worden. Dit maakt het moeilijk om de bevindingen uit een dergelijk experiment door te trekken naar andere omstandigheden.

Tegen consistentie

Een provocerend artikel over problemen met richtlijnen is "The Case Against User Interface Consistency" (Grudin, 1989). Consistentie van een interface is dé heilige koe onder de richtlijnen voor gebruikersvriendelijkheid. Welke argumenten durft deze auteur van stal te halen om daar tegen in te gaan? Wel, hij onderscheidt allereerst drie typen consistentie, met verschillende eigenschappen:

- interne consistentie van een interface. Dit type is voor de ontwerpers van software het duidelijkst zichtbaar en krijgt dan ook verreweg de meeste aandacht;
- externe consistentie van het interface met interfaces van vergelijkbare systemen waarmee de gebruiker bekend is. Dit ligt al moeilijker, want lang niet altijd weet de ontwerper met welke vergelijkbare systemen de gebrui-

ker bekend is, laat staan dat de ontwerper de betreffende interfaces kent;

- overeenkomst van het interface met vertrouwde eigenschappen uit de wereld buiten computers. Ontwerpers maken dikwijls gebruik van metaforen of analogieën. Deze zijn vaak in het begin voor het aanleren erg makkelijk. Ze gaan echter per definitie ergens mank en dat kan leiden tot verwarring bij gevorderd gebruik. De gebruiker voelt niet aankomen wanneer de analogie niet meer zal opgaan.

Voor de gebruiker, betoogt Grudin vervolgens, zijn deze typen consistentie niet te onderscheiden. Hij geeft vele voorbeelden van misverstanden die hierdoor kunnen ontstaan. Consistentie blijkt bij een niet volkomen simpele taak al snel onhaalbaar. Grudin concludeert onder meer dat "(...) highest level task consideration can warrant overriding consistency at lower levels (...)". Ik onderschrijf deze conclusie. Met een eigen voorbeeld wil ik nog eens het belang aangeven van het onderscheiden van typen consistentie. Een bekend probleem voor PC-gebruikers is het verschil tussen "onder DOS werken" of "in een applicatiepakket zitten". Dit zijn voor de ervaren gebruiker twee verschillende systeemtoestanden, elk intern consistent, terwijl het op de beginnende gebruiker overkomt als een zich inconsistent gedragende computer. De oorzaak voor dit probleem ligt hierin dat de gebruiker geen zicht heeft op de interne structuur van software, en dus op de eerste vorm van consistentie volgens Grudin.

Mijn sympathie voor Grudins stellingen is nog aanmerkelijk toegenomen sinds ik een eerste, voltooide versie van dit artikel per abuis weggooidde doordat ik PC-Wordperfect-concepten op een VAX-tekstverwerker toepaste.

Lering

Welke lering valt er nu uit deze strijdigheden te trekken? Kennelijk is het bij experimentele toetsing van gebruikersinterfaces bijzonder hachelijk om de resultaten te extrapoleren naar de wereld buiten het experiment. De strijdigheden duiden er op dat er contextfactoren een rol spelen die niet te meten vallen maar die in hoge mate de experimentele resultaten beïnvloeden. Vooral de taakcontext lijkt een belangrijke rol te spelen, en omdat elke taak unieke aspecten heeft is het moeilijk daarover algemene uitspraken te doen. In het algemeen kan men dan ook stellen dat het bijzonder moeilijk is om in het algemeen iets te stellen over gebruikersinterfaces.

Wat het onderzoek in elk geval heeft opgeleverd is het inzicht dat het gebruikersinterface van invloed is op het aanleergemak. Heckel (1984) sneert in dit verband "A software manual is a designer's list of failures." Volgens Heckel moet het interface dus zichzelf uitleggen. De praktijk stelt daar tegenover dat binnen zekere grenzen met elk interface een taak nog wel te verrichten is door iemand die het betreffende interface geleerd heeft. Mensen blijven vaak liever een slecht pakket gebruiken dan dat ze een nieuw pakket moeten leren. Wanneer iemand eenmaal een interface beheerst gaat het die gebruiker om wat hij met het systeem kan doen en vallen hem de gebreken aan het interface niet meer op (zie bijvoorbeeld Jörgensen, 1987).

Aspekten van interactie

Op dit punt lijkt het me nuttig een verfijning aan te brengen in het begrip gebruikersvriendelijkheid. Een mens-computer interactie kan bekeken worden door verschillende brillen. Een gangbare indeling (zie bijvoorbeeld Hofstede 1990) onderscheidt de volgende vier aspecten:

- pragmatisch: welke taken voert de gebruiker uit?;
- semantisch: welke begrippen spelen een rol?;
- syntactisch: welke signalen worden uitgewisseld, hoe zijn ze gecodeerd?;
- fysiek: welke media worden gebruikt?

Gebruikersvriendelijkheid kan op elk van deze aspecten betrekking hebben. Softwareleveranciers benadrukken, naast geldelijk gewin door gebruik van hun systeem, veelal de laatste twee aspecten. Ook ontwerpers hebben de neiging hun systeem zo gelikt mogelijk te maken. Mooie schermen of een soepele bediening van een pakket zijn inderdaad van belang. Ze kunnen de leertijd bekorten en het werken veraangenamen. Er zijn ook wel formalismen ontwikkeld om een gebruikersinterface op deze aspecten te beoordelen (Kieras & Polson, 1985). Voor de gebruiker gaat het echter uiteindelijk om het pragmatische aspect: wat kan ik met het systeem. Daarvoor is het semantische aspect van belang: de gebruiker moet de begrippen begrijpen die in het systeem een rol spelen. Syntactische en fysieke aspecten zijn voor de gebruiker in feite minder belangrijk, al moeten ze natuurlijk wel 'even' aangeleerd worden.

Met name de belangrijke pragmatische en semantische aspecten van gebruikersvriendelijkheid worden nogal eens onderbelicht in ontwerprichtlijnen. Ze zijn dan ook moeilijk te ontwerpen en te controleren. Dit geldt met name wanneer de functies van een systeem minder eenvoudig zijn. Ontwerpbeslissingen over taakontleding, bijvoorbeeld, grijpen diep in in de werking van het systeem. Een ongelukkige woordkeus kan tot ernstige fouten van gebruikers leiden.

Het beste wapen tegen deze gevaren is participatief ontwerpen, bijvoorbeeld met behulp van prototyping. Ervaren ontwerpers zeggen dit ook, bijvoorbeeld Gould (1987): "Principle 1: Early and continual focus on users". Hierbij moeten dan ontwerpbeslissingen zo vroeg en expliciet mogelijk aan de gebruikers worden voorgelegd. Misverstanden op pragmatische en semantische gebied komen dan tijdig aan het licht.

Discussie

Schermen met de gebruiker

Gegeven dat er nog veel onzekerheid is over wat nu feitelijk een goed gebruikersinterface is worden de pretenties van softwareleveranciers, maar ook van handboeken-schrijvers, wat twijfelachtig. Hoe belangrijk zijn nu werkelijk ontwerprichtlijnen? Hoe weet je of een gebruikersinterface werkelijk goed is? En hoe belangrijk is het nu eigenlijk dat een gebruikersinterface goed is? Deze vragen zijn stuk voor stuk problematisch. En hoewel buiten kijf staat dat gebruikersvriendelijkheid van essentieel belang is moet men gezien deze onzekerheden de aanbevelingen en aanprijzingen toch met een schep zout nemen. De belangstelling voor de gebruiker doet soms even schijn-

heilig aan als de plotseling alom gepredikte bezorgdheid om het milieu. Het is goed te blijven beseffen dat gebruikersvriendelijkheid niet meer dan één van de aspecten is die de kwaliteit van informatiesystemen bepalen.

Oplossingen

Hoe kan een software-ontwerper, ondanks de onzekerheid van ontwerprichtlijnen, nu toch een goed gebruikersinterface afleveren? Wel, uiteraard doen ontwerpers er goed aan, kennis te nemen van de bestaande richtlijnen. Mits kritisch toegepast kan dat de kwaliteit zeker ten goede komen. Toekomstig psychologisch onderzoek zal ook ongetwijfeld meer inzichten over goede interfaces opleveren. Een pasklare oplossing verwacht ik hier echter niet van, omdat nu eenmaal de context waarbinnen de taak valt zo belangrijk zijn bij het uitmaken wat een goed interface is.

Meer heil valt te verwachten van verbeteren van het ontwerpproces, met name door participatief ontwerpen. Voor het benadrukken van gebruikersinterface-kwaliteit tijdens het ontwerp bestaat inmiddels een naam: Usability Engineering. Nielsen en Molich (1989) geven onder deze vlag "Five golden rules" voor interface ontwerp:

- ken de gebruiker;
- betrek de gebruiker bij het ontwerp;
- coördineer het gehele user interface;
- verricht empirische metingen aan het ontwerp (met gebruikers);
- ontwerp iteratief.

Toepassen van deze richtlijnen zal onvermijdelijk een zekere mate van onvoorspelbaarheid van het ontwikkelproces met zich mee brengen. Dat wijst er al op dat ook flexibiliteit en inventiviteit belangrijk zijn. Niet voor niets rangschikt Heckel (1984) software-ontwerpers naast beoefenaars van andere creatieve beroepen als architect en kunstenaar.

Essentieel is dat de gebruiker, en niet de ontwerper, deskundig is over zijn eigen taken. Een anonieme gebruiker, geciteerd in Björn-andersen (1988) merkt hierover op: "Competence about the user problem is something that the manufacturers think users do not have and that users of the system later discover that the manufacturers did not have". Oprecht toepassen van de "five golden rules" kan dit gevaar wegnemen.

Tot slot

De huidige belangstelling voor gebruikersvriendelijkheid is terecht. Er wordt echter relatief veel aandacht besteed aan aspecten van gebruikersvriendelijkheid die uiteindelijk niet zo belangrijk zijn. Daarnaast is er nog veel onzekerheid over wat nu wel en niet gebruikersvriendelijk is, zodat een ontwerper niet blind kan varen op a priori richtlijnen en een potentiële koper niet op aanprijzingen van producten. Samenwerking tussen ontwerpers en (potentiële) gebruikers tijdens het ontwerp, zodat de ontwerper de context van het gebruik leert kennen en misvattingen bijtijds worden bijgesteld, lijkt voorlopig van veel meer belang.

Het maakt niet meer uit of een druk op de muis voldoende is, als er een kat in de zak blijkt te zitten.

Literatuur

- Bemelmans, T.M.A.**, 1987, "*Bestuurlijke informatiesystemen en automatisering*", derde druk, Stenfert Kroese, Leiden 1987.
- Björn-Andersen, N.**, 1988, "*Are 'Human Factors' human?*" in *The Computer Journal*, 31:5, 1988, pp. 386-390.
- Bots, J.M., et al.**, 1990, "*Bestuurlijke Informatiekunde*", Cap Gemini Publishing / Pandata, Rijswijk 1990.
- Bullinger, H.J. & B. Shackel** (eds), 1987, *INTERACT '87*, Proc. Conf. Human-Computer Interaction, Elsevier/IFIP 1987.
- Davis, G.B. & M.H. Olson**, 1985, "*Management Information Systems*", Mc Graw-Hill, Singapore 1985.
- Dickson, G.W., J.A. Senn, N.L. Chervany**, 1977, "*Research in Management Information Systems: The Minnesota Experiments*", *Management Science* 23:9, mei 1977, pp. 913-923.
- Gerstendörfer, M. & G. Rohr**, 1987, "*Which task in which representation on what kind of interface*", in H.J. Bullinger & B. Shackel (eds), *INTERACT '87*, Proc. Conf. Human-Computer Interaction, Elsevier/IFIP 1987, pp. 513-518.
- Gould, J.D.**, 1987, "*How to design usable systems*", in H.J. Bullinger & B. Shackel (eds), *INTERACT '87*, Proc. Conf. Human-Computer Interaction, Elsevier/IFIP 1987.
- Grudin, J.**, 1989, "*The Case Against User Interface Consistency*", *Communications of the ACM* 32:10, pp. 1164-1173.
- Heckel, P.**, 1984, "*The elements of friendly software design*", Warner, New York, 1984.
- Hendriks, Th.H.B.**, 1990, "*Operationele Research*", *Agro-Informatica* 3:2, april 1990, pp. 7-12.
- Hofstede, G.J.**, 1990, "*Datamodel en Database*", Coutinho, Muiderberg 1990.
- Jørgensen, A.H.**, 1987, "*The trouble with UNIX MAIL: Initial learning and experts' strategies*", in: H.J. Bullinger & B. Shackel (eds), *INTERACT '87*, Proc. Conf. Human-Computer Interaction, Elsevier/IFIP, Amsterdam, 1987, pp. 847-854.
- Kieras, D. & P. Polson**, 1985, "*An approach to the formal analysis of user complexity*", *Int. J. of Man-Machine Studies* 22, pp. 365-394.
- MCI**, 1988, Samenvattingen van lezingen op conf. "*Systemen naar menselijke maat: feit of fictie?*" Eindhoven, 1988, NGI werkgroep Mens-Computer Interactie.
- Moran, T.P.**, 1981, "*An applied Psychology of the User*", *ACM Computing Surveys* 13:1, pp 1-11.
- Newell, A. & H.A. Simon**, 1972, "*Human Problem Solving*", Prentice-Hall, Englewood Cliffs, N.J., 1972.
- Nielsen, J. & R. Molich**, 1989, "*Teaching User Interface Design Based on Usability Engineering*", *ACM SIGCHI bulletin*, 21:1, pp. 45-48.
- Shneiderman, B.**, 1980, "*Software Psychology*", Winthrop Pub. Inc., Cambridge 1980.
- Shneiderman, B.**, 1987, "*Designing the User Interface*", Addison-Wesley, Reading 1987.
- Willemse, H. & G. Lindijer**, 1988, "*Software Ergonomie*", Academic Service, Schoonhoven 1988. □

ir. G.J. Hofstede is sinds 1985 verbonden aan de vakgroep Informatica van de Landbouwwuniversiteit waar hij zich bezig houdt met databases en beslissingsondersteunende systemen.