George M. Welling

Digitale Bronnen-transcripties: wie gaat ze invoeren?

'Komt het ooit nog goed tussen historici en de computer?' vroeg George M. Welling zich in 1996 af (*Groniek* 135). De laatste jaren is het computer- en ook het internet-gebruik in rap tempo toegenomen, ook onder historici. In dit artikel bespreekt Welling de voorwaarden en de aandachtspunten voor digitalisering van bronnen en gegevens, aan de hand van zijn eigen ervaringen.

De revolutie die de makkelijke toegang tot het Internet via het World Wide Web gebracht heeft, is ook niet aan de historische wereld voorbij gegaan. Hoewel hier sprake is van een groep, die traditioneel niet zo voorop loopt in het toepassen van nieuwe technologieën, zien we nu een snelle toename van websites die zich richten op historische themata. Genootschappen, verenigingen, archieven, noem maar op, iedereen probeert zo snel mogelijk een kaveltje op het Web af te bakenen. Naast het beschikbaar komen van een grote hoeveelheid nuttige en een nog grotere hoeveelheid onnutte informatie heeft dit proces een tweetal interessante neveneffecten.

Ten eerste heeft bovengenoemde revolutie ervoor gezorgd, dat de negatieve houding ten opzichte van informatisering bij de meeste historici sterk is afgenomen. Vrijwel iedereen erkent het gemak van on-line informatie. Sterker nog: er is duidelijk sprake van een vraag naar een grotere beschikbaarheid van meer en betere informatie. Bij iedere instelling die zich op het Web begeeft, dient men zich te realiseren dat na verloop van tijd een lijstje met links of een korte beschrijving van inventarissen de gebruiker niet lang zullen bevredigen.

Als de eerste verbazing over de mogelijkheden van het medium voorbij is, komt vanzelf weer de kritische kijk op alle vormen van publicaties boven. Dat zal dan voor veel websites negatief uitvallen, omdat die eigenlijk niet veel verder gaan dan heel hard op het internet roepen: 'hier ben ik',

hetgeen snel gaat vervelen. Dan ontstaat er het conflict tussen de belangen van de instellingen en de gebruikers van hun websites. Instellingen willen via hun websites mensen naar zich toe lokken en daartoe laten ze lijsten zien van wat ze in huis hebben. Voor de stukken zelf dien je jezelf alsnog fysiek te verplaatsen. Een belangrijke reden hiervoor is, dat vele instellingen afhankelijk zijn van subsidie die gerelateerd is aan aantallen bezoekers. Virtuele bezoekers worden daarbij nog steeds niet meegerekend!

Gebruikers willen echter steeds meer: als het technisch mogelijk is om inventarissen te vertonen op het web, dan kun je natuurlijk ook de stukken zelf vertonen. Meta-informatie over beschikbare informatie is heel fijn, maar het is 'not the real thing'. Veel instellingen beschikken echter niet over de kennis en mankracht om aan deze wens te voldoen. Het digitaliseren van historische informatie staat eigenlijk nog steeds in zijn kinderschoenen en veelal missen stafleden van historische instellingen de opleiding ervoor.

Een tweede effect van de informatisering van de historische wereld is eigenlijk zeer positief. Het internet heeft het aantal computergebruikende historici zo doen groeien, dat het een interessant marktsegment is geworden. Er verschijnen steeds meer elektronische publicaties in CD-ROM vorm, die nadrukkelijk gericht zijn op historici. Om maar een paar voorbeelden te noemen: het CBS heeft in samenwerking met het NIWI[1] een set van vijf cd-rom's met de gegevens van de volkstellingen uitgebracht. Uitgeverij Het Spectrum heeft een aantal CD-ROM's met historische onderwerpen het licht doen zien. En er is een groot aantal spellen op de markt die historisch getint zijn en misschien wel meer enthousiasme voor geschiedenis bij kinderen opwekken dan hun lessen op school.

Geen reden tot klagen, zou je dus bijna concluderen. Toch lijken een paar kritische noten hier op hun plaats. Bij een nadere beschouwing blijkt veel van hetgeen er aangeboden wordt nauwelijks meerwaarde boven een gedrukte versie te bieden, of niet veel meer te zijn dan een tekst die toch al in digitale versie beschikbaar was. Veel uitgeverijen hebben immers hun publicaties van de laatste jaren toch al in digitale vorm en waarom zouden ze na de gedrukte versie niet ook nog eens een digitale versie uitbrengen. Een beperkte zoekfunctie erbij en klaar is Kees.

1 NIWI=Nederlands Instituut voor Wetenschappelijke Informatiediensten (*http://www.niwi.knaw.nl*) - Onderdeel van de Koninklijke Nederlandse Academie van Wetenschappen (KNAW).

Opvallend is dat de meeste CD-ROM's zich hoofdzakelijk concentreren op beeld en tekstmateriaal. De reden is eigenlijk tamelijk triviaal. Zoals al eerder gezegd is het tekstmateriaal voor een groot deel al digitaal beschikbaar. Beeldmateriaal digitaliseren vereist enige technische kennis, maar met de moderne scanners is het mogelijk een grote hoeveelheid afbeeldingen of documenten snel op te slaan in digitaal formaat. Voor afbeeldingen is dat eigenlijk geen bezwaar, maar voor documenten levert dat eigenlijk weinig meerwaarde op boven een *facsimile* druk van het document. De CBS-uitgave van de volkstellingen is een treurig voorbeeld van wat ik bedoel: vijf CD-ROM's vol met afbeeldingen van pagina's van de volkstellingen met daarbij een eenvoudig zoekprogramma. Indien je analyses wilt maken op basis van deze gegevens, zul je ze alsnog zelf in een database of spreadsheet moeten invoeren. In de meeste gevallen zijn de pagina's wel zo goed gescand, dat je met behulp van een OCR-programma[2] de afbeeldingen naar tekst en cijfers zou kunnen omzetten: er zijn echter een flink aantal gescande pagina's die hiervoor kwalitatief onvoldoende zijn. En als het materiaal eigenlijk alleen als afbeelding bruikbaar is, had men net zo goed een ander formaat kunnen kiezen, zodat met behulp van enige compressie de afbeeldingen allemaal zonder noemenswaardig kwaliteitsverlies op één cd-rom hadden gepast.

Datagericht of probleemgericht?

Waar ligt dan de kern van het probleem? Eigenlijk is het antwoord op deze vraag nogal platvloers: arbeidskosten. Fotoreproductie en het aanbieden van reeds in digitale vorm aanwezig materiaal is goedkoop. Het vraagt niet al te veel investering van hoog gekwalificeerde arbeid. Het corrigeren van digitale bestanden,die via OCR tot stand gekomen zijn, is als het om origineel drukwerk gaat nog wel uitvoerbaar voor laag gekwalificeerde arbeidskrachten. De problemen beginnen eigenlijk pas echt bij de productie van digitale bestanden op basis van handschriften of grote hoeveelheden cijfermateriaal.

Hoewel er enige vooruitgang geboekt is met de elektronische herkenning van handschriften, zijn de resultaten nog niet zo goed dat dit op grote schaal toegepast kan worden. De foutmarges zijn zo groot, dat het handmatig verwerken een aanzienlijk grotere betrouwbaarheid oplevert. Maar dat vergt hoog gekwalificeerde arbeid, omdat het herkennen van hand-

2 OCR=Optical Character Recognition, tekstherkenning na scannen.

schriften, vooral oudere handschriften, een gedegen paleografische kennis vereist. Het produceren van grote historische digitale tekstcorpora wordt hierdoor zo duur, dat een rendabele exploitatie vrijwel onmogelijk wordt.

Zodoende beschikken we nu in digitale vorm hoofdzakelijk over teksten die al in druk aanwezig waren: die zijn immers makkelijk te scannen. Verder zijn er toevallige bijproducten van historisch onderzoek, zoals die bijvoorbeeld bij het NHDA/NIWI[3] gedeponeerd zijn. Vele van deze bestanden lijden echter aan structurele tekortkomingen, waardoor ze nauwelijks geschikt zijn voor hergebruik in nieuw onderzoek, omdat ze 'probleemgericht' tot stand zijn gekomen. Om redenen van efficiëntie zijn alleen de variabelen opgenomen die voor het onderzoek belangrijk waren. Een 'datagerichte' aanpak, waarbij bronnen integraal worden gedigitaliseerd, is vanwege het extra werk dat er aan vast zit nog steeds niet erg gebruikelijk. Veel bestanden zijn immers het bijproduct van onderzoek dat onder hoge tijdsdruk en met beperkte middelen verricht moest worden.

Een goed voorbeeld hiervan is het door Lindblad beschikbaar gestelde bestand *Dutch entries in the pound-toll registers of Elbing, 1585-1700 - Elbing*[4], dat alleen de gevallen bevat die een relatie met de Republiek hadden, hetzij door de herkomst van de schip of de schipper, hetzij door eigendom van het schip. Een dergelijke aanpak sluit een vergelijking met andere nationaliteiten bijvoorbeeld al direct uit. Een recente vergelijking van de gegevens uit deze registers met ander bronnenmateriaal leidt dan ook tot een niet geheel bevredigend resultaat[5]: sommige schepen komen in het ene register wel voor en niet in het andere en het is nu niet meer te achterhalen of de registers lacunes vertonen, of dat er misschien gegevens ontbreken in de dataset. En hoewel ik Lindblad als historicus hoog acht, zou ik toch liever de hele registers gehad hebben om te zien of hij misschien niet toch iets over het hoofd gezien heeft.

Het lijkt een min of meer uitzichtloze situatie. De markt voor digitale Nederlandse historische bronnen voor professioneel gebruik is zo klein, dat er commercieel weinig te halen valt. Grote digitale gegevensbestanden

3 NHDA= Nederlands Historisch Data Archief (*http://www.niwi.knaw.nl/nl/dd_nhda/dd_nhda.htm*).

4 De URL van dit bestand is zo lang (het is het resultaat van een elektronische zoektocht), dat het beter is de URL van het NHDA/NIWI hier te geven: *http://www.niwi.knaw.nl/nl/dd_nhda/dd_nhda.htm.*

5 Tonko Ufkes, 'Vier registers vergeleken over de jaren 1654 en 1655. Een verkenning..', *Tijdschrift voor Zeegeschiedenis* 19 nr. 1 (april 2000) 1-17.

over de Nederlandse geschiedenis zijn er dan ook niet te verwachten van de Nederlandse uitgevers. De bestanden die wel beschikbaar zullen komen, betreffen over het algemeen bronnenmateriaal dat al gebruikt is voor historisch onderzoek. De makers zijn zo vriendelijk geweest ze voor algemeen gebruik beschikbaar te stellen. In veel gevallen gebeurt dat echter ook niet. Onderzoekers zitten jaren op hun materiaal in de hoop de geïnvesteerde tijd nog enigszins te kunnen verantwoorden door er zelf een aantal publicaties aan te wijden.

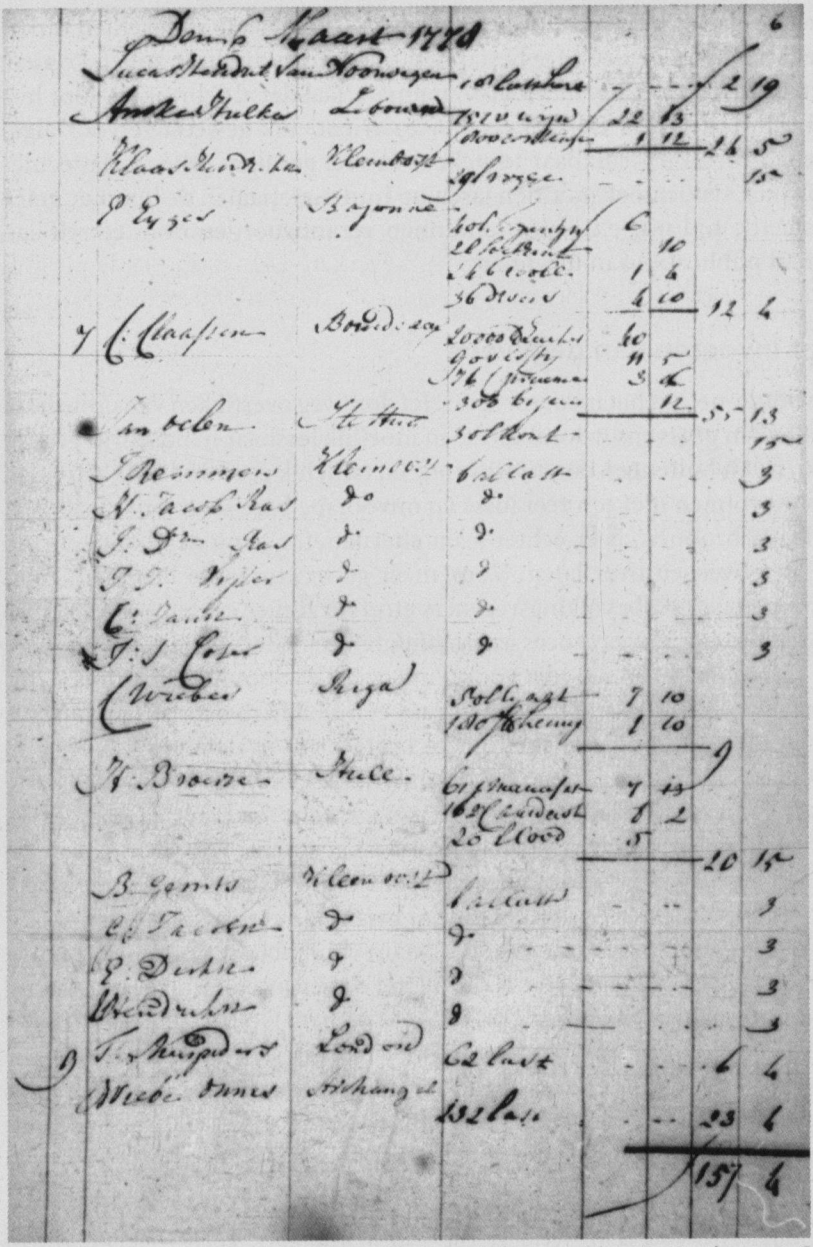
Het invoeren van gegevens

De bottle-neck is het invoerproces. Het domweg overtikken van grote hoeveelheden gegevens, vooral als die in moeilijk leesbaar handschrift gesteld zijn, is een buitengewoon tijdrovende en tamelijk geestdodende activiteit. Voor bronnen met een zeer losse en onvoorspelbare structuur, zoals verhalende bronnen, is er echter geen alternatief, zolang de resultaten van OCR te wensen over laten. Voor meer gestructureerde bronnen, zoals scheepsregisters, bevolkingsregisters etcetera zijn er echter wel mogelijkheden om het invoerproces aanzienlijk te versnellen.

Dergelijke bronnen, die vooral in sociaal-economisch historisch onderzoek gebruikt worden, hebben vaak een zich herhalende structuur die ze bij uitstek geschikt maakt om ze op te slaan in databases. Opslag in databases biedt een aantal voordelen ten opzichte van opslag als 'platte tekst'. Ten eerste bieden database management systemen geavanceerde mogelijkheden voor het bevragen van de gegevens. Daarnaast biedt de mogelijkheid van directe uitvoer naar statistische pakketten en spreadsheet programma's een breed spectrum van kwantitatieve analyses.

Maar voor je ook maar iets kunt analyseren met een computer, dien je het eerst in te voeren.[6] Het is opvallend dat er in de informatica weinig interesse voor het invoerproces is waar te nemen. Hoewel de geleerden het erover eens zijn dat als je ergens rommel in stopt, je er hoogstens ook weer rommel uit krijgt, is er over het algemeen weinig aandacht voor de mogelijkheden voor kwaliteitsbewaking tijdens de invoer. In alle database management systemen zijn mogelijkheden aanwezig, zoals *range-* en *type-*

6 Het onderstaande is een bewerking van een deel van het vierde hoofdstuk van mijn dissertatie: G.M. Welling, *The prize of neutrality, Trade relations between Amsterdam and North America 1771-1817. A study in computational history* (Hilversum 1998).



Pagina uit het *Havenboek van de Heffing van het Paalgeld* te Amsterdam van 6 maart 1778.

checking, validaties en *default-generation*, maar in de meeste gevallen worden ze domweg niet toegepast. De reden hiervoor ligt in het feit, dat het invoeren van data meestal laaggeschoolde arbeid is. In historisch onderzoek is er echter doorgaans geen scheiding tussen data-invoer en analyse: dezelfde persoon vervult beide taken. Maar vaak weet die persoon weer te weinig van het hulpmiddel af om het effectief te benutten. Omdat data-invoer een tamelijk geestdodend werkje is, zullen de meeste onderzoekers het invoeren tot een minimum willen beperken. Dat resulteert dan in de eerder genoemde probleemgerichte aanpak, die bestanden oplevert die voor hergebruik nauwelijks geschikt zijn.

In mijn eigen onderzoek over de achttiende eeuwse handel van Amsterdam heb ik gebruik gemaakt van de *Havenboeken van de heffing van het Paalgeld* te Amsterdam, kortweg meestal de *Paalgeldregisters* genoemd.[7] In deze registers, die het laatste deel van de achttiende en het begin van de negentiende eeuw beslaan, zijn nauwkeurige beschrijvingen te vinden van alle schepen die door de grote zeegaten in Amsterdam aankwamen. Daar moesten alle schepen een heffing over hun lading betalen. Uit de opbrengst van deze belasting betaalde de stad Enkhuizen, die het recht had deze heffing te doen in alle Zuiderzeehavens, het onderhoud van de bebakening van de Zuiderzee. Over ieder onderdeel van de lading werd een verschillende heffing gedaan en zodoende weten we nu welke goederen er in Amsterdam geïmporteerd werden. Natuurlijk zijn dergelijke bronnen nooit helemaal te vertrouwen: smokkel en belastingontduiking zijn van alle tijden.

Aanvankelijk had ik het ambitieuze plan om deze registers totaal te digitaliseren, maar al snel bleek dat geen haalbare kaart te zijn. Een proefproject waarin ik alleen de gegevens van het jaar 1778 verwerkte, duurde al bijna een jaar. Er waren voor 65 jaar registers beschikbaar. Omdat ik de onderneming niet meteen wilde opgeven ben ik gaan zoeken naar methoden om het invoeren van gegevens te versnellen. Omdat ik toch een zo nauwkeurig mogelijke digitale transcriptie van de bron wilde maken, lag de oplossing in het reduceren van het aantal benodigde toetsaanslagen om tot die transcriptie te komen. Tegelijkertijd wilde ik een zo hoog mogelijke accuratesse bereiken. Omdat de door mij gebruikte methodes algemeen toepasbaar zijn en in meerdere onderzoeken tot goede resultaten geleid hebben, lijkt het me nuttig ze hier nader te bespreken. Daarbij zal ik het Paalgeldonderzoek steeds als voorbeeld gebruiken.

7 G. M. Welling, *The prize of Neutrality.*

Validatie

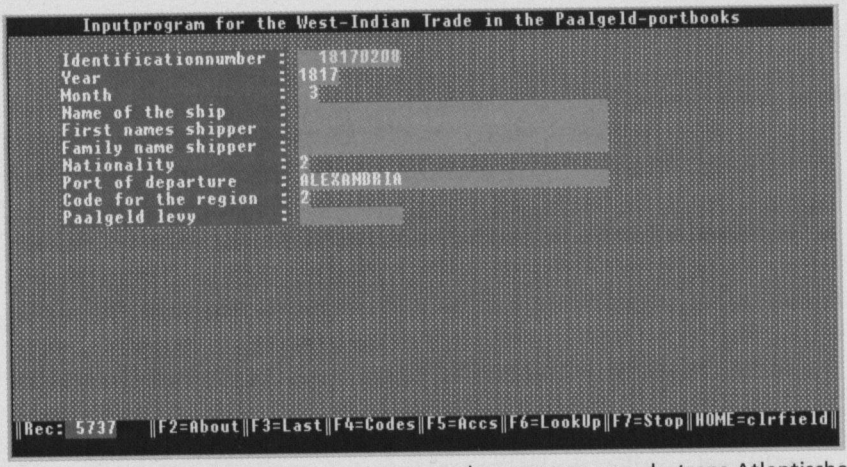
Database management systemen bieden goede mogelijkheden om de invoer direct te controleren. Door hiervan goed gebruik te maken kan je de tijd die nodig is voor de controle of de invoer wel goed gedaan is enorm beperken. Door het opstellen van regels waaraan de invoer moet voldoen – validatieregels - kan het systeem alle invoer die niet aan de regels voldoet herkennen als onjuist en weigeren. Dergelijke regels kunnen simpel zijn: een vertrekdatum kan nooit voor een aankomstdatum liggen. Maar het kan ook zeer complex worden: de basis van de heffing van het Paalgeld voor de verschillende goederen is beschreven in de *Observantie van de Heffing van het Paalgeld.* Voor de meeste graansoorten moest een stuiver per twee last betaald worden. Indien je die regel in het systeem opslaat, zal het een foutmelding genereren als je voor twee last graan een gulden als heffing wilt invoeren. Helaas komen rekenfouten in historische bronnen vaak voor en dan moet je dus een beslissing nemen: of je volgt de bron zo nauwkeurig mogelijk, of je neemt het correcte bedrag op. Ik heb voor de zekerheid maar beide opgenomen.

Het formuleren van validatieregels vereist een goede kennis van de bron en een minimaal inzicht in programmeren: moderne databasesystemen bieden hiervoor allerlei hulpmiddelen. Het verbaast mij iedere keer weer, dat zelfs deze tamelijk eenvoudige beveiliging op de invoer van gegevens in veel historische projecten ontbreekt. Maar ook in vele commerciële systemen kun je rustig een overlijdensdatum invoeren die voor de geboortedatum van een persoon ligt.

Maar met goede validatieregels gaat de invoer nog niet sneller: alleen het corrigeren van de dataset zal minder tijd in beslag nemen. Om echt tijdwinst te bereiken moeten we gebruik maken van de faciliteit van de meeste database management systemen om standaardwaarden (*defaults*) te genereren.

Defaults

Als je een adressenbestand gaat opzetten en je meeste kennissen wonen in Doodstil, dan is het aan te raden om Doodstil als defaultwaarde te kiezen voor de woonplaats: normaal gesproken zou je acht toetsen moeten indrukken om de naam Doodstil te vormen en vervolgens op de ENTER-toets drukken. Dat zijn samen negen toetsaanslagen. Indien Doodstil als



Screenshot van het invoerprogramma voor de gegevens van de trans-Atlantische handel uit de Paalgeldregisters. Een groot aantal waarden is als default reeds ingevuld op basis van het laatste geval.

defaultwaarde gekozen is, hoef je iedere keer dat die waarde inderdaad correct is alleen maar de ENTER-toets in te drukken en dat scheelt dan per keer acht aanslagen. In gevallen dat Doodstil niet de woonplaats is, levert dat geen probleem op: zodra je een andere toets dan de ENTER-toets aan het begin indrukt, verdwijnt de defaultwaarde en kun je intikken wat je wilt. Voordeel is ook dat er maar één spellingsvorm van Doodstil in het bestand komt.

Het nadeel van deze benadering is het statische karakter. De voor de hand liggende keuze voor de defaultwaarde is het meest voorkomende geval (de modus), maar deze waarde is in de meeste gevallen niet correct. Bij een bi-modale verdeling in je gegevens loopt het al snel mis. Defaults die vaker onjuist dan juist zijn hebben een averechts effect. Hoewel ze voor de gebruiker geen extra werk opleveren, gaan ze hem irriteren. Dat is weer het laatste wat je wilt, want dat resulteert in een aanzienlijk lagere productie. Kortom, het gebruik van een statische defaultwaarde is eigenlijk alleen aan te raden indien de modus ook verreweg het meest voorkomt.

Dynamische defaultwaarden genereren is echter gecompliceerd. In wezen proberen we namelijk het intelligente gedrag van de mens na te bootsen. Zonder dat we ons dat bewust zijn, gaan mensen vaak te werk alsof ze regels hebben gevonden in een bron. De schepen die van Archangel naar Amsterdam voeren hoefden nooit hun lading te specificeren, maar betaalden een vast bedrag per last goederen. Na een aantal keren een derge-

lijke vermelding in het register gezien te hebben, ga je automatisch bij de beschrijving van de lading een streepje invullen: en na een tijdje zie je ook dat per 2 last 1 stuiver betaald werd en dan hoef je alleen maar te zien dat er 14 last lading was om bij de heffing 7 stuivers in te vullen. Als die 14 soms niet duidelijk leesbaar is, kun je die reconstrueren aan de hand van de heffing van 7 stuivers. In essentie vindt hier een omkering van de validatieregels plaats: je gebruikt de regels niet meer als controle achteraf, maar om de defaultwaarde te genereren. Deze benadering vraagt een wat programmeerwerk, maar loont snel. Enerzijds kun je zo het aantal toetsaanslagen aanzienlijk beperken, anderzijds helpt dit bij het correct lezen van gecompliceerde handschriften.

Een dergelijke benadering kan alleen werken als er tussen de verschillende gegevens die ingevoerd moeten worden afhankelijkheidsrelaties bestaan. In veel gevallen zijn die er echter niet. Toch zijn er ook in die gevallen vaak mogelijkheden om dynamisch goede defaultwaarden te genereren. In registers als het Paalgeldregister komen vaak dezelfde waarden in clusters voor. In bevolkingsregisters kom je op één pagina vaak steeds dezelfde achternaam tegen, of dezelfde geboorteplaats. Zelfs bepaalde voornamen kunnen in een familie bijzonder vaak voorkomen. Door de achttiende eeuwse gewoonte om met groepen koopvaardijschepen in konvooien te varen, komt het in de Paalgeldregisters vaak voor dat de haven van herkomst van een schip precies hetzelfde is als van het bovenstaande geval: ze zijn immers samen opgevaren. En omdat schepen uit dezelfde plaats vaak dezelfde goederen vervoerden, kun je beslissen al die waarden van het vorige geval die vaak identiek zijn te laten staan voor het volgende geval.[8] Het overnemen van alle waarden is echter weer contraproductief: de invoerder gaat dan twijfelen of het vorige geval wel correct is opgenomen in het bestand en bevestigt voor de zekerheid de invoer nog maar eens en vervolgens heb je een aantal kopieën van één geval in je bestand.

Een andere vorm van het genereren van defaultwaarden maakt gebruik van hetgeen er tot dan toe is ingevoerd. Omdat in dergelijke registers een groot aantal nominale gegevens beschreven zijn, die maar een beperkt aantal vormen kunnen hebben, kun je door analyse van hetgeen er al ingevoerd is defaultwaarden bepalen. De namen van de herkomstplaatsen van de schepen die in Amsterdam binnenliepen vormen een beperkte verzameling, die steeds in nieuwe volgordes weer voorkomt. Door het systeem mee te laten zoeken in de reeds ingevoerde gegevens kun je op een

8 Dit noemt men 'limited carry over'.

zeer doeltreffende manier defaultwaarden genereren. Eigenlijk gaat het hier om hetzelfde als er ingebouwd is in de *Reisplanner* van de NS: met iedere letter die je intikt zoekt het systeem in de verzameling van mogelijkheden die daarvoor alfabetisch geordend is: in 95% van de gevallen is na vier aanslagen de juiste plaats gevonden. Deze techniek (*command completion* of *incremental searching*) zorgt er ook voor, dat er geen spellingsvarianten van dezelfde naam optreden, tenzij je die bewust helemaal gaat intikken. De verzameling waarden die je gebruikt wordt gevormd door hetgeen je al ingevoerd hebt. Hoe meer je ingevoerd hebt, hoe vaker deze benadering snel tot treffers zal leiden.

Ook hier is er sprake van een tweesnijdend zwaard: de snelheid van de invoer neemt toe, terwijl het anderzijds een goed hulpmiddel is bij het herkennen van min of meer onleesbare waarden. Vaak kun je het begin van een woord nog wel herkennen, maar wordt het soms daarna onleesbaar: indien het systeem gebruik maakt van *command completion* zal je nu automatisch een suggestie zien, die in ieder geval correct is voor zover het de tot dan ingevoerde letters betreft en vaak zal dit de juiste waarde zijn.

Toch is deze benadering niet in alle gevallen handig: indien namen alleen verschillen vertonen in de laatste letters, duurt het toch nog te lang voor de juiste waarde gevonden is. CHRISTIAANZAND en CHRISTIAANZUND zijn twee spellingsvarianten van waarschijnlijk dezelfde havenplaats die in mijn onderzoek voorkwamen. Omdat ik alle spellingsvarianten in de transcriptie wilde handhaven en pas later wilde standaardiseren – je weet immers maar nooit of je meteen de goede keuze doet en die is dan niet meer om te keren – wilde ik beide vormen kunnen vastleggen. Maar voor de eerste elf letters zijn beide vormen identiek! Al snel heb ik toen besloten om het invoerprogramma zo aan te passen, dat je gebruik kon maken van de pijltjestoetsen om snel waarden te vinden die volgden op de tot dan gesuggereerde defaultwaarde: zodra je op een pijltjestoets drukte, ging er een venstertje open met de beschikbare waarden en zo kon je dergelijke lange waarden ook veel sneller invoeren. Het zien van de mogelijke waarden was ook weer een hulp bij het lezen slecht leesbaar handschrift.

Screenshot van het invoerprogramma voor de gegevens van de trans-Atlantische handel uit de Paalgeldregisters. In het kleine venster zijn de codes voor coded-input voor havennamen te vinden.

Kan het nog sneller?

Langzaam maar zeker is dit probleem een obsessie voor me geworden: hoe kan het nog sneller? Het laatst ontwikkelde wapen in de strijd tegen de klok heb ik *coded input* genoemd. Mensen hebben allerlei technieken ontwikkeld om dingen snel op te schrijven, zoals bijvoorbeeld steno, en toch de hele tekst te behouden. Iets dergelijks wilde ik ook voor het invoeren van mijn gegevens hebben.

De havennaam St. Petersburg komt in de Paalgeldregisters veelvuldig voor, maar omdat er nogal wat andere plaatsen naar heiligen vernoemd zijn werkt *command completion* hier niet optimaal. Gebruik maken van een korte code om toch de hele naam in te voeren was de oplossing. Voor alle vaak voorkomende plaatsnamen heb ik mnemonische codes bedacht van maximaal drie letters. Om deze codes te kunnen invoeren moest het systeem van *command completion* echter uitgeschakeld zijn: dit gebeurde door alle codes met een spatie te laten beginnen. Zodra een waarde met een spatie begon, zocht het systeem nu in het codeboek naar de bijbehorende plaatsnaam. Zo leidden het invoeren van _SP (vier aanslagen) tot St. Petersburg (vijftien aanslagen): een zeer aanzienlijke winst. Door de meest frequent voorkomende waarden een zo kort mogelijke code te geven viel zo de meeste tijdwinst te maken. Zelfs spellingsvarianten kon ik nog behouden door bijvoorbeeld _SP2 te gebruiken voor de voor St. Pietersburg.

Het handhaven van spellingsvormen is een voorwaarde om de gegevens zo brongetrouw mogelijk op te slaan. Pas in een later stadium van het onderzoek kun je besluiten verschillende spellingswijzen te standaardiseren. Doe je dat in een eerder stadium, dan zijn fouten nooit meer te corrigeren. Om een wildgroei van spellingsvarianten tegen te gaan is het echter wel verstandig om het invoerprogramma er iedere keer op te laten wijzen als er een nieuwe spellingsvariant wordt ingevoerd. Het is in mijn onderzoek verschillende keren voorgekomen dat een bepaalde havennaam eigenlijk niet goed leesbaar was. Pas in een later register in een ander handschrift vond ik soms de meest waarschijnlijke vorm. Door gebruik te maken van hercoderingslijsten heb ik de verschillende vormen in een gestandaardiseerde versie van het bestand tot één vorm herleid.

Wat win je ermee?

Uiteindelijk heb ik niet alle *Havenboeken van de Heffing van het Paalgeld* ingevoerd. Zelfs met deze methodes zou dat te veel werk voor één persoon geweest zijn. Maar het is me wel gelukt om de jaren 1742, 1771-1787 volledig in te voeren en voor de periode 1788-1817 de gegevens betreffende de trans-Atlantische handel, die in een afzonderlijk deel van de registers stonden. Dat wil zeggen ongeveer 60.000 scheepsbewegingen en 105.000 ladingbeschrijvingen.

Maar misschien nog wel belangrijker dan de snelheidswinst door het terugbrengen van het aantal toetsaanslagen was de kwaliteitswinst. Door alle controles in het systeem, was er nauwelijks data-vervuiling opgetreden. De correctie bleek zeer snel uitvoerbaar. En iedereen mag alles controleren, want de complete dataset en coderingslijsten staan op het Web.[9]

Voor mijn eigen onderzoek heb ik berekend, dat ik de benodigde tijd voor het invoeren met bijna een jaar heb ingekort. Maar eigenlijk is die berekening niet kloppend. Ik ben er hiervoor vanuit gegaan dat je een achturige werkdag in zijn geheel zou kunnen besteden aan invoerwerk, maar dat houdt vrijwel niemand vol. Meer dan een uur of vier invoeren op een dag haal je niet. Daarnaast heb ik de verschillende technieken pas gedurende het onderzoek ontwikkeld en de tijdswinst aan het einde was veel groter dan aan het begin. En het tijdsverlies door de tijd besteed aan het ontwikkelen van het programma heb ik niet meegerekend. Alles bij elkaar

9 *http://odur.let.rug.nl/~welling/paalgeld/appendix.html.*

is de tijdwinst waarschijnlijk veel groter geweest. Toepassing van deze methodes in ander onderzoek heeft eigenlijk pas goed de waarde ervan aangetoond.[10]

Conclusie

Invoeren van gegevens blijft een tijdrovende en vaak geestdodende bezigheid, maar het hoeft geen onoverkomelijke drempel te zijn. Het is treurig dat er in veel historisch onderzoek nog steeds invoerwerk gedaan wordt op een manier die geen gebruik maakt van de computer als intelligent hulpmiddel. De minimale investering die er nodig is om dergelijke methodes in intelligente invoerprogramma's in te bouwen, betaalt zich razend snel terug.

Misschien duurt het nog een tijdje, maar de historicus als kritische consument van digitale bestanden kan niet ver weg meer zijn. En dan zullen bestanden die geen aanzienlijke meerwaarde boven gedrukte bronnenedities niet meer voor zoete koek geslikt worden. Kwaliteit, compleetheid en controleerbaarheid zullen de belangrijkste criteria zijn. Om aan de behoeften van die kritische consument te kunnen voldoen, zullen nog vele historische bronnen gedigitaliseerd moeten worden. Hopelijk krijgen we meer dan alleen digitale plaatjes van bronnen.

10 Deze methodes zijn onder andere gebruikt in het Srebrenica-onderzoek van het NIOD, en voor verschillende historische onderzoeken (C. Lesger, M. Peterzon, H. van Wijngaarden).