

Over coderen, analysestrategieën en onderzoeksdesign bij software

Jeanine Evers

Johnny Saldaña, *The Coding Manual for Qualitative Researchers*, London: Sage, 2009, 240 pagina's, ISBN 978-1-84787-549-5, £ 22,99.

Silvana di Gregorio & Judith Davidson, *Qualitative Research Design for Software Users*, Open University Press/McGraw-Hill Education, 2008, 267 pagina's, ISBN 978-0-335-225217 (pb), £ 25,99.

Beide boeken hebben de pretentie als handleiding bij (softwarematige) analyse gebruikt te kunnen worden, maar de insteek en schrijfstijl is wel heel verschillend. Lijken Di Gregorio en Davidson een lichte bias te hebben richting deductieve werkwijze, bij Saldaña ligt dat juist andersom; hij is meer van het inductieve. Is het boek van Saldaña vooral een leeswijzer en reproductie van andere auteurs met een klein deel eigen ervaring erin verwerkt, het boek van Di Gregorio en Davidson is juist een weergave van de praktijk(ervaring) van beide auteurs. En daar komt waarschijnlijk ook het verschil in aanpak van beide boeken vandaan. In het boek van Saldaña komen diverse codeerwijzen voorbij met bij elk type een leeswijzer, Di Gregorio en Davidson hebben het standpunt, dat het werken met software een heel nieuw genre onderzoek inluidt. In dat verband spreken ze van een *E-project*. Zij gaan ervan uit dat het gebruik van software verdisconteerd moet worden in het onderzoeksdesign. In deel I zetten ze heel systematisch uiteen waar zo'n E-project qua design aan moet voldoen, wil je gemakkelijk met software kunnen werken en die ook optimaal benutten. Ze illustreren dit in deel II met een aantal voorbeeldprojecten uit de onderzoekswereld, die door een andere doelstelling, ook qua ontwerp en softwaregebruik anders kunnen uitpakken. Bovendien laten zij daarbij een aantal softwarepakketten de revue passeren. Beide boeken zijn bedoeld om de onderzoeker te helpen bij zijn onderzoeksproject. Maar helpen ze ook? Daar ga ik in deze boekbespreking dieper op in.

In de *Coding Manual* van Saldaña komen 29 (!) verschillende codeerwijzen voorbij; deze inventarisatie is een van de doelstellingen van dit boek. Daarnaast beoogt Saldaña met het boek een discussie over de functies van codes, coderen en analytische memo's aan te snijden, en een leeswijzer ter verdieping aan te reiken. Ik zou zeggen dat hij in alle drie de doelstellingen geslaagd is, maar wel in verschillende mate. De doelgroep lijkt, gezien de toch wat oppervlakkige behandeling van codeerwijzen, met name de beginnend analist; voor de meer ervaren analisten is het eerder een naslagwerk met leuke (nieuwe) *tools* om mee te experimenteren.

Het boek van Di Gregorio en Davidson is vooral bedoeld als een praktische handleiding voor diegenen die overwegen software te (gaan) gebruiken bij hun onderzoeksproject. In

dat kader is de vormgeving met een ringband ook erg handig, hoewel het boek niet dusdanig is geschreven dat je het naast de computer legt als een echte handleiding. En bij normaal lezen (zeg maar: boek op je schoot in een stoel) is de ringband juist weer niet zo handig, door de wel erg slappe voorkaft. Zoals gezegd gaat het in deel I vooral om de integratie van onderzoeksontwerp met *QDAS* (Qualitative Data Analysis Software) waarbij er een heel aantal verhelderende plaatjes en praktische overwegingen worden aangegeeft, zoals bijvoorbeeld de rol van de *QDAS Supremo* (p. 61) bij onderzoek in teamverband, het belang van vooraf de verwachtingen van de software goed in kaart te brengen (p. 81), software als *ideas management tool* (p. 86) en het gebruik van software bij dissertaties (p. 67 e.v.).

In deel II worden de zaken uit deel I toegelicht aan de hand van concrete praktijkvoorbeelden, waarbij verschillende softwarepakketten voorbijkomen (Atlas.ti, MAXQDA, NVivo2, NVivo7 en Xsight). Dit deel is duidelijk bedoeld als inspiratiebron voor lezers die overwegen zelf een project te starten en raakt daarmee aan de kern van het boek, namelijk te laten zien welke implicaties de keuze voor het gebruik van analysesoftware heeft, al vanaf de ontwerpfase van je onderzoek! Het boek sluit af met een aantal zeer praktische en handige bijlagen met vragen, overzichtsplaatjes, woordenlijsten van softwarespecifieke terminologie en een aantal adressen. Voor zijn praktische doelstelling lijkt me dit boek zeer geslaagd.

Weer terug naar Saldaña. Hij definieert een code als volgt:

‘A code in qualitative inquiry is most often a word or short phrase that symbolically assigns a summative, salient, essence-capturing, and/or evocative attribute for a portion of language-based or visual data.’ (p. 4)

Hij maakt daarbij meteen een onderscheid tussen *first cycle* en *second cycle* coderen; het coderen gebeurt in verschillende rondes van verfijning, uitbreiding, hercoderen, abstrahering.

Di Gregorio en Davidson daarentegen definiëren, in samenhang met hun opvatting over het *E-project*, een code als veel meer dan een manier om de data te doorzoeken. Omdat zij codes door het hele project heen gebruiken en dus ook al in de ontwerpfase toepassen op literatuur, maar ook op correspondentie rond het onderzoeksproject en op het onderzoeksvoorstel zelf. Codes krijgen daarmee ook een methodologische functie, omdat zij al vanaf de voorfasen van het onderzoek er methodisch mee analyseren op zowel de aanhangende teksten (correspondentie, literatuur) als de data (transcripten, field notes, reflectiememo's) zelf. Daarmee hebben codes in hun werkwijze niet alleen maar een inhoudelijke/theoretiserende functie met betrekking tot de dataset, zoals ze meestal worden toegepast, maar worden ze veel breder ingezet. Zeker voor dissertaties heeft dit een grote meerwaarde vergeleken met de overheersende praktijk waarin *QDAS* (Qualitative Data Analysis Software) volgens hen tot dusver pas gebruikt wordt als de dataverzameling start.

De 29 codeermethoden worden door Saldaña via een kort profiel en met vermelding van hun analytische mogelijkheden uitgewerkt. Hij ziet al die verschillende codeerwijzen als evenzovele filters, waarmee je de data kunt beschouwen. In mijn eigen onderwijs heb ik dat wel 'brilletjes' genoemd waarmee je naar de data kijkt. Iedere keer als je een andere bril opzet, zie je net weer iets anders in dezelfde dataset. Hoe meer brillen je opzet, hoe beter je als het ware de dataset gebruikt. Terecht merkt Saldaña op, dat je, voordat je de brillen van de codeerwijzen opzet, de werkelijkheid al hebt gefilterd door je onderzoeksvraag, interviewtechniek, observatieverslagen, onderzoeksgroep en de wijze waarop je je data documenteert en organiseert. Het is dus ook zaak, je van deze filters, en soms zelfs ronduit vertekeningensvalkuilen, bewust te zijn.

Zoals eerder vermeld maakt Saldaña een indeling naar 'first' en 'second cycle' codes, en omschrijft dit als volgt:

'Data are not coded – they're recoded. ... First Cycle methods are those processes that happen during the initial coding of data and are divided into seven subcategories: Grammatical, Elemental, Affective, Literary and Language, Exploratory, Procedural, and a final profile entitled Themeing the Data. ... Second Cycle methods ... are a bit more challenging because they require such analytic skills as classifying, prioritizing, integrating, synthesizing, abstracting, conceptualizing, and theory building. If you have taken ownership of the data through careful First Cycle coding (and recoding), the transition to Second Cycle methods becomes easier. But be aware that codes are not the only method you should employ.' (p. 45)

De indeling naar *first* en *second cycle coding*, die in zichzelf nuttig en herkenbaar is, wordt door Saldaña niet altijd even consistent en logisch toegepast, er zitten regelmatig verdubbelingen in de codeerstrategieën die hij presenteert en hij lijkt er niet echt op overstijgend analytisch niveau naar gekeken te hebben. Zijn codeerwijzen werken namelijk soms puur inhoudelijk, dan weer op systeemniveau, dan weer op organisatorisch niveau en dat loopt allemaal door elkaar zonder dat daar een goed overzicht over ontstaat. Sommige codenamen voegen ook niets toe lijkt het ('lumper' en 'splitter' coding, 'simultaneous' coding), ze zijn alleen maar meer terminologie. Terwijl andere, zoals *thematic coding*, toch veel gebruikt in de literatuur (o.a. Franzosi, 2004; Coffey & Atkinson, 1996), door hem wordt afgewezen omdat hij vindt dat thema's een uitkomst zijn van het codeerproces. Hij benoemt ze dan vervolgens als: *themeing the data*. Hij noemt het zo, omdat het niet zou gaan om een korte codenaam, maar om een wat uitgebreidere zin om de data mee te analyseren. In de omschrijving die hij van verschillende auteurs aanhaalt, zou men echter evengoed een thematische code kunnen zien. En ook voor de codeersoorten die hij onderscheidt, geeft hij (soms op dezelfde pagina!) meerdere definities. In die zin is het boek nogal rommelig, en zelfs na een heel uitgebreid uittreksel maken, ben ik er nog niet helemaal uit wat zijn indeling nu precies betekent, of zij inhoudelijk wel klopt en wat ik er nu aan zou kunnen hebben. Voorlopig is mijn oordeel over Saldaña dan ook: het is een aardig boek om eens wat andere codeerwijzen in op te slaan en een leeswijzer te hebben

waar je ze verdiepend kunt vinden, maar het behoeft nog het nodige analytische denkwerk.

Daarentegen is het boek van Di Gregorio en Davidson een toonbeeld van heldere structuur en praktische toepasbaarheid, zeker voor wie overweegt met software te gaan werken. Ik weet niet of het de echte novice voldoende biedt om zelf aan de slag te kunnen, maar het is zeker de moeite waard. En beide boeken samen geven voldoende stof tot nadenken over de precieze implicaties van softwaregebruik in kwalitatief onderzoek.

Literatuur

Coffey, A. & Atkinson, P. (1996). *Making sense of Qualitative Data. Complementary Research Strategies*. Thousand Oaks: Sage.

Franzosi, R.P. (2004). Content Analysis. In M. Hardy & A. Bryman (Eds.), *Handbook of data analysis* (pp. 547-565). London: Sage.