The Qualitative Report

Volume 23 | Number 13 Article 5

3-6-2018

# Current Issues in Qualitative Data Analysis Software (QDAS): A User and Developer Perspective

Jeanine C. Evers
*Erasmus University of Rotterdam*, jcevers@eversresearch.nl



## Recommended APA Citation

Evers, J. C. (2018). Current Issues in Qualitative Data Analysis Software (QDAS): A User and Developer Perspective. *The Qualitative Report*, *23*(13), 61-73. https://doi.org/10.46743/2160-3715/2018.3205

# Current Issues in Qualitative Data Analysis Software (QDAS): A User and Developer Perspective

## Abstract

This paper describes recent issues and developments in Qualitative Data Analysis Software (QDAS) as presented in the opening plenary at the KWALON 2016 conference. From a user perspective, it reflects current features and functionality, including the use of artificial intelligence and machine learning; implications of the cloud; user friendliness; the role of digital archives; and the development of a common exchange format. This user perspective is complemented with the views of software developers who took part in the "Rotterdam Exchange Format Initiative," an outcome of the conference.

## Keywords

Qualitative Data Analysis Software, QDAS, Artificial Intelligence, Machine Learning, ATLAS.ti, Cassandre, Dedoose, f4analyse, MAXQDA, NVivo, QDA Miner, Quirkos, Transana, Exchange format, Interoperability, Qualitative Data Analysis, Learning Curve QDAS, Textual Data Mining, Cloud services.



## Acknowledgements

I am indebted to Anne Kuckartz, Thomas Muhr and Thorsten Pehl for their openness to the idea for the Conference at the Berliner Methoden Treffen 2015, which encouraged me to proceed it further. Last but certainly not least, I would like to thank Harry van den Berg and Richard Staring for their comments on an earlier version of this article and Christina Silver for her comments on a part of the earlier version.

*The Qualitative Report* 2018 Volume 23, Number 13, Special Issue Article 5, 61-74

# Current Issues in Qualitative Data Analysis Software (QDAS): A User and Developer Perspective

Jeanine C. Evers
Erasmus University of Rotterdam, The Netherlands

*This paper describes recent issues and developments in Qualitative Data Analysis Software (QDAS) as presented in the opening plenary at the KWALON 2016 conference. From a user perspective, it reflects current features and functionality, including the use of artificial intelligence and machine learning; implications of the cloud; user friendliness; the role of digital archives; and the development of a common exchange format. This user perspective is complemented with the views of software developers who took part in the "Rotterdam Exchange Format Initiative," an outcome of the conference.*
*Keywords: Qualitative Data Analysis Software, QDAS, CAQDAS, Artificial Intelligence, Machine Learning, ATLAS.ti, Cassandre, Dedoose, f4analyse, MAXQDA, NVivo, QDA Miner, Quirkos, Transana, Common Exchange Format, Interoperability, KWALON, Qualitative Analysis, Research Methodology*

KWALON, the Netherlands Association for Qualitative Research, is focused on the development of qualitative research methodology, including its propagation and reflection on its use. The 2016 international KWALON conference: *Reflecting on the future of QDA software: Chances and challenges for the humanities, social sciences and beyond*[1] sought to stimulate a constructive debate between software developers and users. My interest in developing and organizing this conference originated from my fascination, as a researcher, for both technology and methodology. I am neither a computer scientist nor a developer, but rather a professional user of Qualitative Data Analysis Software (QDAS), software trainer, and methodologist. In my opening speech, which is the starting point for this paper, I reflected on several developments around QDAS, hoping to encourage developers to work towards interoperability of their programs by creating a common exchange format for QDAS. Such an exchange format would make it possible for users to migrate an entire research project (including coded data files, memos and other annotations created by the researcher) from one software package to the other and back again. Current QDAS packages differ in subtle ways in both their underlying architecture and availability of tools and features. This variation has implications for what can be done during the analysis. Being able to migrate a project back and forth between software packages to take advantage of these differences would be very helpful, furthering both qualitative analysis and enhancing software adoption (Evers, 2011).

One impetus for KWALON 2016 was the CAQDAS Networking Project's CAQDAS 2014 Conference: *Past, present and future – 25 years of CAQDAS* held at the University of Surrey. Another was a European project proposal to develop software dedicated to analyzing historical multimedia data in digital archives. Third, several single-functionality software packages, such as nodegoat[2], were presented at THAT Camp Utrecht in 2015, a digital humanities gathering. As observed by Corti and Gregory (2011), the development of new single-functionality software packages fails to take into account the technological baseline offered by

[1] The conference took place on 25 and 26 August 2016 at Erasmus University in Rotterdam, The Netherlands.
[2] At the time of the gathering in 2015, nodegoat used metadata of people's correspondence to project historical social networks on the worldmap. In 2017, nodegoat has added more functionality: incorporating data and the possibility of tagging data.

*existing* QDAS packages. This paper reflects further on the issues raised at the conference, and further discussed them with developers in preparation for this paper[3], which is organized around seven questions:

1. To what extent should underlying design principles guide the integration of new tools in QDAS?
2. To what extent can "light" versions of QDAS be useful?
3. What is the relationship between approaches to training and research methodology?
4. In the age of big data, artificial intelligence and machine learning, what constitutes qualitative data analysis?
5. What security and accessibility issues are at stake when working in the cloud?
6. How might greater access to qualitative research processes conducted in QDAS via digital archives be achieved?
7. Is an "ultimate QDAS package" feasible?

The following sections explore these issues from a user perspective and, in some cases, the developers' perspective as a result of ongoing conversations.

### *1. To what extent should underlying design principles guide the integration of new tools in QDAS?*

Features available in QDAS are converging across packages with each new version. New data types, such as social media, and the use of citation management systems trigger new user needs. QDAS developers understandably respond to those needs by adding new features, resulting in "creeping featurism" (see Wolski, 2018, in this issue). Features are introduced in one package and are adopted by others. From a user perspective, it might be desirable to have as much functionality as is possible in one software package.

The adoption of features from one QDAS package to another, however, does not always result in the same functionality, due to differences in the underlying architecture of each QDAS package. Take the *hyperlinking* tool as an example. According to Silver and Lewins (2014), its functionality and ease of use differs between software packages, with some packages supporting paired linking and others supporting multiple links. This varies across packages. So, while each QDAS may offer a hyperlinking tool with a slightly different name[4] and function, the average user may not be aware of the implications of these differences until confronted with them during analysis. This is taken up by Melgar-Estrada and Koolen (2018, in this issue) as it relates to analysis of audio-visual data.

Users assume that software is purposefully designed with operations influenced by an underlying design philosophy defining both its structure and possibilities. Tools in a software package need to be aligned with the underlying architecture to enable smooth operation. An understanding of this may make it easier for users to explore tools and use them to find new ways of analyzing qualitative data (cf. Evers, 2001a; Van den Berg & Evers, 2006a, 2006b). However, in speaking with developers, not all of these assumptions hold up.

Developers do not always think of their software in terms of an underlying "design philosophy." Instead, designing software is a much more fluid process, starting from practical needs and evolving later on. Some organize yearly conferences with the user community to

[3] I am indebted to Anne Kuckartz, Eli Lieber, Adam Long, Friedrich Markgraff, Normand Peladeau, Thorsten Pehl, Daniel Turner and David Woods in making time available to reflect on my questions.

[4] ATLAS.ti and QDA Miner use the term "hyperlink," MAXQDA uses "textlink" and Nvivo refers to it as a "link" (Evers, 2015).

get feedback but at the same time they are thinking about new tools beyond the user imagination. As one developer put it:

> We are not practitioners. We don't come to the software with our own wishes about what the software must do. We have to try and figure out what users could want and […] would want, even if they don't know it yet or if they can't articulate that.

In lieu with this, another developer shared: "QDAS software can trigger innovations in methods and methodology."

Apparently, the phrase *design philosophy* did not seem to fit their experience in relation to further development. Their software architecture might be more implicit, rather than a map that dictates what can be done with a building. If software *did* have an implicit underlying architecture, adding new functions might increasingly blur the plan. As most of these developers started their career in social sciences, this might account for a disciplinary difference from procedures typically used in computer science. On the other hand, this implicitness might just indicate how programming software in reality works.

Developers do seem to be constantly oscillating between two concerns: (1) user requirements and the architecture of their software; and (2) cost/benefit considerations to remain competitive. All components play a role and have differing impacts on their decision making. Do users need a thorough understanding of software tools "under the hood" to use them well? My own experience with different packages and from observing students, supports my belief that understanding what a software tool is doing, combined with easy and intuitive operation, will make users more confident to explore new avenues of analysis using that tool.

### *2. To what extent can "light" versions of QDAS be useful?*

Around 2013, a movement counter to "creeping featurism" began to emerge: a "light" version of QDAS that offers basic functionality, with an intended result that the program is easier to learn. Two examples are f4analyse and Quirkos[5]. Some packages are now offering tiers of the program from basic or light to a more robust version with more features, for example, NVivo Starter, Pro and Plus; MAXQA Base, Standard, Plus and Analytics Pro; and Transana Basic, Professional and Multiuser. Some QDAS packages also offer a free version for tablets, e.g., the ATLAS.ti App and MAXQDA Reader. From a user perspective, arguments in favor of the light versions are (1) ease of use; (2) shorter learning curve; (3) no unnecessary features and (4) lower cost.

Some of these arguments were echoed by the developers, alongside the perception that a lighter version would be adequate for beginning users. Lighter versions also provide a less expensive option for users who expect these products to be affordable, even though this may not be realistic given the expense involved in software development. These light versions may resemble apps that are designed specifically for a certain functionality (cf. Do & Yamagata-Lynch, 2017).

For novice QDAS users this simplicity might be tempting, as they will be less distracted by the possibilities available in more comprehensive packages. As the need for additional features likely coincides with the maturation of a researcher/analyst, so hopefully will the creativity of thinking, using *thick analysis*[6] to find new ways to analyze the data (Evers & Van

[5] Cassandre and Transana also offer light versions "avant la letter."

[6] Defined as "the purposeful and creative combination of analysis methods to analyze a set of qualitative data" (Evers, 2016, para 1) as "a way to increade the validity, ecomprehend the complexity, and enhance the richness and in-depth understanding of the phenomena under study" (Evers & Van Staa, 2010, p. 756).

Staa, 2010). From a user perspective then, having more than one QDAS package available enables them to select the one that best fits the type of use, desired analytic methods (Evers, 2016), and the needs of the particular project.

### *3. What is the relationship between approaches to training and research methodology?*

Those new to QDAS do not always find it user-friendly or easy to learn. QDAS terminology can be difficult to master, and sometimes new users blame the software when the real issue is a lack of methodological knowledge. Both of these issues will be explored in this section. First, the terminology used in the QDAS interface differs between packages. Various terms are used for the same features and tools. For instance, *nodes* in ATLAS.ti denotes the objects in a network, as they are the center points of relations coming together, while in NVivo the term *node* is actually a synonym for what other QDAS packages call *codes* (Evers, 2015, p. 26). Data chunks are labeled differently as well: "quotations" (ATLAS.ti), "extracts" (Dedoose), "clips" (Transana) or "segments" (Qualrus; Silver & Lewins, 2014).

The methodological significance of the terminology used in QDAS is not very clear. Some terms seem to indicate a certain methodological stance (e.g., "hermeneutic unit" in ATLAS.ti until version 7; "variables" in MAXQDA and QDA Miner). In the past, software companies included their methodological inspiration in their promotional material, for example, hermeneutics and grounded theory for ATLAS.ti, mixed methods for MAXQDA. Developers purposefully chose to or refrained from using certain terminology in their software in their efforts to appeal to a certain research community. The fear of the computer only supporting one kind of method (e.g., grounded theory), or even taking over analysis completely has been a point of critique from the beginning of QDAS (Kelle, 1998). According to Jackson, Paulus and Woolf (this issue), this critique continues in the scholarly community despite consistent evidence to the contrary.

To further complicate things, as QDAS is used in very diverse disciplines, several terms are used differently in the social sciences and humanities[7]. Possibly, what lies at the basis of some of these terminology differences between humanities and social sciences is the distinction between a "project" and a "data source" as distinctive units of operation. QDAS typically thinks in terms of a *project*, and in this unit all of the data sources, contextual information, data manipulations by the analyst and results are stored. In humanities, it seems the distinctive unit is the *data source* as such, combined with the *metadata* about that data source. The data source can be either a textual file, a video file, an audio file or a photograph. The metadata about these data sources will enable researchers to search and find those files. These data sources may or may not be stored into a bigger unit on project level.

An example of different terminology between social sciences and humanities is the term *annotation* (Evers, 2016, para 16). In the humanities, annotation refers to all information about data sources that are used to classify, code, comment or link them together (Corti & Gregory, 2011). The tools to do this type of "annotating" in QDAS are called: codes, comments, memos and (hyper)links between data segments. In QDAS, the term annotation is used only for the notes written by the researcher or analyst about the project or objects within the project. That is a much smaller definition of the term and dependent on the software, they will be named: memos, comments, or both.

[7] Humanities and social sciences are not defined in the same way in different countries and hence do not represent the same disciplines everywhere. For this article, I will consider humanities to include archeology, linguistics, philosophy, religious studies, history, language studies, cultural studies, new media and art history. Social sciences consist of psychology, sociology, public administration, political science, science and technology studies, pedagogical science and cultural/social anthropology.

From a user perspective, this variation in terminology makes it harder to understand *which* tools are available in QDAS and *what* exact functionality they represent. To that end, standardization of terminology would certainly be helpful (Alexa & Zuell, 1999; Jansen, 1999). Although users often blame the software for their lack of understanding, other factors may be at play – lack of methodological fluency, lack of computer literacy or even ineffective approaches to training.

Novice researchers and/or new users often confuse software tools with analysis methods (Evers, 2009). Qualitative analysis is not a simple, predefined process of manipulating data. Rather, it is a heuristic process of collecting, searching and finding, connecting and transcendent interpretation (Evers, 2016). Therefore, it can be helpful to emphasize to learners that they need to distinguish between the interpretations that happen in the *mind* and how the *tools* are only there to support that process. Doing an introductory analysis course on paper before looking at QDAS, has proven useful. The tools within a QDAS can further the process in one's mind, because they enable you to do certain things one could not have done (so easily) without software. QDAS features may even trigger one's mind to think of new ways to analyze data. But the *analytic process* is still taking place in the *mind*. Codes for example can be used in very different ways for very different types of analysis, but the tool remains the same. (See for instance: Evers, 2015; Guest, MacQueen, & Namey, 2012; Richards, 2005; Saldaña, 2013; Silver & Patashnik, 2011).

Developers have experienced that the younger generation are more confident with their smartphones and less literate with desktop computing and see it as their responsibility to ensure that people can understand and operate the tools in their software. They try to improve the software if they find people are having trouble understanding the functionality, but they cannot predict where people will encounter difficulties.

Finally, ineffective training may be part of the challenge. Trainers are always looking for more effective approaches (Silver & Woolf, 2015), even writing books about the best way to use the software (di Gregorio & Davidson, 2008; Friese, 2014; Bazeley & Jackson, 2013). Developers might offer training themselves, and they create sample projects and videos to illustrate the software and make learning easier.

The steep learning curve of QDAS seems to be a multi-facetted problem. QDAS developers could help by standardizing terminology. Trainers could help by teaching both methodology and software functionality in their training, be it separately or integrated.

Effective learning of the software will require not only training but also time spent in the software, working with data. Understanding qualitative analysis *and* understanding the functionality of the software together will result in an effective use of the tool and the ability to reach the goals of the research.

### 4. *In the age of big data, artificial intelligence and machine learning, what constitutes qualitative data analysis?*

Developments in artificial intelligence, specifically in machine learning (or deep learning) are having an impact on qualitative research (see for example Besold, 2016). In essence, machine learning is enabling the computer to predict future events by learning from past events (Vasilev, n.d.). Artificial intelligence has been integrated to some extent into recent versions of QDAS, mostly in the advanced search and text mining tools. As texts and images are available for data mining from the Internet, there is an increasing demand for automated analysis and text mining approaches. Too much data, too little time, and a desire to meet the classic definitions of reliability seem to feed into this development (cf. Seale, 2014, Colley & Neal, 2012).

The relationship between big data, machine learning and qualitative analysis will strongly depend on epistemological stance, and relatedly, the methodological approach one favors. Shulman (2014), developer of TextSifter, in a presentation at the CAQDAS 2014 Conference, discerned three positions in working with qualitative data: (1) purist, concerned with "deep immersion, closeness to data, antipathy to numbers, credible interpretation, in-depth analysis, contextual and subjective"; (2) pluralist, characterized by "experimental, mixed methods, adaptive hybrid, flexible approach and interdisciplinary"; and (3) positivist, with a "quantitative focus on error measurement [is] critical, validity and reliability, replication and objectivity, generalization, hypotheses." Most qualitative researchers would position themselves in between the purist and the pluralist. One of the characteristics of qualitative research is that not only patterns in utterances, but indeed (small) differences in meaning are relevant, as well as the lacking of certain utterances. This is what constitutes both the context and the richness of the data and hence the adequacy of the analysis.

Blank (2014) referred to the "needle-and-haystack" dilemma of social scientists, as they want tools to summarize the haystack, because it takes too much effort to look for the needle. Current algorithms for machine learning and text mining to Blank are still weak, as they are theory- free and cannot handle context, because they only deal with the words and not the meaning of those words. He argued for a productive convergence of computer science and social science via QDAS. Wiedemann (2013) discerned four types of Computer-Assisted Text Analysis approaches: (1) "QDAS: context-comprehensive manual coding," (2) "Computational content analysis: context-neglecting automatic coding," (3) " Lexicometrics/corpus linguistics: context-observing content exploration," and (4) "Text mining: Pattern- and model-based latent context calculation" (para 3). His typology, however, neglects some of the automated data mining tools in current versions of QDAS. According to E. Rizkallah (2017, personal communication), context in artificial intelligence nowadays is still based on computerized recognition of words and their surrounding words in a data corpus.

Developers clarified that the ease of collecting huge amounts of textual data via the internet is creating a demand for automated analysis tools in QDAS. However, they distinguish between "a lot of data" and "big data," the latter being defined as being too big for "qualitative data analysis done by humans with tools." To some developers, automated analysis really refers to data processing and management, not data analysis.

According to the developers, artificial intelligence and machine learning are expected to lead to some "really amazing breakthroughs," but they will still not be able to "actually understand and interpret the world." Users may expect miracles from automated analysis because "… many users lack the technical imagination of what software can do, but also of what it cannot do." Because it is based on frequencies and statistical procedures and cannot very well interpret text in context, as humans can, some developers feel automated analysis is antithetical to qualitative data analysis. Others feel the definition of qualitative data needs to be expanded to include big data. Developers also pointed out that people might think that algorithms are neutral, but in effect, they are not. They are generated by humans and will carry the biases inherent to their creators. Algorithms cannot change their perspectives or interpret what they are seeing in the data.[8]

[8] There is a video in which a sensor in a soap dispenser did not recognize skin colors other than Caucasian. While not a deliberate flaw in the algorithm, it is an example of something that the software engineer did not think about when writing it: http://www.iflscience.com/technology/this-racist-soap-dispenser-reveals-why-diversity-in-tech-is-muchneeded/

The debate about whether or not machines are able to interpret texts as adequately as humans is still ongoing. For now, we still need to rely human interpretation of qualitative data; however machine learning may help by *suggesting* coding options. The analyst can either accept or refuse these.

### *5. What security and accessibility issues are at stake when working in the cloud?*

Storing information in and working from "the cloud" is becoming standard procedure. As researchers are obligated to assure participants that their data will be kept secure and confidential, questions rise about data security. Both technical and judicial issues can interfere with that responsibility, as countries have different regulations regarding data stored on servers[9]. These issues include accessibility of the data and the quality of encryption in ensuring the security of the data. Reliability of power and internet connectivity also impact working in the cloud.

From a user's point of view, the benefit of working in the cloud lies in the accessibility of the project from anywhere, enabling freedom of movement and teamwork. Developers see the ability to access a project from anywhere, across several devices and independent of operating systems as tempting characteristics of working in the cloud. Working in the cloud to them enhances data accessibility as it does not rely on a certain operating system. The downside is that the data are theoretically accessible to a much larger group and the encryption key is known to cloud services.

Regarding security, developers distinguish between "data confidentiality (authorized access to content)" and "data integrity (stored correctly and trustworthily)," as well as "data sovereignty (knowing where the data is stored)" and "data privacy (what can be shared)." To them, data confidentiality and data integrity primarily are the developer's responsibility, be it on a stand-alone computer or in the cloud, while data sovereignty and data privacy are the responsibility of the researcher. To some developers, the cloud calls for "new research ethics." A software company can either have their own server for data, in which case the security of that server is theirs to deal with, or use third-party servers, in which case the security is dealt with by the third party. If a third party is chosen to host the cloud service, this will add an extra layer of responsibility to the model and hence make it more complex. Such servers might be more interesting to hackers. Developers do not see absolute security as possible, but well-designed cloud services should be as safe as having your data on your computer. However, an important part of the security of data and projects has to do with researcher practices. In developers' experience, losing laptops, not making back-ups[10] regularly and in other ways not attending to data security, is more of a safety issue than the cloud. Additionally, developers view training researchers in the technicalities of keeping their data secure, e.g., by choosing strong passwords, as more effective than the technical security of the cloud itself. Researchers using the cloud, should consider an end-to-end-encryption service, in which the provider of the cloud service does not have the key to the encryption of the data. Their responsibility to ensure protection of the personal particulars of their respondents apparently now includes the cloud.

Working in the cloud has made the safekeeping of data a shared responsibility. In deciding on the use of the cloud, researchers could consider (1) how, where and when they want

[9] As of 12 July 2016, the European Committee rectified the EU-US Privacy-shield, protecting data from EU citizens, stored on servers in the US, in accordance to European privacy legislation. This agreement has replaced the earlier Safe Harbor Privacy Principles.

[10] Some QDAS therefore automatically remind users to make a back-up. While writing this paper, Intego declared the "World Backup day," stating: "Our goal is to raise awareness about the importance of regular backups...." Apparently, this problem of missing backups is familiar to other software developers as well.

access to their data; (2) how secure their data are expected to be, and (3) if and with whom they might want to share access.

### *6. How might greater access to qualitative research processes conducted in QDAS via digital archives be achieved?*

In Europe, digitizing and archiving historical and cultural data, as well as research[11] data is a trend[12]. Research funded under the Horizon 2020 program of the European Union must be published in open access journals, and the research data generated in publicly funded projects must be deposited in a digital repository. All of this is aimed at making data "findable, accessible, interoperable and reusable," otherwise known as the FAIR[13] principle. These developments have consequences for both researchers, developers of QDAS, and digital archives, as data should be produced in such a way that it can be archived.

A "data management plan," describing the archiving of data and results, will soon become mandatory for project funding in all disciplines, including humanities and social sciences in European countries. This measure was taken to stimulate the reusability of data for follow-up studies, replication and integrity studies. Digital archiving is gaining ground in the US as well and to that end, the US Qualitative Digital Repository[14], hosted by Syracuse University, organized a meeting at the end of 2016 to discuss best practices in QDAS projects and digital data repositories. Louise Corti of the UK Data Archive, developers in REFI and members of the coordination group exchanged experiences and suggestions for future needs (see Karcher & Pagé, 2017).

As an example, Evers (2015) illustrates an experiment with the Dutch repository DANS (http://www.dans.knaw.nl) in which we deposited both the *data set* and the *analysis products.* The dataset was a subset from the KWALON project (Evers, Silver, Mruck, & Peeters, 2011). As the data were harvested from the internet in 2010, it would lead to certain legal issues if made available through an archive. This problem was solved by referring to the original site if still available, or to other places where the files are now still available. As for some data the original site was no longer available, it was impossible to check the legal status of the file. The repository decided not to publish those files, but to put an explanation on their site explaining the missing files. In depositing the *analysis products*, we encountered other problems, as some of the researchers worked with QDAS or other specialized software and others did not. The *project files* from QDAS could not be transferred to other, broadly used software products like Microsoft Excel or Word, as a QDAS project file is far more complex. From the repository's viewpoint, not only the broadness of use, but also the sustainability of software is an issue (Aerts, Doorn, & Roorda, 2016), as the archive wants to assure the project files can be retrieved many years from now. In the end, we solved this problem by archiving either parts of the analysis outcome in pdf or Excel, or archiving the whole project file resulting from QDAS.

According to the developers, it is possible to recreate a project from several output files, but this is tedious work and requires a programmer and could not be done easily by a researcher. This makes a common exchange format, enabling people to export a whole project and import it in the software of their choice, a critical option for archiving. Earlier, a project led by the UK Data Archive to enable an exchange format resulted in the QuDex schema (Corti, 2008; Corti & Gregory, 2011), but this has not been adopted by QDAS developers. The Text Encoding

[11] See for example http://www.data-archive.ac.uk, http://www.dans.knaw.nl. Accessed 5 March 2017.
[12] For example, http://www.Europeana.eu. Accessed 5 March 2017.
[13] For more detail on FAIR, see https://www.force11.org/group/fairgroup/fairprinciples
[14] https://qdr.syr.edu. Accessed: 27 October 2016.

Initiative (TEI) developed a standard for the encoding of –primarily humanities – digital texts and is used by libraries.

### *8. Is an "ultimate QDAS package" feasible?*

Reflecting on these issues leads us to this final question which stems from the observation that QDAS packages are adding more features, therefore striving to provide all features a user could possibly want – with the goal of "being everything to everyone."

From the developers' perspective, the "ultimate" software does not seem like a realistic outcome. One of the developers compared it to the proverbial "pot of gold at the end of the rainbow…." Another developer says: "feature x takes up space and cognitive room [from the user] in your software," indicating that developers need to be choosy about which features to add and which not.

As software is designed from a particular epistemological stance, its strengths and weaknesses reflect that. However, this might not always be apparent, and the user may be unaware of it. This does not make the software less useful: it makes it very us able to *certain* analytic goals, and less so to *other* analytic goals. From a user perspective, it would be very helpful to move from one package to another in the course of a project, and thereby use their respective strengths to the utmost. Alexa and Zuell (1999) compared fifteen software packages and advocated for the interoperability of software (see Jansen, 1999), to which end Thomas Muhr, developer of ATLAS.ti, proposed the use of XML to enable software packages to "talk to one another" (Muhr, 2000).

At the moment, most QDAS packages still work with proprietary formats, which makes it difficult, if not impossible, to import a whole project created in one software package into another software package. There are different export possibilities in current QDAS packages, but as there is no common standard defined yet, the effectiveness is diverse.

## Conclusion

What can be learned from this thinking exercise in conjunction with the software developers? Although my assumptions about the explicitness of the design philosophy as a master plan for programming software were not necessarily shared by the developers, more intentional reflection on the interplay between the architecture of software and tool proliferation would be welcomed. Focusing on the underlying architecture of a QDAS package, in tandem with methodological and user perspectives, can enable further interoperability and innovation in QDAS and add to the development of new qualitative analysis methods. To this end, interchange between developers and experienced users could be of added value.

As a result of the conference, the developers present[15] agreed to work towards the goal of a common exchange format and have joined the Rotterdam Exchange Format Initiative (REFI), facilitated by a coordination group[16]. The developers engage in working discussions via an asynchronous forum[17], synchronous meetings and live meetings. The coordination group

[15] In alphabetical order: ATLAS.ti, Cassandre, Dedoose, f4analyse, MAXQDA, NVivo, QDA Miner, Quirkos, and Transana.

[16] In alphabetical order consisting of: Fred van Blommestein (University of Groningen), Jeanine Evers (chair, Erasmus University Rotterdam), Kristi Jackson (Queri, Inc.), Élias Rizkallah (Université de Quebec à Montreal) and Christina Silver (CAQDAS Networking Project, Surrey). [16] All members of this group are doing this work in their spare time, while the chair was funded partially by Erasmus Law School from September 2016 until August 2017. [16] All members of this group are doing this work in their spare time, while the chair was funded partially by Erasmus Law School from September 2016 until August 2017.

[17] Asynchronous forum stands for a kind of chat facility, whilst synchronous meetings are done in real time via the internet.

facilitates the meetings, suggests the agenda, takes notes and applies for funding[18]. At the time of writing this article, the group has created a data model[19] and is specifying a proposed common exchange format, which will be tested in the near future.

As a common exchange format will eliminate the need to develop an import functionality for each package individually, developers could focus their programming efforts on strengthening the tools they feel really fit their architecture, thus enhancing the functionality of their existing tools.

If development of a common exchange format succeeds, it will be a major step forward for the research community at large. Such a standard will enable researchers to migrate between software packages – be they lighter or more robust versions. In this way, users will be able to use QDAS to its fullest extent, both for the foreseen and unforeseen needs. This can enhance the methodology of qualitative analysis as well, as researchers will no longer be locked into the confines of one particular package, instead they will have freedom to further their analysis by using another set of tools.

In the academic community, having access to multiple QDAS, and the ability to work across them, would enable students to learn different software tools and packages during their education. This will enhance their analytic potential and the quality of their learning. For faculty, the possibility of migrating between software packages will both enhance their analytic options within one research project and enable them to collaborate with those using different packages. The expansion of QDAS can have a positive effect on pricing, while innovation and quality of both qualitative and mixed methods can be achieved by being able to switch between QDAS packages.

The issue of digital archives and an exchange format still remains. Developers chose not to include archiving in the first version of the common exchange format, as it is quite complex. If one wants to export data out of QDAS into a repository and back into the same (or another) QDAS, it even gets more complicated. For now, the goal is to get the common exchange format working first. The exchange into and out of a repository will remain on the wish list for the future.

## References

Aerts, P., Doorn, P., & Roorda, D. (2016). *Software sustainability. Final report.* The Hague, The Netherlands: Data Archiving and Networked Services (DANS).

Alexa, M., & Zuell, C. (1999). *A review of software for text analysis.* Mannheim: Zentrum für Umfragen, Methoden und Analysen (ZUMA).

Bazeley, P., & Jackson, K. (2013). *Qualitative data analysis with NVIVO.* London, UK: Sage.

Besold, T. R. (2016). On cognitive aspects of human-level artificial intelligence. *Artificial Intelligence, 30,* 343-346.

Blank, G. (May 2014). *Big data and CAQDAS*. Keynote presented at the CAQDAS Conference: Past, Present and Future – 25 Years of CAQDAS, East Horsley, Surrey, United Kingdom.

Colley, S. K., & Neal, A. (2012). Automated text analysis to examine qualitative differences in safety schema among upper managers, supervisors and workers. *Safety Science 50*(9), 1775-1785.

[18] Funding was received from KWALON, The Erasmus School of Law, the Erasmus Trust Fund and Erasmus Studio and the Canadian Social Sciences and Humanities Research Council.

[19] A data model is a representation of the objects that are wanted in the common exchange format and the relationships between those objects. It is based on current objects in each of the QDAS packages.

Corti, L. (2008). *JISC final report. Data exchange tools and utilities (DExT) repositories and preservation tools programme.* Colchester, UK: UK Data Archive, University of Essex.

Corti, L., & Gregory, A. (2011). CAQDAS comparability. What about CAQDAS data exchange? *Forum Qualitative Sozialforschung / Forum Qualitative Social Research, 12*(1). http://dx.doi.org/10.17169/fqs-12.1.1634

di Gregorio, S., & Davidson, J. (2008). *Qualitative research design for software users.* Maidenhead, UK: Open University Press, McGraw-Hill Education.

Do, J., & Yamagata-Lynch, L. (2017). Designing and developing cell phone applications for qualitative research. *Qualitative Inquiry, 23*(10), 757-767.

Evers, J. (2001a). *Instituties op Curaçao: Een politiek-bestuurlijke analyse. Kada fèlt di flor tin sumpiña* [Institutions on Curacao. A political-administrative analysis] (Masters thesis). Public Administration Erasmus University Rotterdam, nominated for the Master Thesis Award Public Administration, 2001.

Evers , J. (2009). Over kwalitatieve data-analyse valt nog veel te schrijven! [A lot can still be written about qualitative data analysis!] *KWALON, 14*(2), 50-53.

Evers, J. C. (2011). From the past into the future. How technological developments change our ways of data collection, transcription and analysis. *Forum Qualitative Sozialforschung / Forum Qualitative Social Research, 12*(1). http://dx.doi.org/10.17169/fqs-12.1.1636

Evers, J. (2015). *Kwalitatieve analyse: Kunst én kunde* [Qualitative analysis: art and expertise]. Amsterdam, The Netherlands: Boom Lemma Uitgevers.

Evers, J. C. (2016). Elaborating on thick analysis: About thoroughness and creativity in qualitative analysis. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, 17*(1). http://dx.doi.org/10.17169/fqs-17.1.2369

Evers, J., Silver, C., Mruck, K., & Peeters, B. (2011). Introduction to the KWALON experiment: Discussions on qualitative data analysis software by developers and users. *Forum Qualitative Sozialforschung / Forum Qualitative Social Research, 12*(1). http://dx.doi.org/10.17169/fqs-12.1.1637

Evers, J. C., & Van Staa, A.-L. (2010). Qualitative analysis in case study. In A. J. Mills, G. Durepos, & E. Wiebe (Eds.), *Encyclopedia of case study research* (pp. 750-758). Thousand Oaks, CA: Sage.

Friese, S. (2014). *Qualitative data analysis with ATLAS.ti* (2nd ed.). London, UK: Sage.

Guest, G., MacQueen, K. M., & Namey, E. E. (2012). *Applied thematic analysis.* Thousand Oaks, CA: Sage.

Jackson, K., Paulus, T., & Woolf, N. H. (2018). *The Walking Dead*: Unsubstantiated criticisms of qualitative data analysis software (QDAS) and our failure to put them to rest. *The Qualitative Report, 23*(13), 74-91. Retrieved from http://nsuworks.nova.edu/tqr/vol23/iss13/6

Jansen, H. (1999). ZUMA vergelijkt vijftien software pakketten voor tekstanalyse [ZUMA compares fifteen software packages for text analysis]. *KWALON 4*(3), 14-17.

Karcher, S., & Pagé, C. (2017). Workshop report: CAQDAS projects and digital repositories' best practices. *D-Lib Magazine, 23*(3/4). https://doi.org/10.1045/march2017-karcher

Kelle, U. (1998). (Ed.) *Computer-aided qualitative data analysis. Theory, methods and practice.* London, UK: Sage.

Melgar-Estrada, L., & Koolen, M. (2018). Audiovisual media annotation using qualitative data analysis: A comparative analysis. *The Qualitative Report, 23*(13), 40-60. Retrieved from http://nsuworks.nova.edu/tqr/vol23/iss13/4

Muhr, T. (2000). Increasing the reusability of qualitative data with XML. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, 1*(3). Retrieved from http://www.qualitative-research.net/index.php/fqs/article/view/1037/2244

Richards, L. (2005). *Handling qualitative data. A practical guide.* London, UK: Sage.

Saldaña, J. (2013). *The coding manual for qualitative researchers* (2nd ed.). London, UK: Sage.

Seale, C. (May 2014). *Analysis of large data sets using Wordsmith Tools and Wordstat*. Keynote presented at the CAQDAS Conference: Past, Present and Future – 25 Years of CAQDAS, East Horsley, Surrey, United Kingdom.

Silver, C., & Lewins, A. (2014). *Using software in qualitative research* (2nd ed.). London, UK: Sage.

Silver, C., & Patashnik, J. (2011). Finding fidelity: Advancing audiovisual analysis using software. *Forum Qualitative Sozialforschung / Forum Qualitative Social Research, 12*(1). http://dx.doi.org/10.17169/fqs-12.1.1629

Silver, C., & Woolf, N. H. (2015). From guided-instruction to facilitation of learning: The development of five-level QDA as a CAQDAS pedagogy that explicates the practices of expert users. *International Journal of Social Research Methodology, 18*(5). doi:10.1080/13645579.2015.1062626

Shulman, S. W. (2014). *Measuring reliability and validity in human coding and machine classification.* Paper presented at the CAQDAS Conference: Past, Present and Future – 25 Years of CAQDAS, East Horsley, Surrey, United Kingdom.

Van den Berg, H., & Evers, J. (2006a). Discriminatie-ervaringen 2005. Een onderzoek naar ervaringen met discriminatie op grond van land van herkomst, geloof en (huids)kleur [Discrimination experiences in 2005. A study into experiences with discrimination based on countries of origin, religion and (skin) color.]. In I. Boog (Ed.), *Monitor rassendiscriminatie 2005* [Monitoring racial discrimination in 2005] (pp. 15-48). Rotterdam, The Netherlands: Landelijk Bureau ter bestrijding van Rassendiscriminatie [National Office for Combating Racial Discrimination].

Van den Berg, H., & Evers, J. (2006b). Methodologische bijlage bij hoofdstuk 1: Discriminatie-ervaringen 2005 [Methodological appendix to chapter 1: Discrimination experiences in 2005]. In I. Boog (Ed.), *Monitor rassendiscriminatie* [Monitoring racial discrimination in 2005] (pp. 213-235). Rotterdam, The Netherlands: Landelijk Bureau ter bestrijding van Rassendiscriminatie [National Office for Combating Racial Discrimination].

Vasilev, I. (n.d.). *A deep learning tutorial: From perceptions to deep networks*. Retrieved from https://www.toptal.com/machine-learning/an-introduction-to-deep-learning-from-perceptrons-to-deep-networks

Wiedemann, G. (2013). Opening up to big data: Computer-assisted analysis of textual data in social sciences. *Forum Qualtative Sozialforschung / Forum: Qualitative Social Research, 14*(2). http://dx.doi.org/10.17169/fqs-14.2.1949

Wolski, U. (2018). The history of the development and propagation of QDA software. *The Qualitative Report, 23*(13), 6-20. Retrieved from http://nsuworks.nova.edu/tqr/vol23/iss13/2

## Author Note

Jeanine C. Evers has a background in cultural anthropology and public administration. She has extensive experience conducting qualitative research in both The Netherlands and the Dutch Antilles. She has held several university positions at Dutch universities since 2001, most recently in the criminology department of Erasmus University Rotterdam. She has published several books on qualitative interviewing and qualitative analysis in Dutch (2007-2015) including *The Qualitative Interview: Art and Skill*. In 2004, she started the KWALON course department for short courses in qualitative methods and data analysis software, which has functioned as Evers Research & Training since 2011. She serves on the board of KWALON, the Dutch Network of Qualitative Research, is a member of the Scientific Committee of the European Congress of Qualitative Inquiry and chairs the Rotterdam Exchange Format Initiative

(REFI). Correspondence regarding this article can be addressed directly to: jcevers@eversresearch.nl.

I am indebted to Anne Kuckartz, Thomas Muhr and Thorsten Pehl for their openness to the idea for the Conference at the Berliner Methoden Treffen 2015, which encouraged me to proceed it further. Last but certainly not least, I would like to thank Harry van den Berg and Richard Staring for their comments on an earlier version of this article and Christina Silver for her comments on a part of the earlier version.



## Article Citation

Evers, J. C. (2018). Current issues in qualitative data analysis software (QDAS): A user and developer perspective. *The Qualitative Report, 23*(13), 61-74. Retrieved from http://nsuworks.nova.edu/tqr/vol23/iss13/5