FROM TEXT TO AGENTS – PROCESS OF DEVELOPING EVIDENCE-BASED SIMULATION MODELS

Ulf Lotzmann
Martin Neumann
Michael Möhring
Institute of Information Systems Research
University of Koblenz-Landau
Universitätsstraße 1, Koblenz 56070, Germany
E-mail: {ulf, maneumann, moeh}@uni-koblenz.de

KEYWORDS

Evidence-driven modelling, qualitative data analysis, conceptual modelling, agent-based simulation, normative agent architecture, criminal networks.

ABSTRACT

Evidence-driven modelling approaches have gained attention in the computational social science community over the past couple of years. A number of research projects have dealt with this topic, and some of the results and experiences gathered in participating in two recent projects are presented in this paper. The main assertion of the paper is motivated by the finding that in order to be reproducibly successful with creating agent-based models from textual evidence, the development process needs to involve conceptual modelling in a well-structured way. This becomes more important if stakeholders should be taken along the modelling, and if due to the amount and kind of evidence to be regarded, appropriate analysis techniques have to be used. This paper aims to give some guidelines on how such an approach could be designed, which has been successfully applied in the realisation of a model about the collapse of a criminal network - from the initial preparation of raw texts, through text analysis, conceptual and formal modelling, right up to the final presentation of simulation results.

INTRODUCTION

The research presented in this paper is part of the GLODERS project (http://www.gloders.eu/) and aims to provide a sort of guideline for an evidence-driven modelling process, a process to transform empirical data into simulation models. This process naturally involves a number of phases, comprising a variety of different steps, involving activities like manual and automatic analysis of textual evidence, conceptual modelling, model transformation, implementation of formal models and simulation experimentation and result analysis. A model of the internal dynamics within a criminal organisation in a European country, building on Police interrogations resulting from a number of investigations of a particular criminal group is used to show the exact activities performed for model development, but also to illustrate each phase by presenting some of the results.

The proposed approach is partly based on a modelling process developed in the OCOPOMO project (Scherer et al. 2013, www.ocopomo.eu/), extended by state-of-the-art data analysis techniques at the initial stages of the process, and adapted to the requirements of the concrete modelling tasks of the GLODERS project. These requirements mainly regard the following topics:

- stakeholder participation: stakeholders provide empirical domain knowledge and discuss the state of development with the modellers on various occasions;
- type of empirical data: reports from criminal investigations (interrogation reports, court files);
- normative agents: an existing theoretical foundation constrains the model design.

These requirements shape also the presentation of the abstract process steps in this paper. Firstly, an overview on the modelling process is given. Subsequently, the single phases are discussed in sections for the conceptual analysis process, the simulation modelling and testing and the simulation experimentation. A section summarising the outcomes and giving an outlook for future research concludes the paper.

EVIDENCE-BASED MODELLING PROCESS

This paper is a demonstration and proof-of-concept of the approach to making use of evidence based modelling as a core tool in a process of grounded theory development (Neumann 2015; Dilaver 2015). Central to this account is to rigorously grounding model assumption in empirical narratives (Lotzmann and Wimmer 2013). Thus it takes a stance in the ongoing debate on simple or realistic modelling strategies (Coen 2009). Evidence-based modelling is an approach of developing models not on simple theoretical assumptions but on factual empirical evidence, sometimes referred to as KIDS principle (Edmonds and Moss 2005) by making use of ethnographic accounts to get away from numbers (Yang and Gilbert 2010). This is particularly popular in participatory modelling (Barreteau et al. 2003). Its growing recognition is demonstrated in a recent special issue on using qualitative evidence to inform the specification of Agent-Based Models in Vol. 18(1) of JASSS.

The following sub-section introduces one of the use cases developed in the GLODERS project, which serves as means to exemplify the main assertions of the paper. Afterwards the design of the process is outlined, which then will be explained in detail in the course of the paper.

Proceedings 29th European Conference on Modelling and Simulation ©ECMS Valeri M. Mladenov, Petia Georgieva, Grisha Spasov, Galidiya Petrova (Editors)
ISBN: 978-0-9932440-0-1 / ISBN: 978-0-9932440-1-8 (CD)

Use case: Modelling the collapse of a criminal network

The case of the collapse of a criminal network is based on files from police interrogation, arising from investigations concerning a criminal gang involved in drug trafficking and money laundering. Within this gang a division of labour was established: 'black collar criminals' were responsible for activities related to drug trafficking, while ordinary business men took part by taking care of the money laundering as 'white collar criminals'.
The relationship between black collar and white collar criminals in respect to money transfers had to be built upon trust (which can be seen as a social norm among the members of the network), as formal contracts are not effective in such covert organisations. For some reason, this trust was corrupted at some point, which triggered a process of spreading mistrust, extortion, violent actions (in a range from intimidation to murder) and finally the complete collapse of the criminal network.
The dynamics of this collapse are of special interest to the stakeholders from police authority, as the exact mechanisms could not be understood completely in the investigations and are, thus, subject of this simulation modelling initiative.
A detailed description of this case is available in publications of Neumann and Lotzmann (2014) and (2015).

Process design

The design of the process has to consider a number of different aspects, which are shown in the background lanes of Figure 1.
The first aspect, the core of the process - the model development - is divided into sub-processes for *conceptual analysis* and *simulation modelling*, which are overlapping but in a specific temporal order. Starting with the availability of documents in the evidence base, the analysis has to be performed in the four steps (GLODERS consortium 2014):

- *data preparation* for ordering and standardising the textual data for further processing. This might include unifying the data from different file formats, data encoding and language translation.
- *concept identification*, i.e. the (automatic or manual) extraction of meaningful words and phrases.
- *relationship identification*, which starts with a categorisation of the identified concepts, followed by an extraction of relationships between the categories.
- *concept network analysis*, based on the identified relationships, e.g. by appropriate visualisations or a systematic indicator-based network analysis based on social network analysis (Wasserman and Faust, 1994).

The first design steps for simulation modelling can be started already when the preliminary results from the relationship identification phase are available, the actual simulation modelling takes place after the conceptual analysis process, and eventually leads to validated and analysed simulation results.
The second aspect is a *theory of normative agents* which drives the entire analysis and modelling process. A normative agent is capable of normative reasoning, i.e. to memorise (social or legal) norms and use them for decision making. Foundations of the applied theory were originally developed in previous research conducted in the EMIL project (Conte et al. 2014), and extended and concretised within GLODERS (Andrighetto et al. 2013).
In contrast, the third aspect regards the practitioners' perspective of the *stakeholders*, who provide the empirical domain knowledge and have a significant interest in the activities and results of the different stages of the development process.
The flow chart in Figure 1 gives further details of the specific activities carried out during the development of the particular use case this paper focuses on, together with information about the applied tools.
As mentioned above, the content of the evidence base are basically police interrogation protocols and were contributed by stakeholders. These files had to be translated into English language using translation software, followed by a manual correction of errors caused by OCR and translation (first round for the qualitative analysis, and a second round for the later quantitative analysis).
As starting point for the conceptual analysis process, a

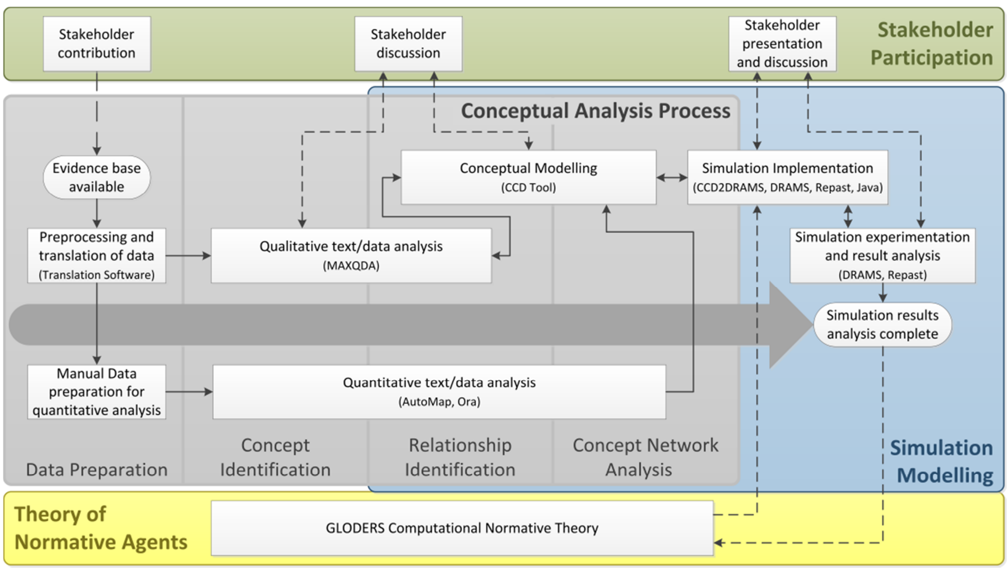

Figure 1: Overview of the evidence-based modelling process (background boxes represent aspects of the process; the flowchart highlights the activities, with solid lines showing the process flow and dashed lines depicting influential relationships)

qualitative text analysis was performed in the sense of a grounded theory approach (Glaser and Strauss 1967; Corbin and Strauss 2008), using the MAXQDA software (www.maxqda.de), a standard tool for Computer assisted Qualitative Data analysis (CAQDAS). The results from this analysis were directly transferred into a Consistent Conceptual Description (CCD; Scherer et al. 2013), which contains concept representations of the entities relevant for the simulation, like actors, (mental and physical) objects, the relations between both kinds of entities, and - most importantly for this stage of modelling - the description of the dynamics constituted by the actors' behaviour by means of condition-action sequences (Neumann and Lotzmann 2015). The CCD, the type of representation used for the conceptual modelling of this case, was developed in the OCOPOMO project.

The process of qualitative analysis and conceptual modelling was heavily interwoven; insights from both activities influenced each other manifoldly. In particular the annotation feature of the CCD (for enabling empirical traceability from model elements back to the evidence; Lotzmann and Wimmer 2013) fertilised the analysis process. Also stakeholders were involved in these two activities.

The (later started) quantitative analysis eventually provided further information for the CCD, especially results of the concept network analysis provided some further insights.

The CCD was then used to develop the simulation model, starting with a model-to-code transformation (CCD2DRAMS) and subsequent implementation as a declarative rule-based model in DRAMS (both tools developed in OCOPOMO; explained further below, around Figure 5) and Java, using an agent architecture derived from the GLODERS computational normative theory, and running in a Repast (North et al. 2006) environment. Simulation experiments were performed, firstly for verification purposes, afterwards productive runs generating results to be analysed and presented to the stakeholders. It is expected that the final results will also have some impact on the normative theory.

In the following sections, the qualitative and quantitative analysis, the conceptual modelling, the simulation modelling and the experimentation will be presented in detail, showing the approach used concretely for the selected use case.

CONCEPTUAL ANALYSIS PROCESS

For the conceptual analysis process two facets have to be regarded, on the one hand qualitative text analysis, on the other hand quantitative text mining techniques. Both approaches are designated for informing the development of a conceptual model, and - as mentioned above - are partly interwoven with the latter task. The process and methods used for the conceptual analysis are related to publications of Diesner and Carley (2005) and Aggarwal and Zhai (2012); details of this particular approach are given in (GLODERS consortium 2014).

Qualitative analysis and conceptual modelling

Figure 2 shows the qualitative analysis and associated conceptual modelling step-by-step. After the initial data preparation (as described in the previous section), a MAXQDA-based analysis could be performed. Relevant text phrases for the main part of the model - the collapse of the criminal network - were identified, concepts and codings developed. Afterwards, the concepts were annotated with related text phrases, resulting in a preliminary concept model, which was discussed with the stakeholders in order to ensure the empirical and practical relevance of the concepts. This discussion lead to a revision of the concept model, based on newly identified relevant text phrases.

After the concept model was agreed, in a discussion between analysts and modellers, concepts relevant for and realisable in a simulation model were identified and recorded in a CCD, firstly as a coarse actor-network diagram (static aspects), followed by the development of a condition-action diagram (dynamic aspects; Neumann and Lotzmann 2014). This conceptual model was compared with the evidence in order to achieve a state of inter-coder reliability, which required several iterations of the involved process steps (i.e. development of new concepts/codings and incorporate these into the CCD).

Once the inter-coder reliability was reached, the CCD content was enriched with annotations of the related text

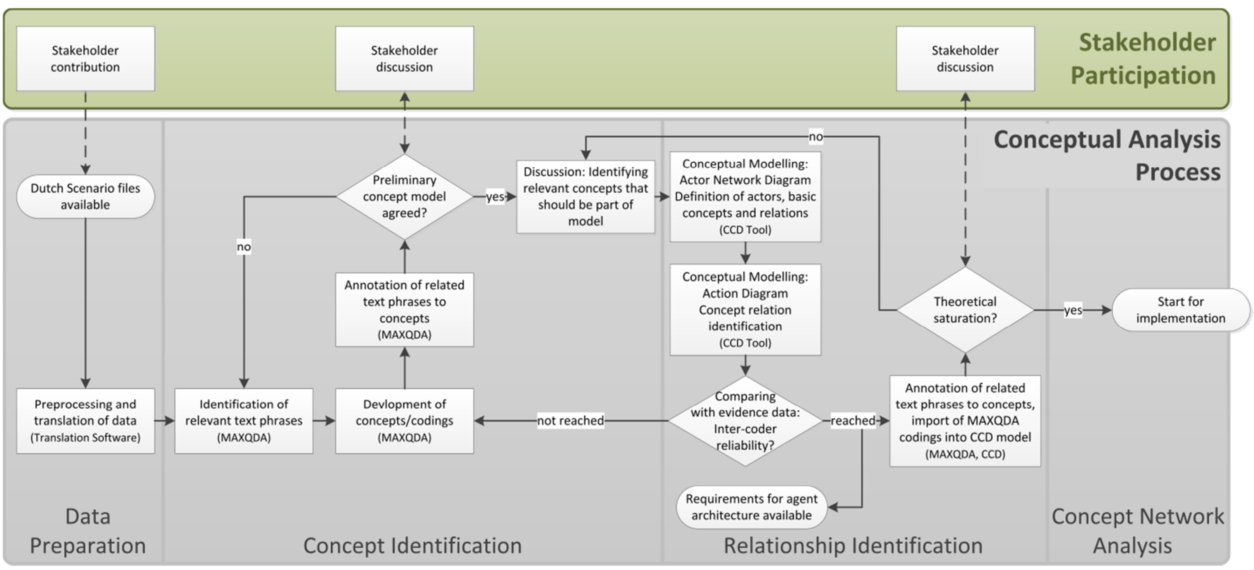

Figure 2: Details of the qualitative analysis and conceptual modelling process

phrases and MAXQDA codings, until a level of theoretical saturation (Corbin and Strauss 2008) was reached (which again was a recursive process) and the validity was ensured by consulting stakeholders. At the same time, the development of the normative agent architecture for this particular model could be started, as the concrete requirements for this case had been extracted from the conceptual model developed so far.

After completion of these two tasks, the actual process of implementing the simulation model could be started.

Figure 3 shows an example of one of the artefacts derived from this process, a part of the CCD action diagram. Here, the consequences of an event that one of the members of the criminal network becomes disreputable due to some action (formulated in the bottommost condition in the diagram). This action is basically linked to a violation of the norm of trust within the network, which could be for example

- (suspected) disloyal behaviour, which might constitute a crystallising kernel of mistrust and potentially triggering a spiral of aggression and violence; or
- an act of (unmotivated) aggression, which might happen in the course of spreading mistrust and violence.

In any case, both black and white collar criminals in the network are able to perceive this condition and might react with some kind of aggression in order to punish the disreputable member, i.e. to sanction the norm violation. This action of aggression can either be intimidation or violence, both in several degrees of severity.

Result of this aggressive action is a condition in which the aggression has been recognised by the affected member, who then tries to find out the reason for this incident in order to react adequately (not shown in the diagram snippet).

In this example, the action 'perform aggressive action against member X' should be pointed out here, as it involves a complex decision process of the agents representing the related criminals. The aspects of this decision process that have been discovered in the data analysis, are preserved as annotations for this CCD action. The resulting implementation of this action that has to take all these aspects into account, is discussed in the simulation modelling and testing section of the paper.

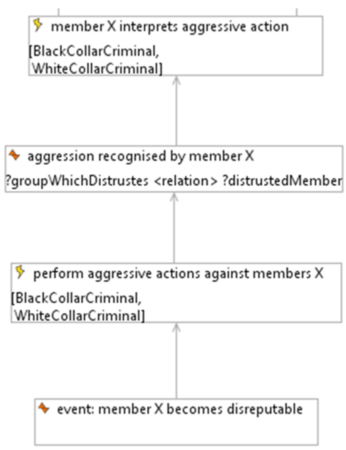

Figure 3: Example of a CCD action diagram

Quantitative analysis

The quantitative analysis process aligned for this use case is outlined in Figure 4. The data already prepared for the qualitative analysis had to undergo another manual clean-up, in order to achieve the best possible performance of the applied tools.

The analysis was basically performed sequentially, using text-mining methods like tokenisation and stemming (Aggarwal and Zhai 2012). Starting with AutoMap for identifying concepts and finding co-occurrences, these results were imported into ORA for a measurement-based network analysis. Details of this work are given in (Sartor 2014), for information on the tools used, see http://www.casos.cs.cmu.edu/tools/.

The results of this analysis could then be incorporated into the CCD and serve as input for extending the simulation model.

For the described use case, the qualitative analysis performed beforehand revealed most of the relevant and practicable details for modelling the collapse of the criminal network, as approved by the stakeholders. However, a quantitative research was carried out in order to deal with numerical complexity (in contrast to qualitative methods, which are able to cope with cognitive complexity). This led to discovering further details of the construction of the network, its members and their specific roles and activities. Not only the core group of black and white collar criminals was investigated, but also other individuals were in focus, e.g. 'straw men', charged with the task of obscuring money laundering.

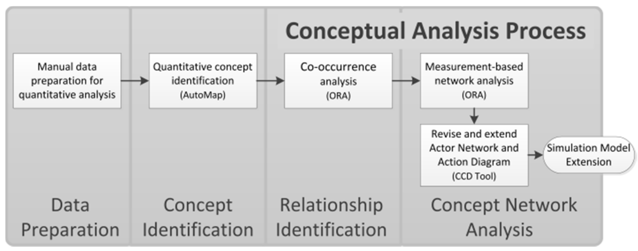

Figure 4: Details of the quantitative analysis process

As it turned out, these findings added quite few new details to the model of collapse, but shed new light on the 'every day business' of drug trafficking and money laundering.

For future extensions to the simulation model, this kind of information will become valuable.

SIMULATION MODELLING AND TESTING

The process of concept analysis of empirical data described in the previous section provides a basis for the process of developing the simulation model, as shown in Figure 5.

In a first step, the CCD was automatically transformed into a skeleton of the simulation model code. Here a software called CCD2DRAMS was used, which is a model-to-code transformation tool that transforms conceptual model elements into code for formal models, in this case in declarative agent rules for the distributed rule engine DRAMS, together with supporting Repast-based Java code.

DRAMS - as technological basis for the simulation model - supports programming and running simulation models while enabling traceability of empirical evidence in agent rules and simulation results (Lotzmann and Wimmer 2013).

These generated rule stubs were then elaborated. This implementation also took other aspects into account, in this case the agent architecture previously designed on base of requirements that have arisen from the conceptual modelling, and the GLODERS computational normative theory.

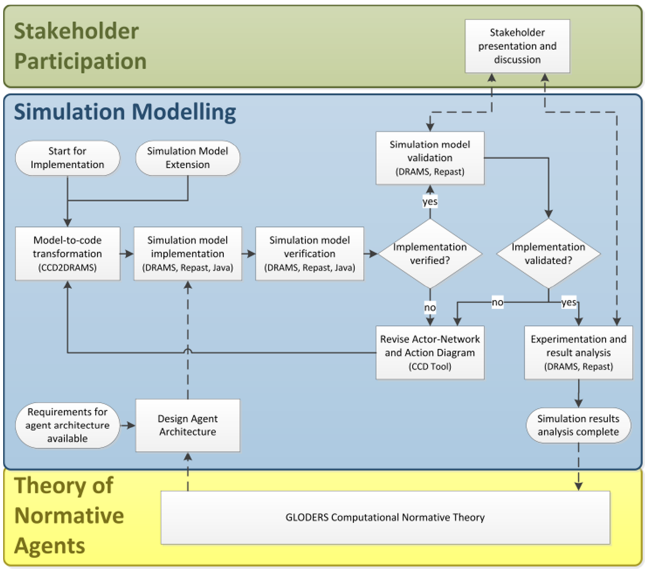

Figure 5: Details of the simulation modelling process

The implementation was an iterative process, associated with persistent testing and inferences for the implementation and partly also for the CCD, when certain crucial model aspects (such as parameters) could not be verified.

After the model was successfully verified, a validation took place, involving again the stakeholders in discussion of the preliminary results. Here again the need for modifications might become apparent, which then would trigger a further adaption or extension of both the conceptual as well as the formal model.

Finally, with the validated model 'productive' experiments produce results for the subsequent analysis, which might give feedback and new insights to the stakeholders, but also regarding the normative theory.

Note that the modelling process is at this point of time at the stage of (repeated) model verifications.

The DRAMS simulation model implemented in this process works in a way quite similar to the representation of dynamic aspects in the CCD action diagram: if certain conditions are satisfied, here expressed by the availability of certain facts stored in fact bases, a number of actions can take place, here by triggering certain agent rules.

A rule consists of a condition part (called left-hand side, LHS) and an action part (called right-hand side, RHS). The clauses which formulate the conditions in the LHS are logically evaluated, and in the case of a successful evaluation, the RHS is triggered, which means that the rule fires and produces new facts (hence, triggering further rules) and outputs (textual and numerical simulation outcome; Lotzmann and Wimmer 2013). However, the granularity of the actual agent rules can be quite different from the construction in the conceptual model. For example, Figure 6 shows the DRAMS rules and facts which implement the action 'perform aggressive action against member X' as part of Figure 3. This so called data-dependency graph is automatically generated by DRAMS on the base of the implemented model. The oval nodes represent the facts (green with facts initially available from model-to-code transformation, red with facts that are produced at some point in the simulation run). It can be directly seen that the complex decision process hidden in the single CCD action results in the implementation of a number of different rules which involve different stages of decisions, in which the agent determines which aggressive action to plan and perform against the disreputable member. For example, a black collar criminal agent could 'plan' a violent action (like beating up the victim) in an emotional outburst (i.e., the opposite of a rational decision), which then triggers another rule that actually performs the aggression against the victim. The rules that would then be triggered (and are not shown in the figure) would reflect reasoning about the cause of the attack, i.e. the recognition whether the aggression has to be interpreted as a sanction, or as an arbitrary event. This sanction recognition mechanism will eventually impact the normative theory.

As mentioned above, the agent architecture used for the simulation model is embedded in a theory of normative agents. Regarding the technical realisation, a Java framework for normative reasoning was developed within the GLODERS project. Since DRAMS is also implemented in Java, it entails the functionality to import Java code in order to extend the functionality available within declarative rules. The normative reasoning is put in motion in the simulation by integrating the normative process framework with the DRAMS environment.

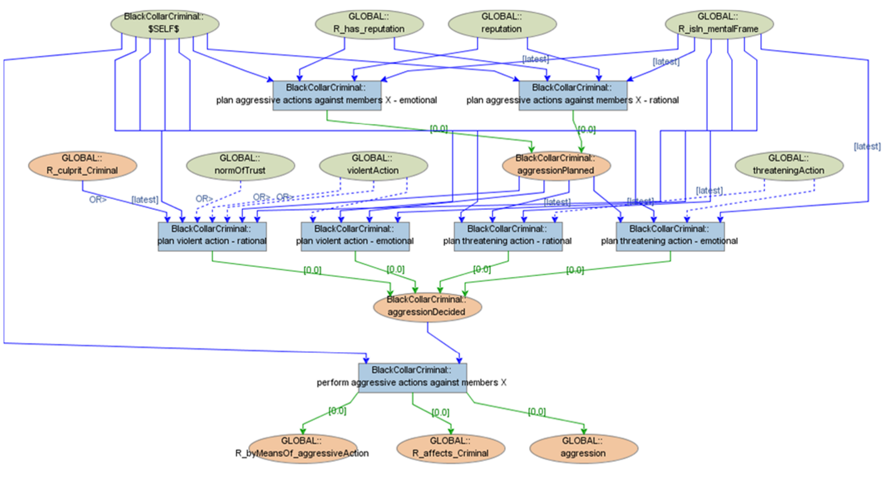

Figure 6: Example of a DRAMS data dependency graph

SIMULATION EXPERIMENTATION

In order to run experiments with the simulation model, parameters have to be set according to the experiment design. Some of these parameters, especially regarding the model structure (types of agents with certain attributes, types of aggressive actions etc.) must be configured in the conceptual model. The CCD allows to define instances for actors, objects and relations, that constitute the simulation world. Other parameters (like number of agents, probabilities for certain decision outcomes) are set by parameter files or via user interface.

Besides visualising e.g. time series of numerical variables and associated statistics, the results of simulation runs are presented as simulation logs, in which 'stories' of the events are told in a human-readable way.

An example of such a textual output generated by a model run is printed in Figure 7. The events that happened within the first two ticks when one of the criminals lost his reputation can be tracked in that text. Some meta information - the rule which produced the output and the time when the rule fired, e.g. '`1.0:0 [global.external event: make random criminal disreputable]`' - are attached as a prefix to the actual outcome. The clumsy language in some of the statements is due to the automatic assembling of the phrases by the simulation program, e.g. keywords describing concrete actions are inserted in predefined text frames.

The experiments to be conducted with the model have to be designed in a way that the stakeholders' issues are addressed, i.e. relevant questions of the stakeholders are treated. The model sketched in this paper is designed to cope with a variety of such questions. On the one hand, it is capable to 'replay' the events that happened in reality in the course of the collapse of the criminal network, on the other hand it allows to explore effects of (slightly) modified underlying conditions. For example, an interesting question would be whether the collapse would have taken place in a similar manner if one of the highly reputable key persons in the criminal network would not have been jailed at the time the spreading of mistrust began, i.e. whether the reputation and mediating skills of this individual would have make the collapse unlikely. This question is related to a more general question from a theoretical point of view: the importance of hierarchy for the stability of criminal organisations, accompanied with the power of enforcement of social norms within such networks.

```
*** 1.0 ***
1.0:0 [global.external event: make random criminal
disreputable] "Criminal ReputableCriminal-1 has become
disreputable due to an unknown (external) event."
1.0:1 [global.decline reputation of member X] "Reputation of
criminal ReputableCriminal-1 drops from high to low ."
Time to run: 0.133 seconds

*** 2.0 ***
2.0:1 [ReputableCriminal-5.plan aggressive actions against
members X - rational] "ReputableCriminal-5 has rationally
decided to threat ReputableCriminal-1 due to the latter's low
reputation."
2.0:2 [ReputableCriminal-5.plan threatening action - rational]
"ReputableCriminal-5 decides to threat ReputableCriminal-1 by
putting a gun into the stomach ."
2.0:3 [ReputableCriminal-5.perform aggressive actions against
members X] "ReputableCriminal-5 is attacking
ReputableCriminal-1 by putting a gun into the stomach."
2.0:4 [global.evaluate consequences of aggression for member]
"ReputableCriminal-1 has suffered an attack ( putting a gun
into the stomach ) and is alive"
Time to run: 0.233 seconds
```

Figure 7: Example of a simulation log

In order to find answers to such questions, the traceability is a valuable instrument in that respect. One of the available analysis tools capable to visualise these traces is the Model Explorer tool, a plug-in for the DRAMS software primarily dedicated for model debugging and code verification, but also useful for validation purposes. Figure 8 shows a screenshot of an actual firing of the rule "plan a violent action - emotional", in order to give an idea how a user interface for analysing simulation results could be designed. On the right hand side, the log file of the simulation run is placed. On the left hand side, the CCD and the corresponding empirical texts are displayed. Both elements, the simulation results and the empirical basis, are related by a visualisation of the rules that fired and, hence, transformed empirical evidence into simulation results.

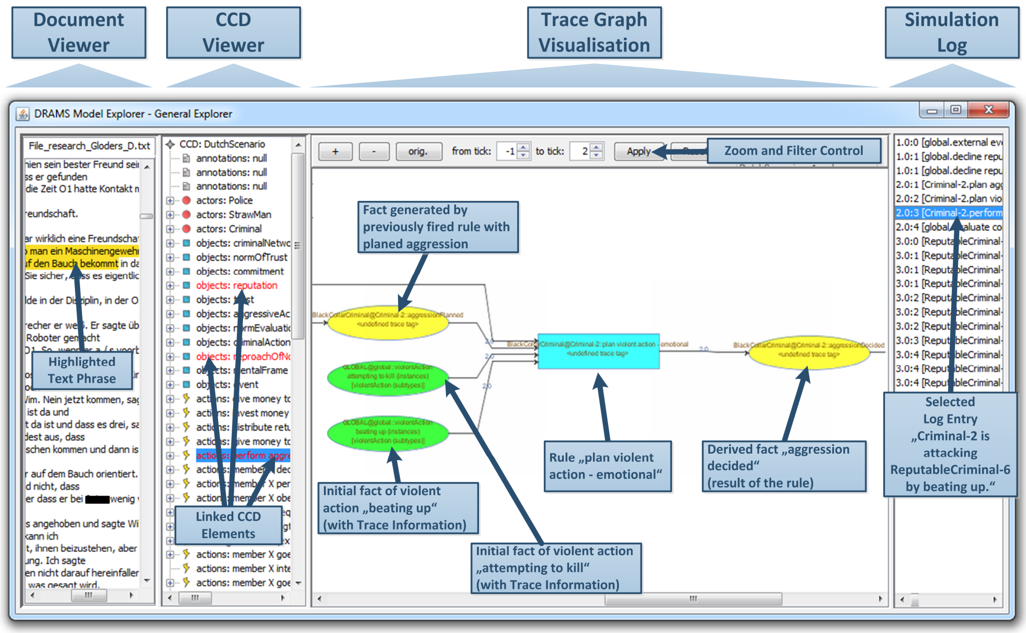

Figure 8: Screenshot of the Model Explorer Tool

CONCLUSIONS AND FUTURE WORK

Aim of this paper is to present and discuss a process, which supports the transformation of empirical textual evidence into agent-based simulation models, and which at the same time allows to trace results from model executions back to the relevant parts of the empirical data. Foci of the paper are the different phases of the modelling process, from qualitative and quantitative data analysis, conceptual modelling, simulation implementation through to experimentation and result presentation. The paper tries to illustrate this approach by a successfully carried out modelling task.

The insights obtained by performing this task revealed a great potential impact of such a process regarding quality and trustworthiness of models, ensured by the close participation of stakeholders and non-simulation but domain experts.

However, next steps to go are to discuss the results with the stakeholders, and make them familiar with the simulator and the simulation method in general. For this purpose a stakeholder workshop is scheduled.

Furthermore - in a mid-term perspective - it would be important to open the process to a broader range of methods and techniques, e.g. for different data analysis approaches, simulation paradigms and techniques, and more sophisticated simulation result analysis procedures which entail the full impact of the traceability means. In this context, other modelling projects need to be carried out, adapting and possibly enhancing the process.

This extension should also consider a technical perspective, i.e. the tighter integration of the different software products used at the single process steps in an integrated toolbox. While conceptual modelling, model transformation and implementation of declarative rule-based models could easily be extended by initial qualitative/quantitative data analysis from a methodological point of view, on the tool perspective this integration is much more difficult. It turned out that the landscape of qualitative and quantitative data analysis tools is eminently heterogeneous. A tool evaluation performed in this context showed that, for the kind of available data, the necessary analysis functionalities are spread over a broad range of commercial and open source tools. This integration should be addressed in future research activities.

ACKNOWLEDGEMENTS

The research has been undertaken as part of the GLODERS project, co-funded by the European Commission under the 7th Framework Programme. This publication reflects the view only of the authors and the project consortium, and the Commission cannot be held responsible for any use, which may be made of the information contained therein.

REFERENCES

Aggarwal, C. C. and C. X. Zhai (eds.). 2012. *Mining text data*. New York: Springer.

Andrighetto G.; J. Brandts; R. Conte; J. Sabater; H. Solaz; and D. Villatoro. 2013. “Punish and Voice: Punishment Enhances Cooperation when Combined with Norm-Signalling”. In *PLoS One,* vol. 8 (6), 1-8.

Barreteau, O. 2003. “Our companion modelling approach.” In *Journal of Artificial Societies and Social Simulation*, vol. 6(1). URL: http://jasss.soc.surrey.ac.uk/6/2/1.html.

Coen, C. (2009). “Simple but not simpler. Introduction CMOT Special Issue–simple or realistic.” In *Computational and Mathematical Organization Theory,* vol. 15(1), 1-4.

Conte, R., Andrighetto, G., Campennì, M. (2014). *Minding norms: mechanisms and dynamics of social order in agent societies*. Oxford: Oxford university press.

Corbin, J. and A. Strauss. 2008. *Basics of qualitative research. 3nd edition*. Thousand Oaks: Sage.

Diesner, J. and K.M. Carley. 2005. “Revealing Social Structure from Texts: Meta-Matrix Text Analysis as a novel method for Network Text Analysis”. Chapter 4 in *Causal Mapping for Information Systems and Technology Research: Approaches, Advances, and Illustrations*, V.K. Narayanan and D.J. Armstrong (Eds.). Harrisburg, PA: Idea Group Publishing, 81–108.

Dilaver, O. 2015. “From Participants to Agents: Grounded Simulation as a Mixed-Method Research Design.” In *Journal of Artificial Societies and Social Simulation, vol.* 18(1). URL: http://jasss.soc.surrey.ac.uk/18/1/15.html.

Edmonds, B. and S. Moss. 2005. “From KISS to KIDS: An anti-simplistic modelling approach.” In *Multi-agent-based simulation 2004,* P. Davidsson (Ed.). Heidelberg: Springer, 130–144.

Glaser, B. and A. Strauss. 1967. *The discovery of Grounded Theory. Strategies for qualitative research*. Chicago: Aldine.

GLODERS consortium. 2014. *D 2.1 Analysis of Content and Structure of Data leading to a shared Ontology. Deliverable 2.1. FP7 Project GLODERS (Global Dynamics of Extortion Racket Systems)*. URL: http://www.gloders.eu/images/Deliverables/GLODERS_D2-1.pdf [last accessed 2015-02-24].

Lotzmann, U. and M. Wimmer. 2013. “Evidence Traces for Multi-agent Declarative Rule-based Policy Simulation”. In *Proceedings of the 17th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications (DS-RT 2013)*. IEEE Computer Society, 115-122.

Neumann, M. 2015. “Grounded Simulation.” In *Journal of Artificial Societies and Social Simulation, vol.* 18(1). URL: http://jasss.soc.surrey.ac.uk/18/1/9.html.

Neumann, M. and U. Lotzmann. 2014. “Modelling the Collapse of a Criminal Network”. In *28th European Conference on Modelling and Simulation, ECMS 2014*, F. Squazzoni, F. Baronio, C. Archetti and M. Castellani (Eds.). European Council for Modeling and Simulation, 765-771.

Neumann M. and U. Lotzmann. 2015. “Ein korruptes Chaos: Gewalteskalation in illegalen Organisationen ”. In *Sitiationen der Gewalt*, C. Equit, A. Grönemeyer and O. Schmidt (Eds.), Wiesbaden, VS-Verlag.

North, M.J.; N.T. Collier; and J.R. Vos. 2006. “Experiences Creating Three Implementations of the Repast Agent Modeling Toolkit”. In *ACM Transactions on Modeling and Computer Simulation 16*, No. 1 (Jan.), 1-25.

Sartor, N. (2014). *Eine Netzanalyse der Organisationsstrukturen im Bereich Geldwäsche*. Master thesis, University of Koblenz-Landau.

Scherer, S.; M. Wimmer; and S. Markisic. 2013. „Bridging narrative scenario texts and formal policy modeling through conceptual policy modelling”. In *AI and law* 21, 455–484.

Wasserman, S. and K. Faust. 1994. *Social network analysis. Methods and applications*. Cambridge, New York: Cambridge University Press.

Yang, L. and N. Gilbert. 2008. “Getting away from numbers: Using qualitative observation for agent-based modelling.” In *Advances in complex systems,* vol. 11(2), 175–185.