In: Nolte, A.; Prilla, M.; Lukosch, S.; Kolfschoten, G. and Herrmann, T.: Proceedings of the 1st International Workshop on Collaborative Usage and Development of Models and Visualizations at the ECSCW 2011 (CollabViz 2011)

Towards Combining ThinkLets and Dialogue Games in Collaborative Modeling: an Explorative Case

S.J.B.A. Hoppenbrouwers[1] and W. van Stokkum[2]
[1]Radboud University Nijmegen, the Netherlands
and CRP Henri Tudor, Luxemburg
stijnh@cs.ru.nl
[2]Everest B.V., the Netherlands
w.van.stokkum@everest.nl

Abstract. We present a next step in our ongoing effort to conceive innovative support approaches for collaborative modeling. We propose to blend the well-established Collaboration Engineering approach (rooted in CSCW) with the Dialogue Game approach (rooted in Conceptual Modeling), viewing the second as a specialized extension of the first, and describing how they can complement each other. We hope to eventually link not only the approaches, but also the two fields. We provide a small but realistic illustration of our proposal at the hand of a real, industrially used elicitation pattern from knowledge modeling, and briefly show how this pattern can be wrapped up as an 'm-thinkLet'.

Introduction

In many uses of collaborative modeling, e.g. in business engineering (den Hengst & de Vreede, 2004), knowledge engineering (Hoppenbrouwers, Schotten, & Lucas, 2010), problem structuring (Vennix, 1996), and enterprise engineering (Barjis, 2009), collaborative modeling with stakeholders untrained in modeling is a required and common practice, but also a continuous challenge, referred to as the 'knowledge acquisition bottleneck' (Hoppenbrouwers et al., 2010).

In the field of collaborative modeling (Renger, Kofschoten, & De Vreede, 2008), most work focuses on the collaborative creation and validation of model diagrams, using some standard modeling language (for example, UML activity diagrams: (Rittgen, 2007)). A different approach, which this paper is an exponent of, concerns more focused, 'smaller' conceptualizations that help gather and communicate highly to-the-point, well structured information that can be the basis for *derivation* (manually or possibly automatically) of more abstract, 'technical' models (Hoppenbrouwers, 2008; Hoppenbrouwers et al., 2010).

Once we move away from the 'collaborative diagram drawing' approach and into more limited and focused conceptualization (closer to the stakeholders' familiar concepts and requiring less skill in dealing with abstract syntax and complex visualizations and verbalizations), we can also move towards more closely guided, wizard-like conceptualization support (Hoppenbrouwers, Weigand, & Rouwette, 2009). We thus, in the long run, work towards the creation of a coherent library of well focused 'modeling games': rule-based, goal-driven interactive procedures that do not involve more than a few meta-concepts each and should be relatively easy to 'play' for stakeholders untrained in formal modeling (Wilmont, Brinkkemper, van de Weerd, & Hoppenbrouwers, 2010).

Such 'conceptualization games' bear considerable resemblance to the thinkLet concept central in Collaboration Engineering or CE (de Vreede & Briggs, 2005; Kolfschoten, Briggs, de Vreede, Jacobs, & Appelman, 2006), and can in fact be seen as a specialized extension of that approach. However, as will be explained in the next section, some additional properties are to be added to thinkLets as they (also) become Dialogue Games (DGs). The DG approach originated in the field of conceptual modeling, whereas CE concerns collaboration more in general, yet in the specific context of collaborative interaction support (in particular, CSCW). We hope to link not only the approaches, but ultimately also the two fields.

ThinkLets and Dialogue Games

The Dialogue Game (DG) approach to collaborative modeling is rooted in a theoretical view on modeling as a *conversation* (Veldhuijzen van Zanten, Hoppenbrouwers, & Proper, 2004). Detailing this line of thinking led to a framework in which the core concepts are Rules, Interactions, and Models (RIM): **R**ules both drive and constrain conversational **I**nteractions that include propositions, but also argumentation *about* those propositions. A set of propositions as accepted by the modelers at some point in time constitutes a current **M**odel. For an elaborate explanation of the RIM framework, see (Ssebuggwawo, Hoppenbrouwers, & Proper, 2009). Interactions include conversational moves like arguing for or against a proposition, agreeing, disagreeing, and of course putting forward or withdrawing a proposition.

From the rule-based RIM approach, it is a small step to viewing modeling sessions as enacted games (instantiations of a game type). In addition, there is a theoretical link between the RIM approach and '*dialogue games*', a known concept in Argumentation Theory (Eemeren et al., 1996).

Let us now consider the CE approach (involving thinkLets) and see how this approach relates to the DG approach to collaborative modeling. Please note that lack of space prevents us from providing a full scale, detailed comparison between the CE and DG approaches here; we intend to do this elsewhere, including identification of overlap between existing thinkLets and (parts of) Dialogue Games. Indeed we know such overlap exists. However, our strategy is to first focus on the creation of playable game implementations; analysis and (re)-use of generic patterns (thinkLets) in these games will have to come later.

The CE concepts we refer to below are based on (Kolfschoten et al., 2006). Symbolical of the overlap between the two approaches, we refer to 'm-thinkLets': a (still mostly fictional) class of thinkLets for use in collaborative modeling and compatible with the structure of DGs.

In (Kolfschoten et al., 2006), thinkLets are defined as "named, packaged facilitation techniques that create predictable, repeatable patterns of collaboration among people working towards a goal". In (Hoppenbrouwers et al., 2009), collaborative modeling is characterized as a "goal-driven interactive activity that requires freedom of action and decision within clearly set boundaries." Games are typically also such activities. A similar direction is suggested in (Kolfschoten et al., 2006)) by shifting from the use of complete and rather detailed, restrictive 'scripts' as part of specifying thinkLets, to defining *rules*. Though they do not explicitly refer to 'games', from the DG/RIM perspective even classic thinkLets *are* games, of a sort.

In dealing with the optimal trade-off between constraint and freedom in guiding interaction, much can be learned from game dynamics. In addition, taking the game metaphor seriously suggests some interesting possibilities: the use of advanced interfacing from gaming to make collaborative interaction more accessible and engaging; even the use of devices like score systems or local competition embedded in over-all collaboration (Hoppenbrouwers et al., 2009).

The DG approach recognizes the long term goal (also highly prominent in CE) of removing the facilitator as much as possible (disintermediation), yet it currently focuses on simplifying and structuring the facilitator's role rather than removing it. A DG for modeling is typically viewed as two entwined games with distinct sets of goals and rules: one (or more) for the stakeholder-participants, one for the facilitator-participants. Again this merges the notion of 'rules' with the notion of 'script', including the facilitator as a role in the game. Such a setup was successfully executed in a pilot DG for Group Model Building, transforming a script into a DG (Hoppenbrouwers & Rouwette, 2011).

In modeling (as opposed to generic collaboration), a key notion is that of a *meta model* or modeling language. Though this aspect is in principle covered by thinkLet design concepts, it could benefit from additional, further specialized views from the DG approach. The *pragmatic* focus of a DG (the intended use of the conceptualization it renders: its desired resulting contents) is driven by focus questions; its *semantic-syntactic* focus (the modeling language or conceptual format of the result) constrains the formulation of focused answers (Hoppenbrouwers & Wilmont, 2010). Small sets of meta concepts used in modeling can thus be deliberately introduced in m-thinkLets, aiding their pragmatic and semantic-syntactic focus.

CE uses the concept of “parameters” of thinkLets: content-specific variables, for example focus questions. One could view such parametrization as an important aspect of the development of m-thinkLets. However, the creation of m-thinkLets would involve the setting of parameters that would still be generic for a certain flavor of modeling, e.g. ontological modeling, process modeling, and so on. Indeed, m-thinkLets require a specific, focused approach to the use of parameters extending into ‘syntax setting’ for m-thinkLet results.

CE covers ‘moves of the game’ that relate to the rendering of results of thinkLets. *Discussion* is explicitly included as a possible ‘action’ in thinkLets, but CE does not guide, constrain, or log its ‘mechanics’. Contrarily, the DG approach considers the typical interactions of discussion and argumentation as discrete ‘moves of the game’ (Hoppenbrouwers & Rouwette, 2011). Logging all “discussion moves” and making them accessible both during and after the game is standard. Possibly, CE in general might benefit from such a mechanism.

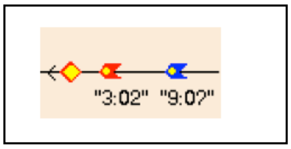
Having explored key similarities and differences between CE and the DG approach, let us consider a realistic example of a potential DG based m-thinkLet.

Example: The ‘Weighted Factor Elicitation Game’

An exemplary ‘m-thinkLet’ interaction pattern was created in context of a project in which a radical new distributed model was conceived for scheduling Dutch railway traffic (van Stokkum, 1999). The pattern involved was applied in a one-and-a-half hour collaborative modeling session with three domain specialists of Dutch Railways, and a facilitator. A role playing setup was used to elicit the weighed factors that influence the creation of scheduling conflicts between trains.

The facilitator (a knowledge engineer) initiated the session by introducing a limited set of scenarios that can lead to a conflict. These scenarios were presented by schematic diagrams (Fig. 1).

The diamonds in Fig. 1 represent junctions. The other icons represent trains. The goal of the game is for the players to set parameters such that, for a specific scenario, there is a given p% chance (e.g. 75%) that the trains will raise a conflict (i.e. arrive at the same time) at the junction. During the game, the facilitator actively varies scenario details like the types of trains involved (e.g. length, load) or events occurring (e.g. wind conditions, engine failure).



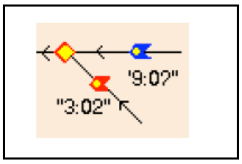


Figure 1. Two of the scenarios in which two trains could be arriving too close together at one infrastructural railtrack point

For example: “let domain expert 1 be the red train. This red train is a long cargo train carrying a heavy load. Domain expert 2 is the blue train which is IC train with high priority. Domain expert 3 is a junction that will assess continuously the chance of collision. Assignment: for this situation, collaboratively conceive and set factors so that there is a 75% chance the trains collide”. The actual, utilitarian goal of the game is to collaboratively define a stable set of factors influencing the chances of collisions taking place. Factor types thus elicited included *speed*, *maintenance record*, *weather influences*, *weight*, *type of engine*, *priority of passengers* and *cargo*; weights (high/low) indicated the importance of the factors.

The domain experts involved had no experience in creating formal models. The described session was one in a series of nine interrelated sessions, each of a similar focused nature. In each session the focus (both pragmatic and semantic-syntactic) was set differently to address a specific aspect: the train, the infrastructural points, creating conflict, creating a plan to prevent a conflict, determining a cost function to evaluate a plan, decision making on plans, determining follow-up conflicts, define a stop criterion for evaluating uncertain follow-up conflicts. By breaking up the problem into small, focused sessions, in the end a very complex distributed scheduling system was collaboratively modeled, without any ‘comprehensive diagram drawing’ (in fact, such a diagram would have too complex to draw in the first place: it was represented as a set of mathematical formulae).

The same patterns have later been reapplied in other projects in need of a real time distributed workflow scheduling solution. For example, the patterns have been used to develop a system for scheduling ground operations at Zaventem airport, scheduling autonomous operating robots in Rotterdam’s largest container handling terminal port, creating simulations to solve traffic jam problems in Holland and for developing an order picking system for distribution centers of a Dutch super market chain.

The technique presented above is an excellent example of a ‘Focused Conceptualization’ or ‘FoCon’ as introduced in (Hoppenbrouwers & Wilmont, 2010). Specifications of FoCons are somewhat similar to conceptual designs for thinkLets, but they were developed strictly in context of collaborative modeling.

FoCon analysis as an instrument concerns questions like: "What goes into a FoCon situation, in terms of existing information and people (including their concerns, knowledge, and skills)"; "What is the intended output of a FoCon situation, in terms of pragmatic goals, conceptual (semantic-syntactic) constraints set, and the required level and sort of agreement between people", "what focus questions are used, and what explicit instructions are to be given by the facilitator, in which situation", and "what rules govern the required or limited interaction between players, in view of a current focus question". Clearly, a similar analysis could be applicable in a thinkLet context. The main points of a FoCon analysis of the m-thinklet described above are given in Table I below:

"IN"	Info	Various given scenarios and given chances of collision
	Concepts	Trains, junctions, situations (diagrams); properties of trains, partly based on results of ongoing elicitation; given chance of collision (P-value, e.g. 0.75)
	People	Train traffic management experts, not trained in formal modeling, some system thinking ability, homogeneous professional background
"OUT"	Info (pragm. focus)	List, generalized over all scenarios used, of weighted factors influencing collision risk
	Concepts (sem.-synt. focus)	Factor types, weight for each factor type (high/low impact)
	Social req.	Factors commonly understood and agreed upon
	Argumentation	Arguments raised and accepted/rejected in discussing the factors and their weights
Substeps/ Strategy		*Facilitator*: iteratively set scenario, then discuss factors, then change details of scenario or set new scenario, thus systematically exploring all factors and developing a generic overview; *Players*: assume role of train or junction; for a series of scenarios, provide weighted factors matching a given chance of collision
Interaction Modes in the game		• Focus on shared understanding of scenario • Focus on identifying relevant factors • Focus on determining the weight of a factor

Table I. Overview of the main points of a FoCon analysis and DG outline of the example

We hope the table sufficiently illustrates how a FoCon analysis can serve as a basis for designing both Dialogue Games and m-thinkLets. Note that in the example, 'argumentation' plays a role in the actual elicitation process (arguments can be looked up during a running game and are a source of ideas about factors for the players) but argumentation is also logged for future reference to details in the discussion (otherwise lost). Structure is inherently provided by the DG setup.

We leave out considerations of mappings between m-thinklets, aptly called "transitions" (Kolfschoten et al., 2006), except by stating that such transitions can be direct *mappings* of resulting concepts to models or model views, but also *derivations* (typically by means of logical reasoning) based on concepts found and possibly leading to further *abstraction* thereof (Hoppenbrouwers et al., 2010).

Acknowledgments

This paper results from the Agile Service Development project (*http://www.novay.nl/okb/projects/agile-service-development/7628*), a collaborative research initiative focused on methods, techniques and tools for the agile development of business services. The project consortium consists of BeInformed, BiZZdesign, CRP Henri Tudor, Everest, HU University of Applied Sciences Utrecht, IBM, Novay, O&i, PGGM, RuleManagement Group, Radboud University Nijmegen, Twente University, Utrecht University, and Voogd & Voogd. The project is part of the program Service Innovation & ICT of the Dutch Ministry of Economic Affairs.

References

Barjis, J. (2009). Collaborative, Participative and Interactive Enterprise Modeling. In J. Filipe & J. Cordeiro (Eds.), *Enterprise Information Systems*. Berlin, Heidelberg: Springer Verlag.

de Vreede, G. J., & Briggs, R. O. (2005). Collaboration Engineering: Designing Repeatable Processes for High-Value Collaborative Tasks. *Proceedings of the 38th Hawaii International Conference on System Sciences (HICSS-38)*. Big Island, HI, USA: IEEE Computer Society.

den Hengst, M., & de Vreede, G. J. (2004). Collaborative Business Engineering: a Decade of Lessons from the Field. *Journal of Management Information Systems, 20*(4), 85--113.

Eemeren, F. H. v., Grootendorst, R., Snoeck Henkemans, F., Blair, J. A., Johnson, R. H., Krabbe, E. C. W., et al. (1996). *Fundamentals of Argumentation Theory: A Handbook of Historical Backgrounds and Contemporary Developments*. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Hoppenbrouwers, S. (2008). Community-based ICT Development as a Multi-Player Game. In C. Benoit-Barné, B. H. Brummans, F. Cooren, H. Giroux, A. Létourneau, D. Raymond & D. Robichaud (Eds.), *What is an organization? Materiality, agency, and discourse: a tribute to the work of James R. Taylor*. Montreal: Dept. of Organizational Communication, University of Montreal.

Hoppenbrouwers, S., & Rouwette, E. (2011). A Dialogue Game for Analysing Group Model Building: Framing Collaborative Modelling and its Facilitation. To appear. *International Journal of Organisational Design and Engineering, special issue on Collaborative Modeling*.

Hoppenbrouwers, S., Schotten, B., & Lucas, P. J. F. (2010). Towards Games for Knowledge Acquisition and Modeling. *International Journal of Gaming and Computer-Mediated Simulations, Special issue on AI and Games, 2*(4), 48-66.

Hoppenbrouwers, S., Weigand, H., & Rouwette, E. (2009). Setting Rules of Play for Collaborative Modeling. *International Journal of e-Collaboration (IJeC), 5*(4), 37-52.

Hoppenbrouwers, S., & Wilmont, I. (2010). Focused Conceptualisation: Framing Questioning and Answering in Model-Oriented Dialogue Games. In P. Bommel, S. Hoppenbrouwers, S. Overbeek, E. Proper & J. Barjis (Eds.), *The Practice of Enterprise Modeling* (pp. 190-204). Berlin, Heidelberg: Springer.

Kolfschoten, G., Briggs, R., de Vreede, G. J., Jacobs, P., & Appelman, J. (2006). A conceptual foundation of the thinkLet concept for Collaboration Engineering. *International Journal of Human-Computer Studies, 64*, 611--621.

Renger, M., Kofschoten, G., & De Vreede, G. (2008). Challenges in collaborative modelling: a literature review and research agenda. *International Journal of Simulation and Process Modelling, 4*(3), 248--263.

Rittgen, P. (2007). Negotiating Models. In J. Krogstie, A. Opdahl & G. Sindre (Eds.), *Advanced Information Systems Engineering, 19th International Conference, CAiSE 2007,* (pp. 561-573). Berlin, Heidelberg: Springer.

Ssebuggwawo, D., Hoppenbrouwers, S., & Proper, E. (2009). Interactions, Goals and Rules in a Collaborative Modelling Session. In A. Persson & J. Stirna (Eds.), *2nd IFIP WG8.1 Working Conference on the Practice of Enterprise Modeling (PoEM 2009)*. Berlin, Heidelberg: Springer.

van Stokkum, W. (1999). A Model for Distributed Multi-agent Traffic Control. *Multiple Approaches to Intelligent Systems* (pp. 480--489).

Veldhuijzen van Zanten, G., Hoppenbrouwers, S., & Proper, H. A. (2004). System Development as a Rational Communicative Process. *Journal of Systemics, Cybernetics and Informatics, 2*(4).

Vennix, J. A. M. (1996). *Group model building. Facilitating team learning using system dynamics*. Chichester: Wiley.

Wilmont, I., Brinkkemper, S., van de Weerd, I., & Hoppenbrouwers, S. (2010). Exploring Intuitive Modelling Behaviour. In I. Bider, T. Halpin, J. Krogstie, S. Nurcan, E. Proper, R. Schmidt & R. Ukor (Eds.), *Enterprise, Business-Process and Information Systems Modeling. Proceedings of the 11th International Workshop, BPMDS 2010, and 15th International Conference, EMMSAD 2010, held at CAiSE 2010, Hammamet, Tunisia, June 7-8, 2010* (pp. 182--194). Berlin, Heidelberg: Springer Verlag.