Collaborative Modeling – A Design Science Approach

Peter Rittgen
Vlerick Leuven Gent Management School, Belgium
and University College of Borås, Sweden
peter.rittgen@vlerick.be

Abstract

We have used a design science approach to study the collaborative creation of conceptual models. We have designed a collaborative modeling architecture based on business needs and applicable knowledge from theory and empirical findings from a modeling study using conventional modeling. A tool for this architecture was then developed and used as an instrument to confirm the practical relevance of the approach and the validity of the employed theory.

1. Introduction

Collaborative modeling refers to a process where a number of people actively contribute to the creation of a model. The weakest form of involvement is feedback to the session facilitator, similar to the conventional way of modeling (brown paper sessions). Stronger forms are proposals for changes or (partial) model proposals. In the following we focus on collaborative process modeling in small groups that is carried out in a synchronous way, i.e. all participants worked on the model at the same time and at the same place. The tool allows for asynchronous and distributed collaboration (the latter when combined with a video conference system) but these features were not investigated.

We have used a design science approach [1, 2] to study the collaborative creation of conceptual models. We designed a collaborative modeling architecture that is based on business needs and applicable knowledge from theory and empirical findings from a modeling study using conventional means. Then a tool for this architecture was developed and used as an instrument to confirm the practical relevance of the approach and the validity of the architecture. The tool itself was not evaluated. It only served to facilitate the evaluation of the underlying architecture.

The broader view on design science embeds the development of the artifact into circles of empirical and theoretical validation and improvement. This ensures both rigor and relevance of the results and puts design science on an equal footing with the more accepted behavioral paradigm. Figure 1 shows the principle layout of design science research.
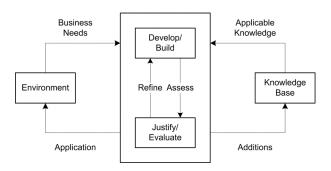

Figure 1. The design science circles (adapted from [1])

At the core is the design circle where an artifact is built. Artifacts can be constructs, models, methods or instantiations. The artifact is evaluated in an empirical study and the results will lead to a refinement (i.e., further development) of the artifact, and so on. The design is driven by business needs, and the subsequent application of the artifact in an appropriate environment (e.g., a real-life organization) provides empirical validation of the relevance of the artifact together with further or changed needs that are considered in the next round (relevance circle). The final circle (rigor circle) embeds artifact design into a knowledge creation process. Existing theoretical knowledge is applied in the design of the artifact and its evaluation updates the base or adds knowledge to it.

In the following we describe the use of this approach in collaborative modeling. The next section reports on related research; section 3 describes how the business needs were derived (upper half of relevance circle). Section 4 elaborates on applicable knowledge (upper half of the rigor circle), which comes from two sources: Theoretical knowledge from the literature about suggested means for addressing the business needs, and empirical knowledge from a modeling experiment. Section 5 describes how this has been used

in the design circle to develop an architecture for collaborative modeling (which is a model artifact according to design science terminology), and a tool implementing this architecture (i.e., an instantiation). The architecture consists of acceptance rules, negotiation patterns and the core modeling activities.

Instantiations in design science are concrete exemplars, e.g. instances of models or prototypes. In our case it is a tool that embodies (or implements) the architecture. The tool, as an instantiation artifact, allows us to put the abstract artifact of the architecture to a concrete use by subjecting it to empirical evaluation. This evaluation will then enable us to draw conclusions on the architecture as well. The tool is in this sense also a vehicle for validating the underlying architecture, which accounts for the lower half of the right circle (additions to the knowledge base).

In section 6 we report on the results from a survey where the architecture has been applied. These results assert its practical usefulness (lower half of the relevance circle) in terms of satisfaction of the business needs, and they confirm the validity of the suggestions derived from theory.

2. Related Research

Collaborative modeling processes have been studied by [3-7]. In [3] modeling involves domain experts, modeling mediators and model builders. It is viewed as a form of information gathering dialogue where knowledge is elicited from the domain experts. This view can be challenged because modeling is a social and communicative process where much of the information is created by and through the process rather than gathered from domain experts. We have therefore studied a situation where the participants had no a priori roles but all started from a similar position of having very little domain knowledge and collectively tried to make sense of the case described in a text document. [3] goes on developing meta model-based strategies which are of a prescriptive nature. Contrary to that and as outlined above our aim is to investigate what is actually done during modeling, i.e., we take a descriptive approach.

[4] emphasizes the importance of natural language as the primary medium and identifies two principal activities and associated roles: the domain expert who concretizes an informal model and a system analyst who abstracts a formal model. A detailed process model of both activities is given that again aims at prescribing steps to be performed to achieve a "good" analysis model. [5] distinguishes between an elicitation and a formalization dialogue and develops a modeling procedure by generalizing existing procedures for specific modeling languages. The authors claim that these procedures are descriptions of the modeling

process (i.e., "documented procedures") but the focus is again clearly on prescription (see, e.g., their use of the term "guidebook"). In this sense all other approaches can also be seen as descriptive but we use this term in a more direct sense, i.e. meaning the direct observation of modeling behavior with the purpose of getting a richer and more detailed description of the modeling process.

[6] acknowledges that modeling is not only a knowledge elicitation process but also a knowledge creation and dissemination process. It is viewed as a structured conversation where sub-conversations are associated with goals and strategies (the latter are elaborated in [3]). We fully agree that modeling is a conversation but we claim that it is a specific type of conversation, namely a negotiation. This idea is implicitly present in [6] where the dialogue structure contains negotiation elements such as propose and accept. We elaborate this point in the following sections and deliver a more detailed negotiation model. [6] also advocates the use of controlled language and validation. We consider the latter as problematic as it has often been observed that domain experts falsely agree with a model not being fully aware of all its implications.

[7] studies the influence of situational factors on modeling (in particular, enterprise modeling). The author's aim is to create an environment that facilitates and supports participative modeling.

3. Business Needs

To discover the business needs in collaborative modeling we conducted 15 in-depth interviews with IT consultants from four different companies. All of them were performing business process analyses on a regular basis. We asked them about the most urgent problems (according to their own views) when doing modeling in groups. To ensure proper understanding we asked them to elaborate on each point and from that we summarized the respective issue in a term describing the problem in short. We also asked them during the interview to verify that the term correctly reflects their problem situation. If we came across the same situation twice we tried to use the established term to avoid redundancy. We only included in our study those problems that were mentioned by at least two of the respondents to ensure their relevance. The result is the following unordered list:

- Low model acceptance
- Conformist behavior
- Eliciting individual views
- Verifying a model
- Participants feel misunderstood

- Slow progress
- Making conflicting views converge
- Facilitator overload
- Low technical model quality
- Low perceived model quality
- Limited model comprehension

Low model acceptance means that the resulting group model is not supported by some of the stakeholders in the modeling group. Conformist behavior refers to an individual who represents the group view instead of his own because of group pressure. Apart from leading to lower model acceptance for the concerned it also implies that the conformist's own view is not represented in the model which thwarts perceived model quality. Eliciting individual views is the problem of "extracting" the relevant knowledge from the respective participant and making sure that it is correctly reflected in the model. This is difficult as participants often do not possess sufficient modeling literacy to see whether their views were properly translated to the modeling language used. Verifying a model means that the facilitator must make sure that the resulting model does not only reflect a particular view alone but fairly represents all views. The problem is that the views can overlap and consensus in these areas is hard to establish.

That participants feel misunderstood is often the consequence of bad elicitation or a wrong understanding of the model. As communication about dissatisfaction is done in a natural language, i.e. in a different one from that of the model, the source of the problem can be hard to detect. Slow progress can have a number of sources such as communication problems or inefficiencies (the facilitator bottleneck). In order to reconciliate conflicting views the underlying conflict has to be exposed and solved. It requires a thorough discussion on the pros and cons of each view. Facilitator overload refers to a situation where the facilitator can no longer handle the coordination efforts in real time, which often leads to slow progress (see above). By low technical quality of the model we mean that the model exhibits objective mistakes. They comprise all syntactic flaws, where the rules of the language are violated, and some semantic flaws (e.g. deadlocks in a process model).

Low perceived quality refers to the individuals' subjective assessment of the model quality. From a semiotic perspective this is located on the semantic and pragmatic levels. It comprises aspects of model comprehension and model agreement. Limited model comprehension means that group members do not or not fully comprehend the model. This is an aspect of perceived model quality (see previous issue).

4. Applicable Knowledge

In the second step we identified potential solutions to the problems found in the needs analysis. We searched the relevant literature on collaborative modeling for means that have been suggested and used in different modeling cases to overcome the mentioned or similar problems. We then put the list of means in front of the same consultants and asked them to pick first the means they deemed most suitable for solving a problem and then possibly further means in order of their suitability. Before that the means had been explained in detail. To reduce cognitive load the respondents had a list over the means with short descriptions. The results are documented in Table 1 together with the references to the relevant literature sources.

Table 1. Means for solving modeling problems

Problem	Means
Low model acceptance	Involve participants directly in model building [8]
Conformism	Anonymity of participants [9]
Eliciting individual views	Gathering relevant concepts (brainstorming) [10], Translate participants' perceptions into the modeling language [8], Involve participants directly in model building
Verify model	Facilitator prepares model, participants criticize/change (prototyping) [11]
Participants feel misunderstood	Involve participants directly in model building, Translate participants' perceptions into the modeling language
Slow progress, facilitator overload	Involve participants directly in model building, Start with a preliminary model [12]
Making different views converge	Negotiation [13]
Low technical model quality	Syntax/semantics check by modeling tool [14]
Low perceived model quality	Iterative modeling (versioning) [15], Involve participants directly in model building
Limited model comprehension	Participant training + expert support [8]

Low model acceptance can be remedied by involving participants directly in model building, e.g. by allowing them to make immediate changes to the respective diagram or by having them draw diagrams themselves wherever possible. Conformism can be overcome by preserving the anonymity of the participants. To elicit individual views, the relevant concepts have to be collected in a divergent brainstorming procedure. In the case of structured views this involves a translation of the view into the modeling language which is typically done by a modeling expert. Where applicable, participants can also be involved directly. For the verification of models it has been suggested that the facilitator prepares (a first version of) a model which is criticized by the group members and changed accordingly, which then leads to a new version and so on. This is called prototyping.

That participants feel misunderstood can be helped by directly involving them or by translating their views in a correct way. Slow progress and facilitator overload are consequences of the chauffeured modeling style which requires all communication to go via the facilitator. This can be changed by giving participants a more active role and/or by starting with a preliminary model prepared by a supporting group. To make different views converge negotiation has to be applied. Technical model quality can be solved with the help of a modeling tool that checks syntax and semantics. But the perceived quality requires more sophisticated means. Iterative modeling is suggested here which means that modeling proceeds in small steps (versions). Again, participant involvement plays a key role, too. Better understanding of the models can be reached by training the participants in the modeling language (at least partially) and by having them supported by a modeling expert who can explain the models and also translate their perceptions to the modeling language.

5. Artifact Development

We conducted 3 experiments that involved a total of 26 groups of 2-3 students in informatics over a period of 3 years. The students were provided with a description of four business processes in a hospital. They were asked to model these processes with the help of two different modeling languages that they could choose freely from a set of four languages: ARIS-EPC [16], FMC-Petri nets [17], UML [18, 19], and DEMO [20]. We assumed that two factors are predominant in model creation:

- The internal mental processes of each modeler, and
- The conversations between modelers and within the group.

To get access to the former we used a think-aloud process-tracing methodology [21, 22] where members of the group speak out what they currently think. The utterances were then transcribed yielding think-aloud protocols. The same is done with the conversations. In addition to that we also considered the product of the modeling process, the models themselves, to fill the gaps in the protocols and to help with interpreting ambiguous phrases in them. Open issues that could not be dealt with in this way were marked on the coding scheme and clarified by ex-post interviews with the respective groups.

As a result of this study we derived acceptance rules (social level), a negotiation pattern (pragmatic level), and a number of core modeling activities.

5.1. Acceptance Rules

The social norms within a modeling team are mainly made up of rules for determining whether a proposal is accepted or rejected. We observed that these rules do not have to be logical complements which allows for situations where a proposal can be neither rejected nor accepted but requires further convincing to decide one way or the other. A termination rule was applied occasionally to force a decision if a negotiation got stuck, i.e., when there were no more changes in the individuals' convictions over an extended period of time. We witnessed two types of rules:

- Rules of majority, where a certain number of group members had to support or oppose a proposal in order for the whole group to accept or reject it (e.g., more than half). A tie-break rule was sometimes specified (e.g., for the case of an equal number of supporters and opponents). The tie-break could involve seniority issues.
- Rules of seniority, where the weight of a group member's support or opposition was related to his or her status within the group. This status could be acquired (e.g., by experience) or associated with a position to which the member was appointed. A frequent example of this was the case of a more experienced modeler who was considered as the leader by the group and took decisions on their behalf. The other members filled the role of consultants in such a case.

These rules were sometimes set up explicitly before the group began their work, or in an early phase of this work. But in most cases they rather emerged as the result of each member's behavior. Individuals making regular contributions of high quality were likely to acquire seniority. In homogeneous teams majority rules were used more often.

5.2. Negotiation Pattern

The majority of the activities on the pragmatic level were associated with negotiation. This is surprising as modeling is typically rather pictured as an intuitive act that is largely the product of a creative brain (e.g., a consultant) that possibly receives some input from other stakeholders in the modeling process (e.g., domain experts from the respective departments). The results rather suggest that modeling is a well-structured process. It consists of a limited number of well-defined activities on all levels of the semiotic ladder. We are aware that further research will reveal more activities but from the experience of the three experiments that yielded a decreasing number of new ones, we are confident that the total number of activities will converge. The activities identified so far can therefore be assumed to be relatively stable. To a certain extent this is even true across different modeling languages, although the terminology of concepts may vary and not every concept is realized in each of the languages.

An analysis of the workflows on the pragmatic level revealed a structure that goes beyond the mere identification of generic activities. We found out that the negotiation process actually follows a certain pattern. This pattern is shown in Figure 2.
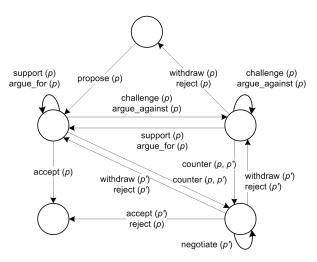

Figure 2. Negotiation pattern

It consists of an initial and reject state at the top, a state where acceptance is favored (upper left-hand corner), a state where rejection is favored (upper right-hand corner), a recursive sub-state for negotiating a counter-proposal (lower right-hand corner) and an accept state (lower left-hand corner). Each of the states allows for a set of certain pragmatic activities that take the negotiation to a different state. We have left out the parameters concerning the modeler who performs the activity and the argument (if present). In general any

modeler can perform any activity but there are a few rules to be observed. A modeler making a proposal is implicitly assumed to support it. He is the only one who may withdraw it. A counter-argument is brought up by a different modeler but a counter-proposal can also be made by the proponent of the original proposal, e.g., to accommodate counter-arguments. With the help of the pattern of Figure 2 we can control the negotiation component of a modeling support system. On the other levels we were not able to discover an equally strong pattern of activities. This will affect the kind of support an artifact can provide at the language level.

5.3. Core Modeling Activities

The analysis of the modeling protocols also revealed recurring activities that the group members engaged in to solve their modeling assignments. In general terms they can be summarized as follows:

- Create/change an individual model
- Discuss a proposal
- Propose a new/changed individual model to the group
- Comment on a proposal
- Vote on/assess a proposal
- Discuss an unclear issue
- Decide on a group model (new version)
- Merge proposals
- Discuss use of modeling language
- Reuse parts of a proposal/version

Functions corresponding to these activities have been implemented in a tool that supports collaborative modeling. It is called the COMA tool and it can be downloaded freely at http://www.coma.nu. COMA is an acronym for COllaborative Modeling Architecture. Details regarding the foundations of COMA and the used modeling language can be found in [23]. The tool itself is described in [24].

Activities involving discussions are not supported. For them we rely on face-to-face conversations, i.e. we assume that the group members are located in the same room. If this is not the case, the group can avail themselves of standard Voice-over-IP teleconferencing for this purpose. Figure 3 shows a screenshot of the COMA tool that was taken during one of the modeling sessions.
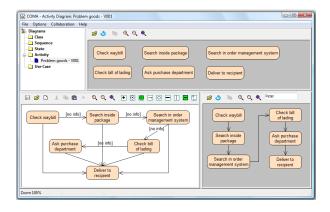
Figure 3. COMA screenshot

It shows a snapshot of the modeling process at a certain stage. This is supposed to give the reader an example of how modeling in COMA proceeds and how it helps with a particular problem, namely that of "Making different views converge". The group in question was concerned with the handling of so-called problem goods, i.e. goods with an unclear recipient. In a first step they simply wrote down all the activities that are involved thus arriving at the first version (Eliciting individual views, upper pane). One member suggested to order the activities in a certain sequence and made a respective proposal (lower right pane). He knew from experience that this was indeed the order in which the activities were carried out at Goods receipt. Another modeler agrees with the principle sequence but he is quite sure that the search for the recipient is terminated as soon as the recipient is identified and further steps are skipped. He draws the respective diagram in his editor window (lower left pane) and makes a counter-proposal. On seeing the apparent conflict the first modeler confirms with the operations staff that this is indeed the case and withdraws his original proposal in favor of the new one. The new proposal received supporting votes by the other team members and was subsequently adopted by the group as version two.

6. Application of the Artifact and Additions to the Knowledge Base

To apply the architecture in a business context we have devised a schema of how each of the means can be supported by the tool. This is shown in Table 2. Please observe that the procedures in the right column are only suggestions for the realization of the means. We did not know in advance whether these procedures could be performed in practice or whether they would have any positive effect regarding the solution of the problems that were identified in section 2.

Table 2. Means and their COMA Support

Means	Implementing the means with the help of COMA
Anonymity of participants	Use bogus user names instead of real ones
Facilitator prepares model, participants criticize/change (prototyping)	Facilitator creates local model & proposes it to the group, Participant comments on proposal, Participant copies & pastes from proposal, changes it & makes new proposal
Gathering relevant concepts (brainstorming)	Participant collects relevant concepts in a local model & proposes it to the group, Facilitator merges all proposals
Involve participants directly in model building	Participant creates local model & proposes it, Participant copies & pastes from proposal, changes it & makes new proposal
Iterative modeling (versioning)	Participants vote on proposals, Facilitator decides on a group model (new version) as starting point for new round
Participant training + expert support	<No tool support>
Starting with a preliminary model	<Not considered here>
Syntax/semantics check by modeling tool	Modeling tool prevents drawing wrong diagrams
Translate participants' perceptions into the modeling language	Modeling expert helps participant in creating the local model with the tool

In order to find out we set up an empirical study where participants were trained in the use of the tool according to the procedures stipulated in Table 2. We asked the 15 consultants that already participated in the needs analysis to apply the tool in one of their real-life modeling sessions. Four of them were not willing to do so because they were concerned that it would affect their projects too much. Of the others five had similar concerns but found the tool sufficiently convincing to give it a try. The others raised no objections. This makes a total of 11 respondents.

All of the participants, i.e. the consultants and their customers, received training with the tool in advance of the study. The training consisted in a presentation of ideas and concepts behind COMA (ca. 1 h), a

demonstration of the tool itself (ca. 1.5 h) and a short trial session where the group modeled a fictitious example (ca. 2 h). In the latter we went through all the steps in Table 2.

After the consultants have used the tool in real sessions, we have interviewed them. In this structured interview we have asked the following two questions for each of the problems shown in Table 1:

- Do you think that the problem can be solved satisfactorily in the traditional way you use to model?
- Do you think that the problem can be solved satisfactorily with the help of the tool?

Respondents were asked to rate their answers on a 7-point Likert scale ranging from Strongly disagree (-3) to Strongly agree (+3). Variables were named after the abbreviated problem (e.g., ModAcc = Low model acceptance) for the conventional case without COMA support. In the tool-supported case we added the suffix "Tool". We computed the mean of each variable and the differences of the variable pairs NoTool-Tool with the help of the SPSS statistical software package. We then performed a T test (paired variables test) to determine whether the NoTool and Tool samples really come from different populations, i.e. if the use of the COMA architecture did indeed have an impact on the solution of the respective problem.

We used a confidence level of 95 %, which corresponds to a significance level of 5 %, a common value in statistical testing. The null hypothesis for a T test assumes that the two samples are from the same distribution, i.e. the use of COMA does not lead to any changes. In this case the zero is contained in the confidence interval. The T test is "successful", i.e. the null hypothesis is rejected, if the confidence interval for the mean of the difference distribution does not contain the zero. Please observe that the non-rejection of the null hypothesis does not mean that it is true but only that the evidence is not strong enough to discount it. A failed T test can therefore mean that the impact of the architecture exists but is too weak to be detected with the relatively small sample size we had in our study.

Table 3 shows the mean of the variables, the confidence interval for the differences of the pairs and the significance level of the result. The significant results are marked in bold.

Table 3. Impact of COMA on the solution of typical problems in collaborative modeling

Pair	Variable	Mean	Difference	95% Conf. Int. of the Difference		Sig. (2-tailed)
				Upper	Lower	
1	ModAcc	0.64	ModAcc - ModAccTool	-1.603	-0.034	**0.042**
	ModAccTool	1.45				
2	Conf	-0.91	Conf - ConfTool	-1.501	0.228	0.132
	ConfTool	-0.27				
3	Elicit	1.82	Elicit - ElicitTool	-1.013	1.377	0.742
	ElicitTool	1.64				
4	Verify	1.27	Verify - VerifyTool	-1.833	0.015	0.053
	VerifyTool	2.18				
5	PartMis	1.09	PartMis - PartMisTool	-1.551	0.279	0.152
	PartMisTool	1.73				
6	Slow	-0.45	Slow - SlowTool	-4.148	-0.579	**0.014**
	SlowTool	1.91				
7	Converge	-0.64	Converge - ConvergeTool	-1.996	-0.004	**0.049**
	ConvergeTool	0.36				
8	TechQual	2.09	TechQual - TechQualTool	-1.372	0.645	0.441
	TechQualTool	2.45				
9	PercQual	0.82	PercQual - PercQualTool	-2.416	-0.311	**0.016**
	PercQualTool	2.18				
10	ModComp	1.36	ModComp - ModCompTool	-1.672	-0.146	**0.024**
	ModCompTool	2.27				

The statistics for pair 1 show that COMA has a significant impact on model acceptance. The probability that this outcome is coincidental is only 4.2 % and the confidence interval does not contain the zero. The null hypothesis must therefore be rejected in favor of the hypothesis that the use of COMA did indeed improve the acceptance of the model by the participants (mean increased from 0.64 to 1.45).

The situation regarding pair 2 is unclear. The mean has increased here, too, but the change is not significant. A larger sample might still provide the necessary evidence to discount the null hypothesis but it seems more likely to assume that the use of bogus names alone is not enough to yield anonymity in collocated sessions as face-to-face communication might reveal the identity of participants. We therefore suggest that group members are placed in different rooms and direct communication is replaced by a text-based chat if conformism is an issue.

Pair 3 reveals that elicitation of concepts is not improved by using COMA. The outcome is highly insignificant and the mean even indicates a slight decrease, which can be coincidental, though. On the whole this result is not unexpected as COMA is geared towards the convergence of existing views rather than the generation of new ones.

Pair 4 shows a large but insignificant improvement of the verification of a model. Contrary to pair 2 the difference is sufficiently large to justify the expectation that a larger sample will provide the lacking evidence for improvement. This assumption is supported by the fact that the significance level only slightly fails the 5 % margin.

Our study has likewise not found sufficient support for the hypothesis that COMA helps solving the problem of participants that feel misunderstood (pair 5). Interviews with consultants that did not find the tool useful in this respect indicated that group members were not sufficiently "fluent" in its use to reap the full benefits. Repeated use of the tool might change this, which calls for a study that involves a longer term.

Regarding modeling speed the result is unequivocal (pair 6). The improvement is both large and highly significant (1.4 %). This is not surprising as the use of tools in a collaborative setting, even if it is restricted to textual input only, has been shown to be beneficial in earlier studies, e.g. [13]. Extending this support to graphical intervention can be expected to have an even larger impact as the frequent changes to the diagrams made by the facilitator only are time-consuming.

Pair 7 shows that COMA enables the convergence of different views. Although the architecture is specifically built for this purpose it is still a surprising result as convergence is considered to be one of the toughest issues in modeling [25]. The study shows that the relatively simple negotiation mechanics of COMA are sufficient to produce both a significant and large increase in view convergence by a full point on a 7-point scale but it also shows that there is still ample room for further improvement as the absolute value lies only in the mid-range of the scale.

The technical model quality improved somewhat (see pair 8) but the change is not significant. The slight improvement is probably due to the fact that the consultants used a drawing tool without syntax checker before. But in spite of that and most likely due to their considerable experience in modeling they made few mistakes even in the absence of a syntax check so that COMA had little impact in that area.

The COMA impact on perceived quality is similar to that on modeling speed, i.e. it is both significant and large (an increase by 1.36 points). In this category we witnessed the clearest improvement (pair 9). We tried to shed some more light on the reasons for this impact and interviewed some of the more active members of the actual modeling groups. The participants confirmed that they experienced a higher model quality in the tool-supported session but gave different reasons for that: a better understanding of the model, seeing their own view better represented and being able to express a controversial view. From this we concluded that perceived quality is a multi-factor variable and we hypothesized that COMA has a positive influence on some of the factors combining to a strong effect on the depending variable. We plan to conduct a study to investigate the dimensions of perceived quality more closely.

Model comprehension (pair 10) also increased significantly but to a lesser extent. Even here we were interested in the reasons behind this effect and we asked both the consultants and some of the team members a corresponding question. The answers revealed that the major reason was the accessibility to the model already during the session. Note that in conventional modeling a new version is only drafted after a session is completed.

7. Conclusions

We followed a design-science approach to study collaborative modeling. In the design circle we developed two artifacts, an architecture (COMA) and

a tool that implements this architecture. The design was driven by theoretical insights and observations of group modeling behavior. In the relevance circle we identified business needs and put the artifact to a practical test to assess the degree to which it fulfils these needs. In the rigor circle we confirmed the knowledge about the existing means to solve typical problems in collaborative modeling.

We found significant positive impact of COMA on five of ten problems in modeling, namely:

- Low model acceptance,
- Slow progress / facilitator overload,
- Making different views converge,
- Low perceived model quality, and
- Limited model comprehension.

We also identified a need for further research in two main areas. Firstly, in the cases where COMA proved useful, we have to explore the exact nature of the relation between causal factors and the respective problem. This would allow us to better understand the roots of the problem and ultimately develop adequate solutions, e.g. an improved architecture and better tool support.

Secondly, we need to investigate the cases where COMA did so far not prove helpful in solving the problem. Here we must further distinguish between two types of cases. On the one side we have problems where a suitable architecture can help in principle but COMA is not yet sufficiently mature. On the other side we have the problems that require a completely different "instrument" for solving them such as new facilitation techniques, novel team organizations or the like.

We believe that COMA is also applicable to other areas of conceptual modeling. The negotiation pattern itself is even likely to be useful in other collaborative scenarios. In our opinion collaboration is a close form of working together which requires that the meaning of terms and the desired output have to be negotiated to make sure that the concerns of all collaborators are met. These hypotheses have not been validated yet, though. Another open issue is the optimal group size for collaborative modeling. Our experiences so far are that it is probably smaller than in conventional modeling due to the more active involvement of the participants.

All options have in common that they involve further iterations of the design science circles to make sure that the results are both scientifically valid and practically applicable.

8. References

[1] A.R. Hevner, S.T. March, J. Park, and S. Ram: "Design Science in Information Systems Research," *MIS Quarterly* 28, 2004, pp. 75-105

[2] S.T. March, and G. Smith: "Design and Natural Science Research on Information Technology," *Decision Support Systems* 15, 1995, pp. 251-266

[3] P.v. Bommel, S.J.B.A. Hoppenbrouwers, H.A.E. Proper, and T.P.v.d. Weide: "Exploring Modeling Strategies in a Meta-modeling Context," in Meersman, R., Tari, Z., Herrero, P. (eds.): *On the Move to Meaningful Internet Systems 2006*: OTM 2006 Workshops - OTM Confederated International Workshops and Posters, AWESOMe, CAMS, COMINF, IS, KSinBIT, MIOS-CIAO, MONET, OnToContent, ORM, PerSys, OTM Academy Doctoral Consortium, RDDS, SWWS, and SebGIS, Proceedings, Part II, Montpellier, France, Vol. 4278. Springer, Berlin, Germany, 2006, pp. 1128-1137

[4] P.J.M. Frederiks, and T.P.v.d. Weide: "Information modeling: the process and the required competencies of its participants," *Data & Knowledge Engineering* 58, 2006, pp. 4-20

[5] S.J.B.A. Hoppenbrouwers, L. Lindeman, and H.A. Proper: "Capturing Modeling Processes - Towards the MoDial Modeling Laboratory," in Meersman, R., Tari, Z., Herrero, P. (eds.): *On the Move to Meaningful Internet Systems 2006*: OTM 2006 Workshops - OTM Confederated International Workshops and Posters, AWESOMe, CAMS, COMINF, IS, KSinBIT, MIOS-CIAO, MONET, OnToContent, ORM, PerSys, OTM Academy Doctoral Consortium, RDDS, SWWS, and SebGIS, Proceedings, Part II, Montpellier, France, Vol. 4278. Springer, Berlin, Germany, 2006, pp. 1242-1252

[6] S.J.B.A. Hoppenbrouwers, H.A. Proper, and T.P.v.d. Weide: "Formal Modelling as a Grounded Conversation," in Goldkuhl, G., Lind, M., Haraldson, S. (eds.): *Proceedings of the 10th International Working Conference on the Language Action Perspective on Communication Modelling* (LAP`05), Kiruna, Sweden, Linköpings Universitet and Högskolan i Borås, Linköping and Borås, 2005, pp. 139-155

[7] Persson, A.: *Enterprise Modelling in Practice: Situational Factors and their Influence on Adopting a Participative Approach*. Department of Computer and Systems Sciences, Stockholm University, Stockholm, 2001

[8] D.L. Dean, R.E. Orwig, and D.R. Vogel: "Facilitation Methods for Collaborative Modeling Tools," *Group Decision and Negotiation* 9, 2000, pp. 109-127

[9] E.A.J.A. Rouwette, J.A.M. Vennix, and C.M. Thijssen: "Group Model Building: A Decision Room Approach," *Simulation & Gaming* 31, 2000, pp. 359-379

[10] D. Shaw, F. Ackermann, and C. Eden: "Approaches to Sharing Knowledge in Group Problem Structuring," *Journal of the Operational Research Society* 54, 2003, pp. 936-948

[11] M.d. Hengst, and G.J.d. Vreede: "Collaborate Business Process Engineering: A Decade of Lessons from the Field," *Journal of Management Information Systems* 20, 2004, pp. 85-113

[12] Vennix, J.A.M.: *Group Model Building: Facilitating Team Learning Using System Dynamics*, Wiley, New York, 1996

[13] D. Dean, R. Orwig, J. Lee, and D. Vogel: "Modeling with a group modeling tool: group support, model quality, and validation," *Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences*, Vol. IV, Information Systems: Collaboration Technology Organizational Systems and Technology, 4-7 Jan 1994, Vol. 4, IEEE Computer Society Press, Los Alamitos, CA, 1994, pp. 214-223

[14] D.L. Dean, J.D. Lee, R.E. Orwig, and D.R. Vogel: "Technological Support for Group Process Modeling," *Journal of Management Information Systems* 11, 1994, pp. 43-63

[15] J. Lee, C. Albrecht, and J.F. Nunamaker: "Experiences with Collaborative Applications that Support Distributed Modeling," *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, 3-6 Jan. 2001, IEEE Computer Society, Washington, DC, USA, 2001

[16] Scheer, A.-W.: *ARIS - Business Process Modeling*, Springer, Berlin, 1999

[17] F. Keller, and S. Wendt,: "FMC: An Approach towards Architecture-Centric System Development," *Proc. 10th IEEE Symposium and Workshop on Engineering of Computer Based Systems*, IEEE Computer Society, Pasadena, CA, 2003, pp. 173-182

[18] OMG: *UML 2.0 Superstructure Specification*, OMG, Needham, MA, 2004

[19] OMG: *Unified Modeling Language: Infrastructure*, OMG, Needham, MA, 2006

[20] J.L.G. Dietz: "Understanding and modeling business processes with DEMO," in Akoka, J., Bouzeghoub, M., Comyn-Wattiau, I., Métais, E. (eds.): *Proceedings of the 18th International Conference on Conceptual Modeling*, ER '99, Springer, Berlin, 1999, pp. 188-202

[21] A. Srinivasan, and D. Te´eni: "Modeling as Constrained Problem Solving: An Empirical Study of the Data Modeling Process," *Management Science* 41, 1995, pp. 419-434

[22] Ericsson, K., Simon, H.: *Protocol Analysis: Verbal Reports as Data*, MIT Press, Boston, 1993

[23] P. Rittgen, "Negotiating Models," in Krogstie, John; Opdahl, Andreas; Sindre, Guttorm (eds.): *Advanced Information Systems Engineering*, 19th International Conference, CAiSE 2007, Trondheim, Norway, June 2007, Proceedings, LNCS 4495, Springer, Berlin, 2007, pp. 561-573

[24] P. Rittgen, "COMA Handbook," 2007, http://www.coma.nu/COMA_Handbook.pdf

[25] H.H. Clark, and S.E. Brennan: "Grounding in communication," in Resnick, L.B., Levine, J., Teasley, S.D. (eds.): *Socially Shared Cognition*, American Psychological Association, Washington, DC, 1991, pp. 127-149