

Attacking the Knowledge Acquisition Bottleneck through Games-For-Modelling

S.J.B.A. (Stijn) Hoppenbrouwers¹ and P.J.F. (Peter) Lucas¹

Abstract. Many model-based methods in AI require some sort of formal representation of knowledge as input. Acquisition of such formal models is either done manually, using a knowledge elicitation and modelling method, or automatically, applying knowledge discovery and machine learning techniques to available data. For the acquisition of highly structured, domain-specific knowledge, machine learning techniques still fall short, and knowledge elicitation and modelling is then the standard. However, obtaining formal models from informants who have few or no formal skills is a non-trivial aspect of knowledge acquisition, which can be viewed as an instance of the well-known “knowledge acquisition bottleneck”. In addition, if there are social requirements on knowledge representations, e.g. constructive agreement on concepts and propositions, this poses a further challenge. Based on our work in conceptual modelling and method engineering, we propose to cast methods for knowledge modelling in the framework of games. The resulting games-for-modelling approach is illustrated by a number of examples from ongoing projects. Our chief long-term aim is to decrease the threshold for formal knowledge acquisition and modelling.

1 INTRODUCTION

In this paper we propose and illustrate an approach to knowledge acquisition and formalisation that does not primarily address the formal structures to be delivered, but rather the *process* of conceptualization and modelling, yielding formal models.

Formal knowledge models of some sort are essential in AI and related fields, not just as part of the theoretical foundations of the fields, but also for application: computation based on knowledge structures inherently demands some artefacts, which may vary from “lightweight formalisations” (e.g. diagrams or strictly structured text) to expressions with formal semantics. Focus in most cases is on the syntax and semantics of the formal artefacts, and on associated reasoning methods to apply them to problems. Obtaining such formal models is nowadays tackled in AI by using some knowledge acquisition (KA) and modelling method, such as CommonKADS, which suggests a step-wise approach, starting with informal, conceptual representations and methods and refining these until a formal model is obtained [1]. Although such methods were initially proposed as solutions to the *Knowledge-Acquisition Bottleneck* (KAB), experience shows that the KAB is as real as it was more than a decade ago when CommonKADS was proposed [2]. The enormous increase in the volume of knowledge discovery from data and machine learning research during the last decade, which was largely motivated by

the appeal of *automatic* knowledge acquisition from data [3], is evidence that the KAB is as prominent nowadays as when it was first mentioned in the 1980s.

Within our current focus, the most urgent KAB aspects are the following:

- Knowledge is hard (and expensive) to make explicit, and even harder to formalise;
- The domain experts required for this job are usually not available for lengthy involvement in KA activities, nor do they possess the required modelling and formalisation skills;
- The expert knowledge engineers/modellers that could be hired to do the modelling job are few and expensive. Breaking the KAB by structurally employing expert modellers will only work in the most urgent of cases, covered by unusually large allocation of resources.

Many of the promises of AI concern a global user community of organisations and citizens that will never have access to expensive knowledge engineering experts. This will simply prevent many of the promises with respect to the wide availability of knowledge-based AI solutions from being fulfilled and, furthermore, it casts doubts about the future of the semantic web.

The practical problem of the KAB presents us with challenges that are urgent and interesting enough to warrant focused academic efforts for understanding and alleviating the problem. As acknowledged by Wagner [4], in fields like software engineering, information system engineering, and enterprise engineering, we are confronted with KAB-like problems on a large scale, and consequently solutions are actively sought.

In our research we are developing alternatives to existing knowledge acquisition and modelling methods. One idea we are exploring is to look at formal knowledge modelling activities as *games*, forcing ourselves to look at contextualised, operational modelling in which human factors are inevitably included. Because the way in which we employ games for formal knowledge modelling involves human-computer interactions (HCI), these games-for-modelling systems can best be tested using HCI-like evaluation methods, including existing methods specifically aimed at game evaluation. This combination, games-for-modelling and exploitation of HCI methods for evaluation, is, to the best of our knowledge, new to AI.

After providing a brief overview of related work, we will argue in favour of this approach, explain how methods can be viewed and designed as games, and provide some examples of such games, though admittedly only preliminary results on tested games are available as of yet. The paper is rounded off with conclusions of what has been achieved so far, and we offer a sketch of envisioned future work.

¹ Institute for Computing and Information Sciences, Radboud University Nijmegen, P.O. Box 9010, 6500 GL, Nijmegen, the Netherlands. Email: {stijnh,peter1}@sc.ru.nl.

2 RELATED WORK

We need to be clear about two distinct categories in KA: *automated* and *manual*. The first category uses knowledge discovery from data and machine learning techniques to derive models [3], the second depends on the construction of models by hand (aided by tools), by individuals or teams. Although there are certainly many situations where knowledge discovery from data and machine learning can be very useful, the fact must be faced that learning technology will not resolve the KAB for cases in which highly domain specific knowledge (ultimately kept in individuals' minds) has to be made explicit and formalised.

Early work on KA by Newell and Simon mainly focused on elicitation of verbal data collected from domain experts in the act of solving problems, called *think-aloud protocols* [5]. Useful knowledge was subsequently extracted from the protocols, using a technique called *protocol analysis*. Although the intention of protocol analysis was to obtain representations that could be manipulated by a computer, little attention was given to the actual semantics of the representations. The innovation by Newell and Simon was mainly to introduce techniques to AI which were originally developed in the area of psychology.

As mentioned above, in knowledge engineering, perhaps the foremost comprehensive method is CommonKADS [1], though many more exist. The essential idea is to work from informal, yet conceptually rich, models towards more formal models (using for example predicate logic), using a selection from a given set of *problem solving methods*. Problem solving methods can be best seen as generic methods that are aimed at solving particular tasks, such as diagnosis. A problem solving method can be instantiated for a particular domain, and the result is then a system that is able to solve the task for a particular problem in the domain. Despite the large size and huge number of people years invested in CommonKADS projects, only a limited collection of problem-solving methods are being offered by the CommonKADS methodology. The researchers who were originally involved in the development of CommonKADS are no longer active in this area, and the methodology has never become the industry standard of knowledge acquisition and modelling. Knowledge modelling is nowadays also called ontology building [6].

Roughly similar approaches are also widely used in system development, e.g. RUP [7] (typically in combination with the UML [8]). They all make use of roughly defined, iterative phases in the modelling process, from exploration and informal sketching to formalization and implementation; also, they all, to a stronger or lesser degree, suggest or prescribe specific artefacts (descriptions, models) for particular phases and purposes, often involving strong structuring and/or specific modelling languages.

Useful as all this is (though the number and diversity of specific modelling languages is rampant), such deliverable-oriented textbook methods only provide very limited help for non-expert modellers in the actual execution of their modelling tasks; they still require considerable study and above all practice to be mastered. The availability of tools provides some help, but currently such tools are usually highly technical model editors that require in-depth technical knowledge, and do not actively assist in the act of conceptualisation and formalization of the models. In other words, they support *model-centric modelling* instead of *modeller-centric modelling*.

An additional problem with domain specific, manual KA is posed by the social context of domain specific knowledge mod-

elling, which in many cases calls for intensive negotiation and validation of models by heterogeneous teams of stakeholders. In line with this, there is increasing interest in approaches for *collaborative modelling* [9,10]. Related issues are on the agenda in context of the Web 2.0 effort, and also in ontology engineering [11].

In [4,12,13], a conversation-based approach to knowledge modelling is suggested; in this vein, actual systems for modelling support have been created and studied, e.g. Wiki-based approaches [4] and a negotiation-based approach [10], the latter of which is most closely related to our own work, and in fact is an exception in that it does focus on the (conversational) process of modelling, or rather on supporting it. Yet, it still positions model editing (UML) as a central activity, assuming basic, diagram-oriented modelling skills to be available in the participants.

In view of the considerable challenge posed by the KAB, and focusing on manual KA, we find a dissatisfying lack of interest in issues that prevent real-life, operational modelling from becoming successful. In addition to focusing on representational issues (still the mainstream topic in literature on modelling), we believe that the situated act of modelling itself warrants study (starting from initial, sketchy conceptualization and moving on to actual formalization), including any relevant human factors involved. We may, for example, look at like usability/playability, learnability, even enjoyability, but also, of course, effectiveness and efficiency.

3 WHY GAMES?

Let us briefly explain what we mean by the *game metaphor*. People often refer to activities, tasks, or challenges (even complex, elaborate ones) as games, e.g. "the game of politics", "the game we play in this firm", "that sort of practice is not our game". At times, this referential metaphor is extended into actual identification or introduction of game aspects: competition, scores, declaration of winners/losers, rules, and so on. It seems justified, even fruitful, to use the game metaphor as well as actual game design as instruments in the study and development of tools for modelling support: games for modelling. This is in line with a well-established tradition of "Serious Gaming", prominently including management games [14]. We will now elaborate on our proposal to apply the game metaphor to thinking about modeller-oriented support systems for knowledge modelling.

In [15], a number of arguments are developed in favour of approaching the creation of operational methods/tools for modelling as game design. We briefly list the main arguments below:

Make formal modelling available to non-modellers As discussed, if low-threshold, domain-specific use AI is to really take off, large scale and low-threshold formal modelling *will* be required. An obvious but non-trivial way to proceed is to create software applications that make creation of required models as painless and efficient as possible: bring lightweight formal modelling to the masses through the virtual world emerging on the internet, and by shaping such applications as games.

Improve motivation of modellers

In the wake of Von Ahn [16], who managed to harness the creative energies of great numbers of on-line game players to perform "human computing", we believe it would be very helpful from both a methodological and a productivity point of view to make modelling more attractive (challenging, enjoyable), and

thereby boost modelling in order to answer the needs and help bring AI to its full utilitarian potential. We believe games are a highly promising way of doing so.

Improve quality of modelling

More in line with common objectives in the field of knowledge modelling, a gaming setup may help improve the quality of the products of modelling, both *textual* (the models as such) and *contextual* (knowledge, understanding, agreement etc. across communities involved with models and modelling). Useful strategies for modelling can be built into the game design (e.g. shaped as sub-games, tasks, challenges) or be left to the participants (the players' strategies), as best fits the situation.

Tooling: virtual environments for collaborative modelling

The relation between digital tools/environments for modelling and digital games is obvious. Video games are highly advanced interactive systems. Completely virtual work environments may not be accepted on a large scale yet, but completely virtual multi-player games most certainly are. It is quite possible that the knowledge modelling tools and environments of the future feature serious game characteristics.

Apart from the above arguments that focus on the support of actual, operational modelling, there is one that concerns research and development methodology with respect to games for modelling:

Research and development approach: improving performance by improving game design

The game metaphor as well as the actual application of game design theoretical concepts will help *focus on the relevant research questions* concerning model oriented interaction systems and duly constrained modelling. Games can be tried and tested on various audiences, providing ample and well-structured data on interactions and results, and therefore offering an empirical hold on modelling processes that otherwise would be much harder to obtain in large volumes. This will enable modelling-oriented research using evaluative approaches from AI and HCI.

4 GAMES EMBODYING METHODS

For the link between methods (i.e. systems for modelling support) and games-for-modelling, we turn to Game Design Theory. Järvinen [15] provides clear concepts for analyzing and designing games that help greatly in performing game design (and therefore also aid method and tool engineering in a gaming context). Below we list generic game elements according to Järvinen, and add the equivalent thereof for the construction of methods.

1. **Components:** objects that the player is able to manipulate and possess in the course of the game. In methods this corresponds to any objects manipulated in the modelling process, typically brief fragments of natural language text (even individual terms) and elements of diagrams, including instantiations of modelling concepts. These are in fact the items now manipulated by means of editors; however, we expect that the explicit incorporation of more fine-grained *intermediary deliverables* in the process (related to taking smaller steps in conceptualization and formalization) will add to the number of different game components.

2. **Rule set:** rules produce each individual possibility and constraint that a game has to offer for its players, including *set goals* and *procedures*. In methods, such rules constrain the liberty of action of the modeller; one could say the rules *consti-*

tute the method, plus situational goals set for a particular modelling job. In section 6, we will briefly return to this in view of a study into rule setting for modelling sessions.

3. **Environment:** the stage for game play. For example: a board, a field, or a virtual environment in a digital game. In operational methods, this can range from a meeting room to a whiteboard to a digital editor; in a completely digital (virtual) setting, interactive and possibly collaborative tools will be involved. Editor-like environments may be used, but beyond these, series of assignments may also be executed in less technical settings resembling virtual game boards or even 3D worlds.

4. **Game mechanics:** describe possible means with which the player can interact with game elements as she is trying to influence game states in order to complete a goal. For example: throwing in basketball, hitting in tennis; in more verbal games (and more relevant to our sort of gaming), proposing, asking, rejecting, and so on. The link with interaction mechanisms in operational modelling is obvious, but do note that game mechanics are not at all part of traditional (textbook) methods. In a conversation-oriented approach, the mechanisms associated with "verbal games" apply quite directly.

5. **Theme:** game theme is the subject matter that is used in contextualizing the rule set and its game elements to other meanings than those which the game system as an information system requires. For example: real-estate market in Monopoly, or a fictional context, or a historical event. For methods, setting themes is quite unusual so far (except for the actual, real modelling context as such). An inspiring yet rather radical idea for a theme would be, for example, *performing magic*, since this metaphorically corresponds nicely to applied knowledge modelling: "in order to get something (some service, information, prediction, and so on) you have to describe something precisely, conforming to procedures that the magic practice demands (ritual) and using the appropriate magical language."

6. **Information:** what the system and players need to know; the game state communicated. For example, a scoreboard, or a screen display, and/or component attributes such as value or number. In operational models, this can be the state of the model and procedural knowledge, but also feedback to the modellers (players) on the model (model checking, AI-based analyses) and the modelling process (status, progress, results, efficiency, etc.).

7. **Interface:** the tools to access game elements via game mechanics when direct (i.e. physical) access to game objects is impossible. For example, game pads, dance mats, mouse, steering wheels, etc. Though in operational modelling, rather standard games/systems interfacing is obvious, more innovative forms of interfacing may be worthwhile considering (e.g. 3D physical interfacing ("data gloves") or "surface computing").

8. **Player(s):** the human factor in the game; their behaviour, mood, abilities and skills, relationship with games, game tastes. In modelling, this of course applies to modellers or other participants, and their competencies, interests, expertise, and preferences. Interestingly, player characteristics may be linked to specific roles, expertise, concerns, and preferences of participants (stakeholders) in the modelling process.

9. **Context:** the physical location of the game, the time, players' personal histories, and other informal, external aspects to the game system that possibly affect the experience of playing the game. In modelling, this refers to the situational aspects of a particular modelling task and session.

At least the following elements are minimally required to design a game (constituting a working *definition*): a) components complemented with rules governing their behaviour, b) an information structure to store the game states and component attributes and relations, c) at least one game mechanic to give players something to do, and d) a *goal* that the mechanics are designed to help completing, combined with *end or victory conditions*.

Goal setting is a key aspect of rule setting in games. In line with this, designing interactive games for modelling can be fruitfully driven by *goals of modelling*. This concerns both *utility goals* (i.e. what the model/modelling is useful for) and *modelling goals* (i.e. sub-goals pertaining to details of the modelling process as such).

The typical *utility* goals are, of course, knowledge creation, description, and formalisation, but additional goals include organisational and individual learning, consensus building; ultimately, they relate to typical strategic business goals involving investment and some sort of gain (commercial or otherwise). This aspect is too often ignored when academics get involved in actual application of KA.

For an overview of key *modelling* goals we refer to [17]. The chief goals are:

- **Creation goals:** which items (documents, objects, conceptualizations) are to be delivered when playing the game;
- **Grammar goals:** which language rules (syntax, vocabulary, possibly also semantics) does the player need to comply to;
- **Validation goals:** what sort of agreements, about which items, and between whom, is required in the game.

A number of sub-goals can be distinguished underneath the main goals, like argumentation goals, sense making goals, proof goals, abstraction goals, and so on.

Goals, sub-goals, and combinations of goals can be set for concrete modelling sessions or activities, involving one or more participants. *End goals* may be worked towards via *intermediary goals*. Strategies and techniques can be selected and deployed to achieve specific goals for concrete situations [17], in line with goals set but also with resources available and capacities and attitudes of participants. Once clear goals are set, and made more concrete by means of the definition of (a hierarchy of) combined assignments, challenges, etc., we can move towards actual game design. Importantly, the assignments given to the players need not overtly reflect the utility and modelling goals that the game designers have in mind. Any assignment that appropriately focuses, guides, and stimulates the player(s) will do. In fact, creative invention of (combinations of) appropriate assignments is key to successful game design.

This brings us to an issue that is possibly the one farthest removed from classical thinking about modelling: motivation. In the gaming world (both academic and industrial), much purposeful thought has gone into ways of making games captivating [18]. Indeed, game designers have now become so good at this that serious addiction is sometimes the result. If we allow ourselves to run with the devil for at least a few yards, we might learn something about how to make dull or hard tasks (including collaborative ones) more pleasantly challenging, more easily learnable and doable, and generally more effective. Perhaps modelling does not always have to be *great fun*, but we may at least succeed in making it *less boring or more positively challenging* for an audience not intrinsically motivated by the challenges of creating good formal models. The game design ap-

proach to the support of modelling thus creates opportunities for *designing motivation*.

5. GUIDING CONCEPTUALISATION

Contrary to what is often assumed, knowledge modelling is not just a matter of “translating informal into formal language” [19]. In addition, and perhaps more fundamentally, formalization requires rational, “clean” construction of representations according to utilitarian rather than associative principles. It entails rationally governed construction (engineering) of conceptual structures conforming to conceptual patterns dictated by some formalism. Such rational construction needs to take place before actual formal representations are produced, and possibly even independent from a specific formal syntax and semantics. Skilled formalists can perform such analysis and construction implicitly, and thus can produce formal representations (though perhaps sketchy ones) as an initial product. Laymen need a much more gentle, stepwise form of guiding and structuring. If procedures for achieving this can be successfully created, lightweight formalization can be achieved without confronting a player with any form of math, or even a semi-formal diagram. Models are then not elicited directly, but indirectly. After a guided conversation in which specific knowledge descriptions are elicited stepwise, conform rational principles (rules) governed by well-defined “goals for modelling”, it should be possible to automatically derive formal representations based on strictly structured bits of natural language text, or simple visualizations, and the strictly governed relations between them.

From a method perspective this will force us to look at “pre-formal”, *intermediary* products that may include information that does not belong in the end product, but which is used to *derive* the end product (formal model) by means of reasonably basic reasoning.

For example, as illustrated in the middle column of figure 1, a Business Process Model in the standard language BPMN (Business Process Modelling Notation [20]) typically shows an ordering of activities, e.g. activities D and E must be completed before activity F can be started. However, the reason why this is the case is that D and E respectively produce entities *n* and *o* that are needed in F (resulting in what is technically called an “AND-join”). This is illustrated by the text in the leftmost and rightmost columns of figure 1, in which these entities and dependencies are made explicit. However, such dependencies and entities are not made explicit in a regular BPMN diagram, even if they are crucial for creating a useful, “good” one. As a consequence, the entities and dependencies involved are usually left implicit and exist only in the head of the modeller –if you are lucky. Even if the objects in the process are made explicit, perhaps in another model, they are not explicitly used as a basis for deriving AND-joins.

This idea inspired our first design of a game-for-modelling (briefly discussed below), which is not to say it will be the basis for all such games. Exploration of possibilities has only just begun.

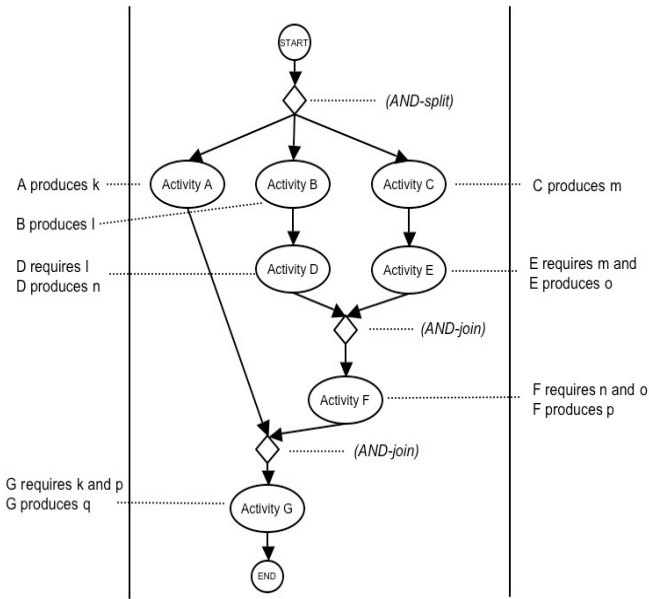


Figure 1: dependency information underlying a basic process model

6 SOME FIRST PROJECTS AND RESULTS

Various lines of work have been started in view of the “games for modelling” concept. The most fundamental line (in progress) concerns a methodology for the evaluation, in view of clear goals set, of modelling-activities as games. Rules are a crucial topic in this PhD project, directly linked to the game metaphor. The evaluation methodology is intended to serve as a key component in a design science cycle aiming at the development of principles and systems (i.e. games) for knowledge modelling support. The evaluation takes the shape of a transparent and traceable score system that is influenced by goals set for the game and weights assigned to them. This implies that not just the score system must be clearly (even formally) described, but also the game as such (i.e. all rules governing a particular interactive modelling session, *including* the goals set for it, in terms of concrete game results but also concerning collaboration, agreement, and efficiency).

As an initial part of the abovementioned project, we have performed an explorative study into “rule setting in modelling”, also based on the game metaphor [21]. Using qualitative research techniques, we recorded, coded, and analysed a semi-realistic, 18-minute collaborative modelling session involving three modellers. Applying the game metaphor, we reverse engineered the session as a game, identifying precise goals and rules by which the game was played. Results allowed us to perform a preliminary comparison of game-like methodological concepts with related work on the quality of modelling [17] and collaborative modelling [10]. In addition, we became very much aware that in collaborative modelling, modellers discuss and introduce rules of their own, i.e. shape their game play together, as *part* of the game. This leads us to distinguish “rules set *for* the game” and “rules set *within* the game”.

A second project that is well on the way concerns the design and implementation of a “Task Description Game”, which is

based on the idea presented in section 5. The game requires a player to describe some procedure (task; a favourite example is baking an apple pie) in terms of the items required to perform it and resulting from it, the steps taken, and various dependencies among steps and items used in/resulting from them. The player must fill in a number of form-like, interrelated “cards” (*item cards* and *step cards*), and adhere to explicit rules governing the relationships between the cards and their content fields until she feels confident enough to risk a “try”. After a try, a score is given at the hand of a set of scoring rules independent of the conceptual contents of the task description but rather guarding the rules constraining the description’s components and their interdependencies. Penalties are dealt if rules have been violated, and another try may have to be prepared. Multiple tries are allowed, but more tries does mean a lower score.

After the game finishes, a simple algorithm helps derive a BPMN model from the structured information gathered in the game.

Various evolved board game-like versions of the game have been exploratively tested on a small scale. The initial design remained roughly intact, but a main lesson learned is that playability depends on many small details in rules, and in adequate communication of those rules to the player. For example, it has to be made *very* clear to the player of a game-for-modelling that it is necessary to play by the rules even if this is more difficult than free-format and perhaps more intuitive description of the task; also, that something is *gained* by doing this (“a computer understanding the description”). We are now in the process of applying basic principles from gaming-oriented HCI [22] to evaluate the finalised game more systematically. An implementation of a digital version of the game is on the way.

In this first attempt to design a game-for-modelling, we have not yet aimed at the game actually being *fun* (note that Järvinen’s reasonably authoritative definition of “game” in section 4 in fact does not demand this), but merely for a *playable* game which should enable a layman to produce a simple formalisation “as a side effect”. Once the game is sufficiently playable, we do intend to improve its design to increase the fun factor, using insights as discussed in [24]. However, we do not necessarily expect the game to become “very much fun”; we would settle for “mildly entertaining”, as it also has a job to do.

Besides being a proof-of-concept for the idea of “formalization without formal language”, the digital Task Description game should allow for empirical data gathering on a sufficiently large scale, as it can be exposed on-line to a large population of players.

More distant from knowledge modelling, but nevertheless strongly related, is the use of a game setup in testing a user friendly method for query formulation (the Interactive Query Language or IQL) intended to provide a more user friendly alternative for SQL [23]. In a game context, players use either IQL or SQL to answer questions by means of querying a fixed data set; results have led to preliminary proof that IQL speeds up the process of query formulation, and is easily learnable. Possibilities to extend the IQL approach to rule based modelling (in this case, Business Rules) are being considered.

Another project that is in its write-up phase concerns exploration of possibilities to introduce gaming aspects in an existing, operational industrial Business Engineering environment, with a strong AI interest (rule based modelling). Emphasis in this project is on focusing and motivating teams of business engineers

(and others involved). The project is rendering considerable insights into design and evaluation of emotion and motivation, merging method engineering, HCI, and game psychology [24].

In addition, we are in the process of designing three more games, one aiming at value chain modelling, one concerning simulation (experiencing) and manipulation of enterprise architecture models, and one exploring possibilities to capture strategies for interactive formal proofing (natural deduction) within a game.

7 CONCLUSIONS AND FURTHER WORK

We discussed a Gaming approach to methods for knowledge elicitation and formalisation, in view of attempts to break, or at least widen, the Knowledge Acquisition Bottleneck. We discussed our perspective on the KAB, presented arguments for introducing the Game Metaphor, discussed the link between game design and operational method design, and provided an exemplary discussion of a conversation based, low threshold approach for guiding lightweight conceptualisation without using formal representations. Next we listed a number of projects within the frame of the Games For Modelling concept, thus providing some illustrations for it.

We realize that so far, no substantial published results have ensued. This is mostly because there simply has not been much time to do so: projects were all initiated less than a year ago. Still, we believe the concept as such is interesting enough to report on, and we hope it may fire up a discussion on the KAB and “methods embodied as games”.

Further work was partly covered by the previous section, but in general concerns the continuation of our effort to design, test, and improve various sorts of games-for-modelling. This extends to generic, fundamental aspects like the development of adequate metrics for the quality of the games and the models they bring forth [17] and the development of design principles. In addition, we continue our exploration of theoretical contributions from AI, HCI, collaborative systems, and psychology/cognition (to name but a few) that might help us understand and further the creation of interactive systems for supporting formal modelling.

Acknowledgement We want to thank Denis Ssebuggawo, Bart Schotten, Dennis Aarts, Jeroen Claassen, and Ilona Wilmont, to who’s work we refer in section 6.

8 REFERENCES

- [1] A. Th. Schreiber, J. Akkermans, A. Anjewierden, R. De Hoog, N. Shadbolt, W. Van De Velde & B. Wielinga: *Knowledge Engineering and Management: The CommonKADS Methodology*. Cambridge, MA, USA: MIT Press., 1999.
- [2] F. Hayes-Roth, D.A. Waterman and D.B. Lenat. *Building Expert Systems*. PA: Addison-Wesley, 1983.
- [3] M. Berthold and David J. Hand (Eds.): *Intelligent Data Analysis, An Introduction*. Springer-Verlag, Berlin, 1999.
- [4] C. Wagner: Breaking the Knowledge Acquisition Bottleneck Through Conversational Knowledge Management. In: *Information Resources Management Journal*, 19(1), 70-83, January-March 2006.
- [5] A. Newell and H.A. Simon: *Human Problem Solving*. Prentice-Hall, Englewood Cliffs, NJ., 1972.
- [6] M. Cristani, and R. Cuel: A Survey on Ontology Creation Methodologies. In: *International Journal on Semantic Web & Information Systems*, Vol. 1, Issue 2, p. 49 – 69, 2005.
- [7] P. Kruchten: *The Rational Unified Process: An Introduction*. 2nd edition. Addison Wesley, 2000.
- [8] G. Booch, J. Rumbaugh, and I. Jacobson. *The Unified Modelling Language User Guide*. New York: Addison Wesley, 1998.
- [9] N. Kock and P. Rittgen, editors: Collaborative Business and Information Systems Design, special issue of the *Journal International Journal of e-Collaboration (IJeC)*. Information Resources Management Association. Hershey, USA: IGI Global. 2009 (in print).
- [10] P. Rittgen: Collaborative Modelling – A Design Science Approach, *Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS-42)*, Waikoloa, Big Island, Hawaii, USA, January 5-8, 2009, CD-ROM, Los Alamitos, CA: IEEE Computer Society, 2009, 10 p.
- [11] Diaz, A., Baldo, G. and Canals, G. Co-Protégé: Collaborative Ontology Building with Divergences. *Proceedings of the 17th International Conference on Database and Expert Systems Applications* p156 – 160. 2006. Washington, DC, USA IEEE Computer Society.
- [12] S.J.B.A. Hoppenbrouwers, H.A. Proper, and Th.P. van der Weide. Formal Modelling as a Grounded Conversation. In: G. Goldkuhl, M. Lind, and S. Haraldson, editors, *Proceedings of the 10th International Working Conference on the Language Action Perspective on Communication Modelling (LAP’05)*, pages 139–155, Kiruna, Sweden, EU, June 2005. Linköpings Universitet and Hogskolan I Boras, Linköping, Sweden, EU.
- [13] G.E. Veldhuijzen van Zanten, S.J.B.A. Hoppenbrouwers, and H.A. Proper. System Development as a Rational Communicative Process. In: *Journal of Systemics, Cybernetics and Informatics*, Nr: 4, Vol: 2, International Institute of Informatics and Systemics (IIS), 2004.
- [14] Elgood, C. (1993): *Handbook of Management Games*, 5th ed. Aldershot: Gower Publishing.
- [15] S.J.B.A. Hoppenbrouwers, P. van Bommel, and A. Järvinen. Method Engineering as Game Design: an Emerging HCI Perspective on Methods and CASE Tools. In: *Proceedings of EMMSAD’08 (Exploring Modelling Methods for System Analysis and Design)*, held in conjunction with CAiSE’08. Montpellier, France, June 2008.
- [16] L. Von Ahn: Games With a Purpose. In: *Computer* 39, 6 (Jun. 2006), 92-94.
- [17] P. van Bommel, S.J.B.A. Hoppenbrouwers, H.A. Proper, and J. Roelofs. Concepts and Strategies for Quality of Modelling. In: T. Halpin, J. Krogstie, and E. Proper: *Innovations in Information Systems Modelling, Methods and Best Practices*; Chapter IX, p167-89. Advances in Database Research series, IGI Global Publishing, USA, 2008.
- [18] K. Salen, and E. Zimmerman: *Rules of Play, Game Design Fundamentals*. Cambridge, MA: MIT Press.
- [19] S.J.B.A. Hoppenbrouwers: Community-based ICT Development as a Multi-Player Game. In: *Proceedings of International Conference “What is an Organization? Materiality, Agency and Discourse”*, Montreal, May, 2008. Dept. of Organizational Communication, University of Montreal.
- [20] Object Management Group (OMG): *BPMN 1.0, OMG Final Adopted Specification*, 2006.
- [21] D. Ssebuggawo, S.J.B.A Hoppenbrouwers, and H.A Proper: *Analyzing a Collaborative Modeling Game*, Submitted to the 21st International Conference on Advanced Information Systems Engineering (CAISE 2009).
- [22] Heather Desurvire, Martin Caplan and Jozsef A. Toth: Using heuristics to evaluate the playability of games. In: CHI ’04, extended abstracts on human factors in computing systems p15-12. New York: ACM.
- [23] J. Claessens: *The Expressive Powers of an Interactive Query Language Compared to SQL*. Technical report, Radboud University Nijmegen, 2007
- [24] I. Wilmont: *A Gaming Approach to Collaborative Modelling*. Master’s Thesis, Radboud University Nijmegen, 2009.