Collaborative, Participative and Interactive Enterprise Modeling

Joseph Barjis

Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands
j.barjis@tudelft.nl

Abstract. Enterprise modeling is a daunting task to be carried out from a single perspective. A challenge to this whole complexity is conflicting descriptions given by different actors when business processes are documented. Often enterprise modeling takes rounds of iterations and clarification before the models are verified and validated. In order to expedite the modeling process and validity of the models, in this paper we propose an approach called collaborative, participative, and interactive modeling (CPI Modeling). The main objective of the CPI approach is to furnish an extended participation of actors that have valuable insight into the enterprise operations and business processes. Achieving this goal with any modeling method and language could be quite challenging. For CPI Modeling to succeed the modeling method should adhere to certain qualities. Next to the CPI Modeling approach, this paper discusses an enterprise modeling method that is simple, and yet powerful to capture intricate enterprise processes and simulate them.

Keywords: CPI Modeling, Collaborative modeling, Interactive modeling, Participative modeling, Business process modeling, Business process simulation, Enterprise modeling, Enterprise simulation, DEMO methodology, Language-action perspective, Petri net.

1 Introduction

Enterprise modeling is a daunting task and usually error-prone process. The problem is not the maturity of the modelers and analysts involved, but the complex socio-technical nature of an enterprise. In particular, enterprise modeling in its broader context encompasses business processes, human and organizational issues, technical aspects such as information systems and enabling IT applications. The enterprise modeling challenge could be best seen in the definition given in [14], which is defined as “computational representation of the structure, activities, processes, information, resources, people, behavior, goals and constraints of business, government, or other enterprise”. Basically, this definition alone suffices to persuade the reader that a traditional way of enterprise modeling will not yield too much success if non-innovative approaches and methods deployed.

A traditional approach towards enterprise modeling, especially enterprise processes modeling is to delegate the work to analysts and modelers who will be normally visiting the enterprise under study, read the existing documentation, conduct a series

J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 651–662, 2009.

of interviews, after which enterprise models are developed by the modelers alone. Only several back and forth visitation and refinements will allow for a more complete model of the enterprise to emerge. This approach has a number of problems. As the interaction of the analysts with the enterprise employees become more and more often, the enterprise becomes more reluctant and less interested to allocate their most needed human resources to be involved in the project, which will be seen as waste of time. This is the reason that many enterprises will consider the modeling part as hindering the IS development project. In turn, modelers, not having sufficient rounds of iteration, will end up with a model that is either incomplete, or there are many assumptions that are intuitively made by the molders. As a result, the model may contain a lot of flaws. These flaws remain quite undetected as majority of enterprise process modeling is not based on formal semantics to check the models and simulate their dynamic behavior.

A comprehensive enterprise modeling requires that the models have to capture three phenomena such as enterprise processes, enterprise business rules, and enterprise information, which should be integrated in the corresponding information system deliverable [23]. Thus, it is very important that enterprise models are accurate and complete as flawed models will result in inadequate final systems, especially that system developers may rely on the models for developing an actual system as many of these complex systems are already driven by models – model-driven system development.

Traditional approaches of enterprise modeling, where modelers and analysts are the players and the rest (process owners, mangers, stakeholders, experts) are either passive participants or even absent from the scene, are least likely to result in accurate models. Firstly, the modern enterprise business processes are too complex to be understood, captured and documented by modelers and analysts alone. Secondly, because the resultant models should be approved by all stakeholders and decision makers, and only then the models can be implemented.

In order to address theses challenges in enterprise modeling, innovative approaches have been discussed and introduced such as participative enterprise modeling [22, 27]. A central goal of enterprise modeling is to discover domain knowledge and document the enterprise existing business processes. The role of participative modeling is to represent this knowledge in a coherent and comprehensive model, create shared understanding, consolidate different stakeholder views, and in order to do so an extended participation of stakeholders is crucial [25].

From the foregoing, it becomes obvious that participative enterprise modeling is rather a necessity than a choice. However, two other things that need to be considered for a successful participation are collaboration and interaction. We refer to this approach as collaborative, participative, and interactive modeling (CPI Modeling). It is imperative that each of the three notions is given explicit attention, which we will do in the following section.

2 Collaborative-Participative-Interactive Enterprise Modeling

In this section we will explain why the emphasis of this work is specifically on 'collaboration', 'participation', and 'interaction' as three constituents of success for complete, accurate, acceptable, and expedited enterprise modeling.

2.1 Collaboration

- Collaboration has been proven to result in a fast and accurate model development when modeling of complex phenomenon of socio-technical nature is involved. But collaboration itself is a subject of engineering approach, where explicit scenarios and guidelines are required to be designed for facilitating collaboration.
- Collaboration also requires guidance and orchestration by an experienced facilitator. Facilitators with modeling experience and basic domain knowledge are more successful to lead the participants.
- Even a more extended collaboration of interdisciplinary nature (analysts, consultants, IT professional, social researchers) would be required to cope with complex design objects and propose innovative solutions.
- Finally, collaboration requires that modelers (with knowledge of modeling language and techniques) and analysts (with expertise in analysis and modeling) are collaborating to design accurate models.

2.2 Participation

- Often, the goal of enterprise is to document the operations and business processes, and create shared understanding of what and how an enterprise is operating. This shared understanding, which is achieved through a complete picture of the enterprise business processes, would be difficult without participation of key employees of units involved by implication.
- Often, also enterprise models convey different accounts of business processes from different unit's perspectives. Numerous iterations of enterprise modeling are required to build an accurate model, which makes the process tedious, extensive and costly. Consolidation of different accounts and expedition of enterprise modeling are another challenge that is hard to cope with, unless the modeling involves participation of process owners.
- Often, the ultimate deliverables of enterprise modeling are changes in the current practice, organizational restructuring, or investment in new technology. It will be hard to achieve these ultimate goals without an extended participation and approval of stakeholders.
- Finally, verification and validation of complex models have always been challenging. Presence of the process owners and business units managers and their participation in the modeling will result in immediate verification and validation of the models, especially when simulation methods are deployed in the process, which demonstrate enactment of the models.

2.3 Interaction

- Innovative tools and technologies are needed to furnish interaction of modelers, analysts, and participants of enterprise modeling. While technologies such as large interactive smart boards (including the one that can be shared remotely) are creating interactive environment, these are specific tools that furnish the success.
- Tools should be intuitive, easy to follow, and powerful to capture complex interactions.

- Static models are no longer sufficient to create the shared understanding that complex enterprise process models should accomplish. Tools that simulate the processes allow to capture dynamic behavior of the constructed models, observe the effects of changes, and manipulate the models.

These three aspects constitute the so called CPI Modeling approach, where each aspect is a dimension: the collaboration aspect represent the *Experts* (analysts) dimension; the participation aspect represents the *Users* (stakeholders) dimension; and the interaction aspect represents the *Technology* (tools) dimension. However, these constituents comprise only the approach we adapted for enterprise modeling. The next challenge is the method, language, and notations used for enterprise modeling. The question is can we use UML and IDEF with the same level of success? Is there any advantage of using EPC over UML? Or none of these methodologies are suitable to support the CPI Modeling approach that requires interactive modeling with different participants onboard (from mature business analysts and professional molders to employees that have no technical or modeling background). In the following section, we briefly discuss why some enterprise modeling methodologies may not be suitable.

3 The Modeling Method Consideration

Depending on specific situations and contexts, enterprise analysts can develop a prototype of the envisioned system, and study their behavior [4]; the analyst can develop mathematical models [3, 15] and abstractions of systems and study them by calculating output parameters of the models; they can draw static pictures using diagrams and then study the diagrams such as IDEF [16, 21], UML [7, 17, 26], EPC [11], Petri Nets [2], etc. Each of these approaches presents certain benefits and, of course, certain limitations and drawbacks. However, diagrammatic representation of models represents enormous interests and practical value as it poses least cognitive load and great communication capability [18, 19]. In selecting a diagrammatic method, it is important that the chosen approach (method and tool) adequately fits the problem situation. It is extremely important for the system's ultimate success to ensure the quality of the modeling methods and tools [6].

As discussed in [20], there are certain quality attributes that a modeling method should adhere to such as syntactic, semantic, and pragmatic qualities. Syntactic qualities require rules and grammar that drive modeling and prevent construction errors. In pragmatic qualities, strong emphasis is put on executability of models, their visualization, simulation and animation. Adding to these qualities, in fact, CPI modeling implies participation of non-modelers and non-analysts, therefore ease of use, natural compliance with the way the organization conducts its operations, and intuitiveness require a huge extent of model simplicity. Model simplicity means that the modeling method should be easy and simple for understanding and construction, and yet powerful to capture the complexity of underlying situation.

For example, UML is difficult to learn, it is complex, and there seem to be too many diagrams [24]. For that reason, not all of the UML diagrams are used by analysts [10]. Therefore, to use UML as a modeling language in CPI approach will pose certain challenges. Moreover, an enterprise is a social environment where human actors naturally interact while requesting actions or committing themselves to certain actions. Therefore capturing these social characteristics will definitely surface the requirements for an enterprise modeling method.

Simulation of models in a CPI environment will add a lot of value for sharing understanding. Therefore, the modeling language should lend to simulation and allow to check the models consistency and completeness. We propose that Petri net possesses balanced properties (expressivity, intuitiveness, formal semantics) to serve the purpose. Most of the conventional models are checked and analyzed via translation to other formal diagrams using mapping procedures. For instance, UML activity diagrams are often translated to Petri nets for checking [12, 13]. Another widely accepted method, investigated in [8], is Event-driven Process Chain (EPC). The authors propose a 5-step guideline to translate EPC models to Petri net models in order to investigate whether the process is correctly described in EPC. The analysis showed that ambiguities of EPC models will result in faulty Petri net executions. Finally, IDEF diagrams are also semi-formal diagrams that present little pragmatic value in collaborative and participative modeling environment, where simulation of the models is very important.

As for the modeling notations that compete with Petri net, e.g. BPMN, EPCs, Role-Activity-Diagrams, IDEF, UML, RIVA etc., Petri net is known for rigorous semantics, logics and formalism, and also is widespread among researchers, practitioners and a variety of academic disciplines. In addition, Petri net is supported by a large number of tools for its analysis. In [1], author identifies three main reasons why Petri net possesses advantageous features: formal semantics despite the comprehensive graphical representation; state-based representation instead of event-based; abundance of analysis techniques. Process modeling techniques ranging from informal techniques (e.g., dataflow diagrams) to formal ones (e.g., process algebra) are event-based, while Petri net approach allows state-based modeling.

The enterprise modeling method we propose is based on the DEMO transaction concept (developed for social systems) and Petri net graphical notations lending to simulation.

4 DEMO Transaction

In this section we first briefly discuss the DEMO transaction's original diagram and concept, and then we introduce a diagram, which is based on Petri net semantics. We use Petri net graphical notations to allow the resulting model to be simulated.

4.1 Original Notations

This section is based on the original works conducted in the framework of DEMO methodology. The results of more than a decade development of this methodology are summarized in [9]. DEMO is an acronym for Design and Engineering Methodology for Organizations (see www.demo.nl for more information).

According to the DEMO theory, social actors in organization perform two kinds of acts: *production act* (P-acts, for short) and *coordination act*s (C-acts, for short). By engaging in P-acts, the actors bring about new results or facts, e.g., they deliver service or produce goods. Examples of P-acts are: register a student into new course;

issue a ticket for a show; make a payment. By engaging in C-acts, the actors enter into communication, negotiation, or commitment towards each other. Examples of C-acts are: making a request for new course; presenting an issued ticket to the customer. The generic pattern in which the two kinds of actions (P-acts and C-acts) occur is called *transaction*, see Figure 1. In fact, a transaction is steps of **C-act→P-act→C-act** that correspondingly result in C-fact (e.g., commitment to register a student) and P-fact (e.g., do register a student).

A transaction is carried out in three phases: the *Order* phase (O-phase, for short), the *Execution* phase (E-phase, for short) and the *Result* phase (R-phase, for short). These three phases involve two actor roles. The actor role that initiates a transaction is called *initiator*. The actor role that carries out a production act is called *executor*.

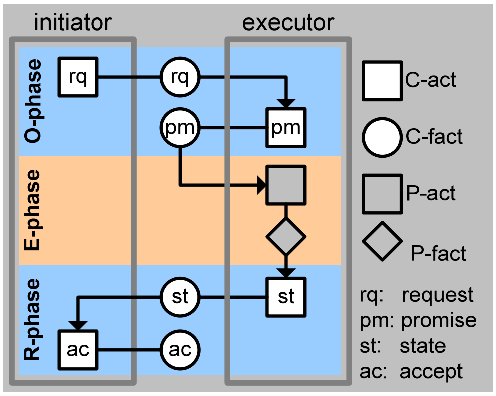

Fig. 1. Basic Transaction Concept (adapted from Dietz 2006)

In the following sub-section we introduce and discuss the extensions we made to the DEMO transaction diagram based on Petri net. In the terminologies used, we return to earlier terms used for P-acts and C-acts, i.e., instead of P-act, we refer to it as *action*, and C-act as *interaction*.

4.2 Extended Notations

A business transaction is a pattern of action and interaction. An *action* is the core of a business transaction and represents an activity that brings about a new result. An *interaction* is a communicative act involving two actor roles to coordinate or negotiate a particular action.

Each business transaction is carried out in three distinct phases (see Figure 2a):

- *Order phase (O)*, during which an actor makes a 'request' for a service or good towards another actor. This phase represents a number of communicative acts or interactions. This phase ends with a commitment ('promise') made by the second actor, who will deliver the requested service or good.

- *Execution phase (E)*, during which the second actor fulfills its commitment, i.e., 'produce' the service or good. This phase represents a productive act.

- *Result phase (R)*, during which the second actor does 'present' the first actor with the service or good prepared. This phase also represents a number of communicative acts or interactions. This phase ends with the 'accept' of the service or good by the first actor.

These phases are abbreviated as O, E and R correspondingly (see Figure 2b). The figure illustrates a business transaction in detailed generic form and simple OER form. Note that the order (O) and result (R) phases are interactions and the execution (E) phase is an action.

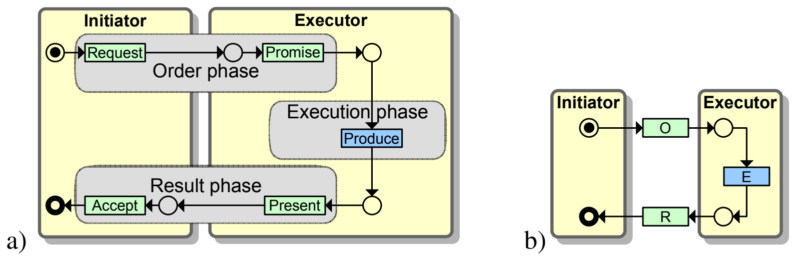

Fig. 2. Business transaction: a) a generic form, b) an OER form

In a structured language, a transaction is described according to Table 1, where a transaction is portrayed through the *activity pattern* it represents (e.g., placing an order), its *initiator* (e.g., customer), *executor* (e.g., supplier), and the *result* it delivers (e.g., a new order is created).

Table 1. Transaction description in a structured language

Transaction:	**Atomic process (e.g., *placing an order)***
Initiator	Name of the role that initiates the transaction (e.g., *customer)*
Executor	Name of the role that executes the transaction (e.g., *supplier)*
Result	The result created as the transaction is carried out (e.g., *a new order is created)*

Now that we have discussed the CPI Modeling approach and the modeling method supporting it, we introduce a case study where we tested both the approach and the modeling method. However, we have to skip a whole set of notations and modeling constructs we have developed that can be used as building block or components in enterprise modeling. The interested reader is referred to [5] for more reading on modeling notations and constructs developed based on the DEMO transaction concept and Petri net graphical semantics.

5 Case Study: DutchPlast BV Enterprise

This case study was conducted on DutchPlast BV, a plastic production company, located in Westland-Area, The Netherlands. The company recently launched an initiative to review its business processes and improve the current processes with reducing delays, and developing new information systems that will support the redesigned business processes.

5.1 CPI Enterprise Modeling

In the CPI Modeling approach we applied for conducting enterprise modeling of DutchPlast, we pretty much followed the recommendations for participative modeling suggested in [25]. However, we put strong emphasis on interactive modeling, tools and technology. We organized a half day session with the following participants:

- DutchPlast – technical director and order and procurement director – business process owners with high expertise and authority about the enterprise operations.
- Enterprise modeling expert – an author of enterprise modeling methodology
- Modelers – expert modelers.
- Facilitator – a professional collaboration facilitator with PhD and expertise in using interactive smart-boards
- Observers – a group of observers to document the session.
- Graduate students – a group of graduate students who are graduated from an enterprise modeling course.

The modeling session was conducted on a large interactive smart-board allowing use of electronic color pens on a touch-screen. The modeling covered the whole enterprise business processes.

The end of the session feedback and discussion allowed to learn about the importance of simple and intuitive modeling notations. What was the most important feedback received is that the participants would prefer a simulation model over static diagrammatic models, although we have not conducted the simulation part in the session. The processes that we modeled are described in the following sub-section.

5.2 The Customer Order Process

It should be noted that the description presented in this section is significantly reduced for this paper. Accordingly, the number of transactions and the enterprise processes model are also reduced.

A customer wants DutchPlast to produce a product. He can send an e-mail or fax with a short description of the desired product to a salesperson. The customer can also contact the salesperson directly by telephone. The salesperson needs to have as much information (colour, measurements, etc.) as needed about the product. The salesperson uses this information in a calculation program to estimate the costs. Also the delivery time is estimated based on the planning in the information system. A salesman creates an offer with the costs and delivery date. The offer is sent to the customer. The customer may decline the offer or accept.

The order can either be a standard product or a customized product. A standard product is easy to produce as information for production is already available. A customized product requires designing and work preparation. For customized products, the technical designer sketches the product based on the information in the order. If the product cannot be designed because information is missing or incorrect

Table 2. DutchPlast business transactions

T1:	**Making an offer**
Initiator	Customer
Executor	Salesman
T2:	**Placing an order**
Initiator	Customer
Executor	Salesman
T3:	**Produce the product**
Initiator	Salesman
Executor	Product producer
T4:	**Check the product quality**
Initiator	Salesman
Executor	Production manager
T5:	**Internal transportation**
Initiator	Salesman
Executor	Internal transporter
T6:	**External transportation**
Initiator	Salesman
Executor	External transporter
T7:	**Pay external transportation**
Initiator	External transporter
Executor	Salesman
T8:	**Pay the order**
Initiator	Salesman
Executor	Customer
T9:	**Report to collection agency**
Initiator	Salesman
Executor	Collection agency
T10:	**Recurring stock control**
Initiator	Stock controller
Executor	Stock controller
T11:	**Restocking**
Initiator	Salesman
Executor	Supplier
T12:	**Pay for restocking**
Initiator	Supplier
Executor	Salesman
T13:	**Production preparation**
Initiator	Salesman
Executor	Production manager
T14:	**Create technical drawing**
Initiator	Production manager
Executor	Technical drawer
T15:	**Customer approval**
Initiator	Salesman
Executor	Customer
T16:	**Production planning**
Initiator	Salesman
Executor	Production planner
T17:	**Order required materials**
Initiator	Salesman
Executor	Supplier
T18:	**Pay for the materials**
Initiator	Supplier
Executor	Salesman

in the order, he contacts the customer to supply the needed information. Also new materials will need to be ordered if a customized product requires it.

A standard product is produced using an open mould technique. After preparing everything for production the product is ready to be assembled. The producer will assemble the product.

When a product has been produced, it's checked by the production manager for quality purpose. Only then transportation of the product is arranged. When the delivery is within the Westland-area, the transportation is being done by the internal transportation department of DutchPlast. Otherwise, it will be outsourced to an external transport company. In that case, DutchPlast also has to pay the external transport company.

On a monthly basis, the financial administrator checks for unpaid bills. He will then contact the client or contact a collection agency to handle the debtor.

In the following section, based on the above description, we try to identify business transactions according to the DEMO transaction concept.

5.3 DutchPlast Business Transactions

The business transactions contained in Table 2 were identified in a collaborative manner with participation of experts, business process owners and business analysts using the description of business process of the previous section.

This list of transactions served as an input for the enterprise model construction. A complete model of the DutchPlast enterprise was developed in an interactive manner. Since the model itself is not in the scope of this paper, we did not include the model into this paper, however, through the list of business transactions included in the table and the related actors (units) we want to convey the whole complexity that an enterprise model may present. The purpose for this case study was to apply the CPI Modeling approach and reflect on the experience. Some conclusions are drawn in the following section where we discuss the CPI Modeling experience.

6 Conclusions

In this paper we have discussed the CPI Modeling approach and its importance for enterprise modeling, especially enterprise business processes. We also discussed that CPI approach can be only useful if it is supported by a suitable enterprise modeling method and language that is simple, and yet powerful.

As the DutchPlast case study revealed, an enterprise process model can be very complex. Although we omitted almost half of the case, still the processes are complex enough to be captured without collaboration of business modelers, analysts, facilitator and participation of process owners and managers. The CPI modeling approach not only helps with the production of enterprise models and expedites the modeling process, but it allows validation and verification of the model almost immediately. To this end, both the participation and presence of the business owners and simulation of the model can help (in the DutchPlast case we skipped simulation due to time limitation).

Application of suitable methodology is crucial to communicate among the participants. A method that consists of a moderate number of elements, intuitive notations, and closely resembles the natural way of working in an enterprise has significant advantage as the participants do not have modeling background. A small set of elements reduces cognitive load of the participants not familiar with the modeling method. The modeling method should fit into the social sitting of an enterprise, which is a social system by virtue.

The discussed case study is a starting point for more extended research. The CPI Modeling approach opens up an opportunity for many potential directions, into which the CPI modeling approach can be developed. First of all, we intend to create a rich simulation environment around this approach and method allowing simulation of enterprise processes with animative features including building a library of customized entities. This will create a more realistic replica of the enterprise under study. Another interesting research topic would be to develop scenario based guidelines that will allow practitioners to apply CPI approach without extensive involvement of a facilitator. One more interesting observation that also leads to some research was that the participants did not fully collaborate at the beginning of the CPI Modeling session. This means that fro a full collaboration of the participants certain procedures and measures should be developed.

Acknowledgements. It is a pleasure to thank several people who participated in the session: first of all, the author of the DEMO Methodology for his participation in the session as an expert; the DutchPlast senior employees (commercial director, technical director); the facilitator of the session; colleagues and graduate students who helped with observing the session.

References

1. van der Aalst, W.M.P.: Three Good Reasons for Using a Petri-net-based Workflow Management System. In: Proceedings of the International Working Conference on Information and Process Integration in Enterprises (1996)
2. van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.): Business Process Management: Models, Techniques and Empirical Studies. Springer, Heidelberg (1998)
3. Aris, R.: Mathematical Modelling Techniques, New York (1994)
4. Arnowitz, J., Arent, M., Berger, N.: Effective Prototyping for Software Makers. Morgan Kaufmann, Elsevier, Inc. (2007)
5. Barjis, J.: The Importance of Business Process Modeling in Software Systems Design. Journal of The Science of Computer Programming 71(1), 73–87 (2008)
6. Bollojy, N., Leung, S.S.K.: Assisting Novice Analysts in Developing Quality Conceptual Models with UML. Communications of the ACM 49(7) (July 2006)
7. Booch, G., Rumbaugh, J., Jacobson, I.: The Unified Modelling Language User Guide. Addison-Wesley, Reading (1999)
8. Dehnert, J., van der Aalst, W.M.P.: Bridging the Gap Between Business Models and Workflow Specifications. Int. Journal of Cooperative Information Systems 13(3), 289–332 (2004)
9. Dietz, J.L.G.: Enterprise Ontology –Theory and Methodology. Springer, Heidelberg (2006)

10. Dobing, B., Parsons, J.: How UML is used. Communications of the ACM 49(5) (May 2006)
11. van Dongen, B.F., van der Aalst, W.M.P., Verbeek, H.M.W.: Verification of EPCs: Using Reduction Rules and Petri Nets. In: Proceedings of the 17th Conference on Advanced Information Systems Engineering (2005)
12. Eichner, C., Fleischhack, H., Meyer, R., Schrimpf, U., Stehno, C.: Compositional Semantics for UML 2.0 Sequence Diagrams Using Petri Nets. In: SDL Forum 2005, pp. 133–148 (2005)
13. Eshuis, R.: Symbolic Model Checking of UML Activity Diagram. ACM Transactions on Software Engineering and Methodology 15(1) (January 2006)
14. Fox, M.S., Gruninger, M.: On Ontologies and Enterprise Modelling. In: International Conference on Enterprise Integration Modelling Technology 1997 (1997)
15. Gershenfeld, N.: The Nature of Mathematical Modeling. Cambridge University Press, Cambridge (1998)
16. IDEF, Family of Methods web page (2003), `http://www.idef.com`
17. Jacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process. Addison Wesley Longman (1998)
18. Koehler, J., Hauser, R., Küster, J., Ryndina, K., Vanhatalo, J., Wahler, M.: The role of visual modeling and model transformation in business-driven development. ENTCS, vol. V, p. 211 (2008)
19. Larkin, J.H., Simon, H.A.: Why a diagram is (sometimes) worth ten thousand words. Cognitive Science 11(1), 65–100 (1987)
20. Lindland, O., Sindre, G., Sølvberg, A.: Understanding quality in conceptual modeling. IEEE Software 11(2), 42–49 (1994)
21. Mayer, R.J., Painter, M., deWitte, P.: IDEF Family of Methods for Concurrent Engineering and Business Re-engineering Applications. Knowledge Based Systems, Inc. (1992)
22. Persson, A.: Enterprise Modelling in Practice: Situational Factors and their Influence on Adopting a Participative Approach, PhD dissertation, Stockholm University (2001)
23. Prakash, N.: Bringing Enterprise Business Processes into Information System Products. In: Stirna, J., Persson, A. (eds.) The Practice of Enterprise Modeling. LNBIP, vol. 15 (2008)
24. Siau, K., Cao, Q.: Unified Modeling Language (UML) - a complexity analysis. Journal of Database Management 12(1), 26–34 (2001)
25. Stirna, J., Persson, A., Sandkuhl, K.: Participative Enterprise Modeling: Experiences and Recommendations. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 546–560. Springer, Heidelberg (2007)
26. Torchiano, M., Bruno, G.: Enterprise modeling by means of UML instance models. SIGSOFT Software Engineering Notes 28(2) (2003)
27. de Vreede, G.-J.: Participative Modelling for Understanding: Facilitating Organizational Change with GSS. In: Proceedings of the 29th HICSS (1996)