

Using Roles and Business Objects to Model and Understand Business Processes

Artur Caetano
artur.caetano@dei.ist.utl.pt

António Rito Silva
rito.silva@dei.ist.utl.pt

José Tribolet
jose.tribolet@dei.ist.utl.pt

Organizational Engineering Center, INESC INOV &
Department of Information Systems and Computer Science,
Instituto Superior Técnico, Technical University of Lisbon,
Av. Rovisco Pais, 1049-001 Lisboa, Portugal.

ABSTRACT

Business process modeling focus on describing how activities interact with other business objects while sustaining the organization's strategy. Business objects are object-oriented representations of organizational concepts, such as resources and actors, which collaborate with one another in order to achieve business goals. These objects exhibit different behavior according to each specific collaboration context. This means the perception of a business object depends on its collaborations with other objects. Business process modeling techniques do not clearly separate the multiple collaborative aspects of a business object from its internal aspects, making it difficult to understand objects which are used in different contexts, thus hindering reuse. To cope with such issues, this paper proposes using role modeling as a separation of concerns mechanism to increase the understandability and reusability of business process models. The approach divides a business process model into a business object model and a role model. The business object models deals with specifying the structure and intrinsic behavior of business objects while the role model specifies its collaborative aspects.

Categories and Subject Descriptors

D.2.9 [Management]: Software Process Models

D.2.10 [Design]: Methodologies

H.1.0 [Models and Principles]: General

Keywords

Business Process Modeling, Role Modeling, Business Object, Organizational Engineering

1. INTRODUCTION

Representing and keeping the alignment between the multiple elements of an organization is fundamental to understand how it operates and how it can be adapted to cope with a changing business environment [5]. This requires understanding how business activities interact and are aligned with other organizational elements while supporting the operation of the business.

In the past years, significant work, particularly in the area of business process modeling has been proposed, ranging from gen-

eral modeling concepts to business automation languages [10, 16, 17, 18]. Business process modeling can be used for multiple purposes, such as facilitating human understanding and communication [29], supporting process improvement and re-engineering through business process analysis and simulation [8, 17] and automating the execution of business processes [1, 22].

A business process model captures the relationships that are meaningful to the business between different organizational concepts, such as activities, the resources used by activities and the human or automated actors who perform these activities. Identifying the properties and relationships of these concepts is fundamental to help understanding and evolving the business since it facilitates the communication between stakeholders, business specialists and support system specialists.

We model business concepts as classes of business objects in a consistent object-oriented glossary of business concepts from where objects can be composed, specialized and reused.

However, fully characterizing the type of a business object, its properties and relationships is not straightforward. This results from a business object generally being used in different contexts and relating to several other business objects in the organization.

For example, a business object modeling a Product may be brought into play in several processes, such as Manufacturing, Logistics and Selling. In each of these contexts, it relates with different activities and resources, displaying different and possibly overlapping properties and behavior that are context-dependent. This means the object acts as a multi-dimensional concept.

If business objects are modeled as one-dimensional concepts, i.e. without its properties and behavior being described as dependent on the context, then the objects will not have explicit information on how to guide the design of a business support system that is able to cope with evolution. For example, if the Manufacturing process changes, there may be changes to the Product object. However, if the Product object does not explicitly represent the aspects related to its manufacture, then there will be no information on the properties requiring modifications.

This paper focus on describing on how to break up the universe of process modeling and its business objects into different aspects or areas of concern, of which each can then be handled independently and later composed to synthesize a complete model. To do so, we propose defining two complementary conceptual models, a role model and a business object model. The role model describes business object collaborations and the properties of business objects that are concerned with each role, being each role a type on its own. The business object model describes the structure and the properties of a business object that are independent of a specific context. The relationships between business objects are specified

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SAC'05, March 13–17, 2005, Santa Fe, NM, USA.

Copyright 2005 ACM 1-58113-964-0/05/0003...\$5.00.

by roles the objects play while collaborating. We argue that using roles and business objects to model business processes improves the understandability of the individual business objects and of the process model. It also improves model reengineering since it promotes reuse and makes explicit the dependencies between the model elements.

The remainder of this paper is structured as follows: next section reviews some of the research on business process modeling. Section 3 reviews role modeling, describes how roles can be identified and defines the concepts of business objects and role. Section 4 presents how the business object and the role model can describe a business process, followed by an example of application in section 5. Finally, section 6 sets out the conclusions and future work.

2. MODELING BUSINESS PROCESSES

The Workflow Reference Model [31] defines a business process as a set of one or more connected activities, which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional responsibilities and relationships. This definition extends the definition proposed by Davenport and Short [7] stating that a business process is a set of logically related tasks performed to achieve a defined business outcome. Most approaches to business process modeling concentrate on some sort of process map or diagram, which shows how activities are scheduled in the course of a business process. Indeed, there is little disagreement about the key elements process diagrams. There are usually ways to represent decision points and to express various activity coordination patterns, such as sequential flow, branching and parallel execution. Some techniques introduce swim-lanes to indicate the responsibilities of participants, such as departments or individuals. This allows representing the activities performed by actors in the context of a process.

Two representative coordination-oriented business process modeling techniques that make use of actors, activities and swim-lanes are Role Interaction Networks [24] and Role Activity Diagrams [21]. Role Activity Diagrams provide the means to identify roles and interactions. Roles organize a process' activities into sets of operations associated with a given participant in the process. Interactions show the dependencies between those participants. While this approach improves the understandability of a process model since it depicts what a participant does in a process, it falls short to explain the behavior of the business objects in a specific context of interaction. Additionally, roles are defined as groups of activities and not as types, so they cannot be explicitly composed or specialized.

Business process modeling is not limited to process diagrams. The focus of this paper is not on process diagrams but on describing the roles that are used to specify the responsibilities of business objects. A business object is the model of a concept in the business universe of discourse. It plays roles in a business process by means of participating in different activities. Business objects participate in different business processes in different contexts, thus playing multiple roles. It is important to note that process diagrams do not fully describe the business object structure and relationships, and do not emphasize why activities are performed or roles are enacted. Besides, they only identify actor roles, i.e. the roles of the performer of an activity. This means, for example, that the properties of a resource that is used by multiple activities are not separated according to its usage context. The next section introduces the fundamental concepts behind role theory and role modeling.

3. ROLE MODELING

In the late 1920s, role theory started to generated interest among social scientists from many backgrounds, such as psychology and sociology. Its central concern has been with patterns of human conduct, context and social structure as well as with individual response. The motivation for roles is to allow particular viewpoints regarding the factors presumed to be influential in governing behavior. It lies on a theatrical analogy of actors playing parts or roles in a play. As Biddle and Thomas [4] have stated: "When actors portray a character in a play, their performance is determined by the script, the director's instructions, the performances of fellow actors, and reactions of the audience as well as by the acting talents of the players. Apart from differences between actors in the interpretation of their parts, the performance of each actor is programmed by all of these external factors; consequently, there are significant similarities in the performances of actors taking the same part, no matter who the actual actors are."

There are many complementary definitions for the concept of role but still there is no consensus on the properties to represent it. In the late 1970s, sociological role theorists defined a role as "a comprehensive pattern for behavior and attitude" [26] or as "behavioral repertoire characteristic of a person or a position" [3]. Nonetheless, the concept of role is used in computer science and software engineering as a modeling technique that deals with separation of concerns, i.e. the separation of the behavioral repertoire characteristics of some concept. It is used in methodologies such as RM-ODP [14] and in several object-oriented frameworks [10, 12, 14, 15, 25].

3.1 Business Objects and Roles

Modeling is an abstraction technique that consists of identifying concepts of interest in some universe of discourse and representing its essential features for a specific purpose in a model. In business modeling, the universe of discourse corresponds to what is perceived of an organization as being reality by business domain experts.

Ontologies typically distinguish *entities* (nouns) from *activities* (verbs). Entities are things that exist in the business, either concrete (e.g. a person) or abstract (e.g. an organization). Activities are things that happen in the business. Activities make use of the business entities. We model both of these concepts as business objects. A *business object* is then the super type of all objects that represent business concepts with a well-defined boundary and identity. It encapsulates the definition, attributes, behavior and relationships to other business objects [20]. The state of a business object is characterized by the values of its attributes. The behavior is given by the actions that the business object is capable of performing to fulfill its purpose, including changing its intrinsic attributes and collaborating with other business objects.

Business objects have intrinsic and extrinsic features. Intrinsic features describe it in isolation, while extrinsic features arise from the relationships or collaborations with other business objects. For example, a Person has intrinsic features such as Age and Sex, and extrinsic features such as Job Position and Salary, which derive from a transitory relationship between the Person and some Organization or Company. Intrinsic features may change over time (e.g. Age) but always characterize the object. However, extrinsic features may become inappropriate (e.g. the Job Position property is not relevant when characterizing an unemployed person).

One way to separate the intrinsic features from the extrinsic features of an object is by means of roles [4, 15, 23]. Roles, as a modeling construct, aim at separating the concerns that arise from

business object collaborations. We define a *role* as the observable behavioral of a business object defined in a specific collaboration context. Thus, a role represents the extrinsic features of a business object when it collaborates with other business objects.

3.2 Identifying Roles

To distinguish roles from entities, Guarino *et al.* proposed two criteria [11]. A role is a type that (1) is founded and (2) lacks semantic rigidity. Something is *founded* if it is defined in terms of relationships with other things in a given context. For instance, the concept of Reader is founded since for a Person to be a Reader there must be something being read. Conversely, a Person is not founded for the reason that its intrinsic properties are defined on their own regardless of the collaborations with other things.

Something is *semantically rigid* if its identity directly depends on being kind of some class. A Book is semantically rigid since its identity is still that of a Book regardless someone is reading it or not. In contrast, Reader is not rigid because an entity filling the role of Reader retains its identity outside the context of that role. For example, a Person is a Reader while reading a Book, but when it stops reading it, it is still a Person.

Therefore, roles are founded, semantically non-rigid types while entities are non-founded, semantically rigid types.

4. ROLE-BASED PROCESS MODELING

The proposed approach deals with decomposing the business process modeling universe into two complementary models, the *business object model* and the *role model*, and later binding these two models into an integrated specification of the business process. The business object model deals with the structure and intrinsic properties of business objects. Here, a process is modeled as a network of business objects. However, business objects relate to other business objects in specific contexts and are often used in more than one context, where they may play different roles. So, the roles for a business object only need to be included in its definition when the object acts in the collaboration contexts described by the roles. It is also impossible to forecast all of the possible roles of a business object. Thus, adding superfluous roles to the object impairs several design quality attributes such as understandability, maintainability and reusability. To deal with such a concern, roles and business objects should be dealt with separately and later bound together.

The concept of role allows a system to be decomposed into a set of business objects capable of clearly separating core parts and collaboration-dependent parts and then to abstract and compose such objects. Consequently, a set of roles helps business objects to be defined to be more reusable and extensible. Roles may also be reused as an independent unit encapsulating specific collaborations. Roles are organized into role models, which deal with specifying the network of related roles required for a collaboration to happen.

We propose defining and represented both of these models using the Unified Modeling Language [19] since its graphical syntax and semantics is well-know by software specialists and, although to a lesser scale, by business specialists. However, the standard UML does not have explicit constructs to represent the required business domain concepts. We make use of the UML extensibility package to define such concepts. The extensibility package specifies how UML model elements can be extended and customized with new graphical representations and new semantics by specifying stereotypes, tagged values and constraints. A coherent set of such extensions defined for a specific purpose makes up a UML *profile* [2,

19]. The next subsections describe how the business object models and role models are represented.

4.1 The Business Object Model

The business object model specifies the structure and intrinsic properties of business objects. Business objects are coordinated towards the achievement of goals that describe why actions occur. A business process describes how objects are coordinated.

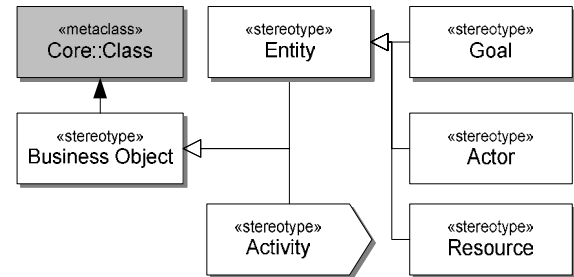


Figure 1. Classes in the business object model profile.

Figure 1 is a class diagram describing the UML stereotypes (classes in white) that are used in the business object model. A Business Object is a UML Class and it is specialized as a noun or verb by means of the Entity and Activity class stereotypes.

Business object models are represented as UML class diagrams and the intrinsic behavior of its objects is represented using UML's behavioral diagrams. Note that collaborations between business objects are not represented in this model but in the role model.

The stereotypes within the business object model can be summarized as follows:

- Business Object: an abstraction of a concept of interest in the organization. It is a UML Class.
- Activity: a specialization of Business Object. It is a verb describing how a piece of work is performed. Activities are performed by Actors, and operate over Business Objects, especially those acting as Resources.
- Entity: a specialization of Business Object. It is a noun describing a concrete or abstract business concept.
- Resource: a specialization of Entity, which is the input or output of an Activity. It represents things such as materials or information.
- Actor: a specialization of Entity. It is someone (a human actor) or something (an automated actor, such as an information system or a production machine) that can perform the actions required by an Activity.
- Goal: a specialization of Entity that represents a measurable state that the organization intends to achieve. Goals are achieved by Business Objects, especially Activities.

A business process is composed of Activities that use input Resources, such as materials or information, to produce output Resources. Nevertheless, the input of an Activity may be any other Business Object or a composition of Business Objects. For instance, changing or reengineering a business process is in itself a process. This process takes as input a business object model (i.e. a network of relationships between business objects) and produces a modified model. Therefore, the composed business object model is being used as a resource in this context.

Activities are performed to achieve specific business Goals. Analyzing Goals and their relationships with the Activities produces an alignment measure between the processes and the organization's operational strategy. The Activities of a business process are not

autonomous in the sense they require one or more Actor or Business Support Systems to perform them. Actors represent people, systems (mechanical or computer based) or a combination of both.

At a large scale, business processes are aggregated into *value chains* (which are also business processes) that produce a measurable value that is visible to external customers.

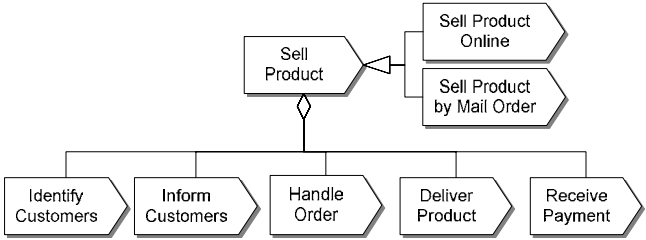


Figure 2. Example of activity composition and specialization.

Business objects are classes conforming to a type. They can be specialized and composed just like ordinary objects. Figure 2 shows an example of a class diagram depicting composition and specialization. Each chevron icon represents an activity or process as previously defined. The Sell Product activity is composed by a set of sub-activities such as Identify Customers and Handle Order. These activities can be further decomposed into actions that are more refined. The activity Sell Product is specialized as Sell by Mail Order and Sell Online. Note that composition and specialization do not imply any collaboration constraints between the activities.

4.2 The Role Model

Roles are a separation of concerns mechanism that allows business objects to be observed from different perspectives. Role models identify roles as types and describe the network of roles required for a specific collaboration to happen.

As a player of a collaboration, a role defines the set of extrinsic properties and behavior necessary to realize its participating collaborations.



Figure 3. Representation of a role model package (left). Pair or related roles (right).

Role models are represented as UML packages with two compartments (v. Figure 3, left). The bottom compartment of the role model is a standard UML activity or interaction diagram describing how the roles are orchestrated. The top compartment of the package depicts the roles within the role model. Roles are represented by rounded rectangles, connected by a navigable collaboration relationship between the roles. The representation of a role always shows its name. Optionally, it also depicts in parenthesis the name of the role model to where the role belongs so that its scope is clearly defined (v. Figure 3, right).

Figure 4 show an example of three role collaborations contained in two role models. The Tutorship role model defines a collaboration pattern between two roles, Tutor and Student. In this Course role model defines two pairs of collaborations: Participant/Taken Course and Lecturer/Given Course.



Figure 4. Example of role collaborations.

Roles are modeled as classes and represented in class diagrams. Methods and attributes concerning the specific collaboration context can be specified in the class diagram. Roles can also be constrained. A constraint asserts conditions between the roles in a role model. It can be expressed informally or formally (e.g. in plain text or OCL). An example of a constraint is disallowing two roles to be played simultaneously by the same player, such as forbidding an object playing the role of Tutor and that of Student simultaneously and in the same context.

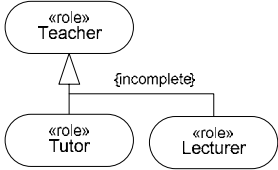


Figure 5. Example of role specialization.

Figure 5 is a class diagram that shows how the Teacher role is specialized as Tutor and Lecturer. Role specialization means that if a business object is able to play a child role, then it is also able to play the super role. We have not yet found the need to define abstract roles, i.e., a role that may only have its non-abstract specializations instantiated.

4.3 Binding Roles to Objects

Roles are bound to business objects pertaining to a given business object model. The binding is accomplished via the «play» relationship stereotype, which links a business object to a role. It means the business object is able to exhibit or play the behavior specified by the target role.

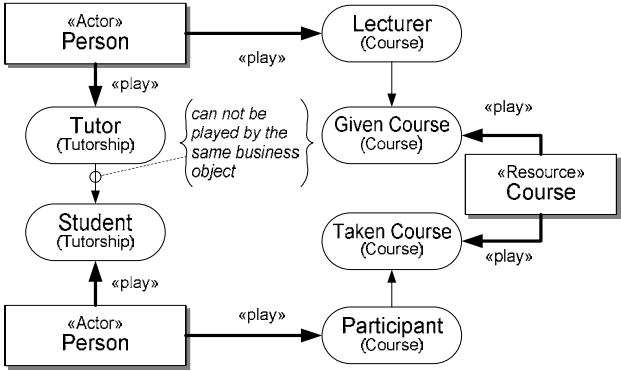


Figure 6. Binding roles to business objects.

Figure 6 shows a class diagram where the pairs of roles Tutor/Student, Lecturer/Given Course and Participant/Taken Course defined earlier in Figure 4 and Figure 5, are bound to two different business objects, Person and Course. The binding between objects and roles is depicted as a strong arrow. The light arrow represents the relationships between roles. The model also defines a constraint in the Tutorship role model. It asserts that the instances actually playing the Tutor and Student role must be distinct. In this example, it means the Person acting as a Tutor and the Person acting as Student must be different objects, as expected.

5. EXAMPLE

Figure 7 shows two base role models, Supply and Pay and a composed role model, Purchase. Each role is a class and has methods and attributes concerning the specific collaboration context (e.g. the Supplier role in the Supply role model has the inquire and order methods). The Purchase role model describes the collaborations between a Client and a Supplier, while the Pay role model, specifies Payer and Payee.

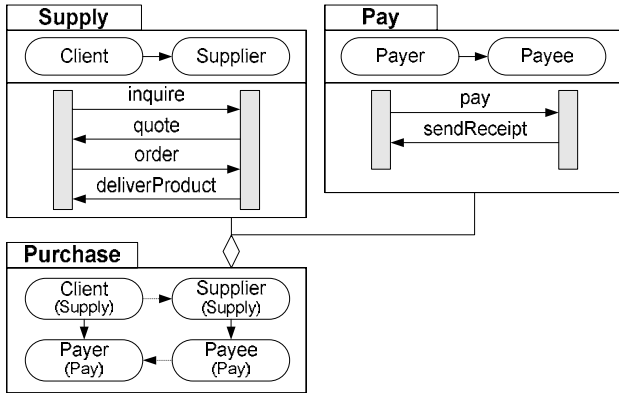


Figure 7. Supply, Pay and Purchase role models. Purchase is composed of the role model Supply and role model Pay.

The Purchase role model is a composition of the Supply and Pay role models. A purchase results from supplying a product and paying for it. Figure 8 shows the binding of the roles within the Purchase role model to a set of business objects. In the first case, a Retailer acts as the Client and Payer to a Producer who is a Supplier and a Payee to the Retailer. However, the Retailer also acts as a Supplier (and a Payee) to a customer.

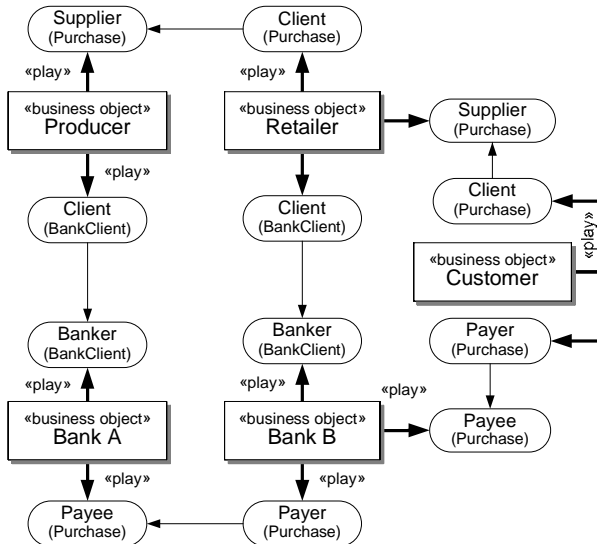


Figure 8. Binding roles to business objects.

6. CONCLUSIONS

This paper has presented the fundamental concepts towards a conceptual object-oriented framework for role-based business process modeling. It relies on defining two distinct models. The business object model focus on describing the components of a business process (activities, goals, resources and actors) as busi-

ness objects. This model depicts the type of each business object, its intrinsic behavior and properties but does not address the representation of the object's features that are related to its collaborations with other objects.

The role model depicts the collaborative behavior between roles and the constraints that regulate them. Roles are bound to business objects in a specific business object model, thus defining their usage context. This model describes roles as types on their own that can be specialized and aggregated. Role reuse is possible whenever the semantics of the interaction pattern is the same, regardless of the interaction context.

The proposed approach separates the specification of the intrinsic features of a business object from its extrinsic features, meaning that the properties and behavior that arise from the collaborations with other objects are separated from the properties concerning the object. This separation results in an increase of the understandability of the business process since each different aspect of the business object may be discussed, analyzed and dealt with separately.

Additionally roles also contribute to keep the alignment between the multiple organizational levels where a business process is defined. When a business object specified at business level is mapped to a component at business process support systems level, roles provide information on how to design the component so that changes to other levels can be traced and managed. Since the collaborative aspects of a business object are specified outside the object as roles, changes to a business process only interfere with the roles which derive from the corresponding activities, leaving the intrinsic properties of the object and its remaining roles unchanged. This means that only the implementation of the concerned roles needs modifications. The same reasoning applies the other way around. When the implementation of a specific role or business object is changed due to technical modifications or to the evolution of the software, these changes can be traced up to the processes and goals depending on it.

The value of using role modeling increases with the need of making explicit the patterns of interaction between business objects. This is the case of processes where its business objects relate to several other business objects. In this case, understanding and reengineering such a process is often difficult due to the number of dependencies between objects, which are not separated or organized according to the interaction context. This also makes difficult to abstract common behavior patterns so that the business process elements may be reused in other contexts.

We are currently extending this framework to enhance the representation of the interaction between business objects and the corresponding business support systems. The goal is to analyze the gap between the existing human skills and information system services of an organization and the requirements imposed by the as-is and to-be business models so that the alignment between these two levels may be improved.

7. REFERENCES

1. W.Aalst, K.Hee, *Workflow Management*, MIT Press, 2002.
2. S.Alhir, *Unified Modeling Language Extension Mechanisms, Distributed Computing*, 1998.
3. C. Bachman, M. Daya. *The role concept in data models*, Proceedings of the 3rd International Conference on VLDB, 1977.
4. B. Biddle, E. Thomas, *Role Theory, Concepts and Research*, Kluwer Publishers, 1979.

5. Y. Chan, *Why Haven't We Mastered Alignment?: The Importance of the Informal Organization Structure*, MISQ Executive, Vol.1, No.2, 2002.
6. B. Curtis, M. Kelner, J. Over, *Process Modeling*, *Communications of the ACM*, Vol. 35, No. 9, 1992.
7. T. Davenport, J. Short, *The New Industrial Engineering: Information Technology and Business Process Redesign*. Sloan Management Review, 1990.
8. H. Eertink, W. Janssen, P. Luttighuis, W. Teeuw, C. Vissers, *A Business Process Design Language*, World Congress on Formal Methods, Springer, 1999, pp. 76-95.
9. H. Eriksson, M. Penker, *Business Modeling with UML*, OMG Press, 2001.
10. G. Gottlob, M. Schrefl, B. Röck, *Extending Object-Oriented Systems with Roles*, *ACM Transactions on Information Systems*, Vol, 14, 1996 pp. 268-296.
11. N. Guarino, M. Carrara, and P. Giaretta. *An ontology of meta-level categories*. Proceedings of the Fourth International Conference on Knowledge Representation and Reasoning, pages 270–280. Morgan Kaufmann, 1994.
12. T. Halpin, *Augmenting UML with Fact-orientation*, *34th Hawaii International Conference on System Sciences*, IEEE Press, Hawaii, USA, 2001.
13. ISO, *ISO/IEC 10746 ODP Reference Model*, International Standards Organization, 1995.
14. E. Kendall, *Agent Roles and Role Models, New Abstractions for Multiagent System Analysis and Design*, International Workshop on Intelligent Agents, 1998.
15. B. Kristiansen, *Object-Oriented Modeling with Roles*, 1st Conference on Object Information Systems, 1996.
16. M. Madhavji, *The Process Cycle*, *Software Engineering Journal*, Vol. 6, No. 5, 1991.
17. C. McGowan, L. Bohmer, *Model-based business process improvement*, 15th International Conference on Software Engineering, IEEE Computer Society Press, 1993.
18. D. Miers, *Business Process Engineering*, C-T Colin, Kogan Page, London, 1996.
19. OMG, *Unified Modeling Language Specification*, Version 1.5, formal/03-03-01, 2003.
20. OMG, *Business Object Management Special Interest Group (BOMSIG) Glossary of Terms*, 1995.
21. M. Ould, *Business Processes, Modeling and Analysis for Reengineering and Improvement*, John Wiley & Sons, 1995.
22. A. Scheer, *ARIS – Business Process Modeling*, 2nd edition, Springer, 1999.
23. T. Reenskaug et al., *Working With Objects: The OOram Software Engineering Method*. ManningPublication Co., 1996.
24. B. Singh, G. Rein. *Role Interaction Nets (RINs): A Process Description Formalism*, MCC, 1992
25. D. Taylor, *Business Engineering with Object Technology*, John Wiley & Sons, 1995.
26. R. Turner, *Strategy for Developing an Integrated Role Theory*. *Humboldt Journal of Sociology and Religion* 7: 123-139, 1979.
27. M. Uschold, M. King, S. Moralee, Y. Zorgios, *The Enterprise Ontology*, *The Knowledge Engineering Review*, Vol. 13, 1998.
28. E. Verharen, *A Language-Action Perspective on the Design of Cooperative Information Agents*, CIP-Gegevens Koninklijke Bibliotheek, 1997.
29. T. Walford, *Business Process Implementation for IT Professionals and Managers*, Arthech House, MA, 1999.
30. E. Yourdon, *Modern Structured Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
31. Workflow Management Coalition, *The Workflow Reference Model*, 1995