Activity Systems and Context Working as Core Concepts in Modeling Socio-Technical Systems

Heidrun Allert, Christoph Richter

Research Group Knowledge Media, Upper Austria University of Applied Sciences,
Softwarepark 11, 4232 Hagenberg, Austria.
{heidrun.allert;christoph.richter}@fh-hagenberg.at

Abstract. Current modeling approaches in the field of learning and work resemble the notion of workflows and hence fall short in describing the situated and socially mediated nature of practices. Against this background the paper describes an alternative modeling approach as well as its theoretical foundation and practical implications. As this paper rethinks the epistemological foundation of modeling socio-technical systems, the approach goes beyond specifying specific concepts and relations and addresses the meta-level of modeling. The cultural historical activity theory as well as the theory of social systems work as the theoretical foundation.

Keywords: Knowledge representation, activity theory, socio-technical systems

1 Introduction

The formal description of socio-technical systems as well as learning processes has attracted attention among researchers and developers in recent years and has resulted in a couple of specifications focusing on individual as well as collaborative processes of learning and working. The explicit and formal representation of such processes is relevant for diverse reasons. Besides their technical and economic relevance they also provide an important communicative tool for designers as they allow to share experiences and to coordinate activities among those involved in the design and development process. Furthermore they are of interest for scientists as they provide a frame of reference for the analysis and comparison of different scenarios. While current modeling languages such as e.g. IMS Learning Design [8] overcome the problem of de-contextualized objects by describing the use of these objects within a unit of study, they resemble traditional workflow models and hence reproduce the problem of contextualization on a higher level, as the unit of study is again de-contextualized. Even though these approaches acknowledge the complex nature of situated processes they are reductive in the sense that they equal the processes with the sum of the activities entailed. Thereby the situated and socially mediated character of human action is neglected. Against this background this paper outlines an alternative modeling approach which draws on activity-theoretical as well as systemic theories to depict practices. The formal concept of role types is used to represent the systemic nature of activity and its situatedness adequately. The paper is structured as follows: Key assumptions of the cultural-historical activity theory as well as the theory of social systems are introduced to outline the underlying rationale of the modeling approach. Referring to the theoretical foundation the modeling approach is developed step by step. Finally the practical implications of the modeling approach are discussed.

2 The Concept of Practice

The concept of practice can be defined as *the ways of doing work, grounded in tradition and shared by a group of workers* [4]. In general it has to be distinguished between practices as implemented by a specific group of people (e.g. the way a particular lecture is given at a particular university) and practices as prototypical conceptualizations of a certain activity within a broader community (e.g. the way lectures are given usually). While the concept of practice can basically be defined as a customary way of doing things, it seems worthwhile to have a closer look at this concept from a theoretical point of view. Theories this work is founded in are the activity theory and the theory of social systems. The following is a list of key-assumption on human activity and social systems, which is based on activity-theoretical and system-theoretical (systemic) considerations. A more detailed and extended outline of these key assumptions is given in [2].

2.1 Key Assumptions of Activity Theory

Activity theory is a powerful philosophical framework and descriptive tool focusing on understanding human activity and work practices. It is based upon the anthropological and psychological theory of Leontjew [11] and Vygotsky [19].

(1) Human activity is object-oriented, i.e. it is directed towards a physical or conceptual object that is manipulated or transformed by the activity. It is the *object of activity* and not the goal that allows distinguishing different activities from one another. (2) Activities are always mediated by *tools* and signs, which are constitutive elements of the activity. Tools and signs are mediators which range from material tools over less tangible artifacts like plans and spreadsheets to scientific theories and languages. Tools capture and preserve the socially shared knowledge developed in a given community and mediate the subjects' relation with the object of the activity as well as with other human beings [11], [16]. (3) Activities are shaped by contextual conditions and circumstances. The *subject* has to continuously adapt its actions and operations to external events and circumstances. As a consequence human activity is guided but not predefined and determined by the plans of those engaged in the activity [3]. The variability of contextual conditions and circumstances inevitably results in a variation of the way the activity is carried out and can result in the evolution of the activity system if improper variations are selected and proper variations stabilized [14]. (4) The relationship between subjects, objects, and tools is reciprocal. These elements are mutually interdependent, which means that a change in one of them will inevitably alter the other ones. In this sense the constituents of an activity form a system where each component is defined in relation to the other

components. (5) Activities are hierarchically structured. According to [11] three levels of activities can be distinguished, namely collective *activities* which are carried out on a communal level often involving multiple actors, *actions* that are performed by a single subject to achieve a certain goal relevant to the collective activity, and *operations* in the form of fine grained automated routines. But even though activities are structured hierarchically the relation between operations and actions as well as actions and activities is not an additive one [11]. Therefore it is not possible to simply decompose an activity into a set of actions. (6) Activities are never static but evolve when contradictions or tensions emerge between the elements in an activity system. Human activity whether carried out by an individual or collectively cannot be detached from its social context as its meaning is bound to its interpretation within a collective.

2.2 Key Assumptions of the Theory of Social Systems

The Theory of Social Systems is a descriptive framework presenting a system-centered view. It is a non-prescriptive meta-theory. [20] characterizes this theory as universal regarding domains and disciplines as many disciplines are confronted with similar problems, e.g. the problem of increasing complexity, which can not be reduced to simple categories and principles. A comprehensive introduction into the Theory of Social Systems is given by [20] and [9]; the foundational work is Social Systems [10]. Here only few key assumptions are presented.

(1) Personal systems as well as social systems are meaning processing systems as they process information by constructing meaning. A social system is not equivalent with the group of people in the system, but it is of different quality. Personal systems and social systems reduce environmental complexity by processing environmental complexity selectively. Thus, inner and outer complexity is different. Systems organize their inner complexity and reduce contextual (environmental) complexity. (2) The Theory of Social Systems is a descriptive framework which describes the world in terms of systems, drawing the difference system/environment, whereas in object-oriented modeling objects and categories are defined. *The central paradigm of recent system theory is 'system and environment'. The concepts of function and functional analysis no longer refer to 'the system' (...) but to the relationship between system and environment. (…) This leads to a radical de-ontologizing of objects as such (...)* [10]. The difference system/environment is not an ontological but an epistemological. (3) Systems are closed and self-regulated. Within a system, elements generate each other reciprocally, e.g. in listening, the audience creates the speaker and vice versa. An entity, such as a person (personal system) does not belong to a social system but to its environment [10]. This means, a person (and any other entity/type) does not belong to a system for all intents and purposes but in some respect, filling a specific role [20]. For example: The person Peter and the person John belong to the environment of the system family. Only Peter filling the role son and John filling the role father, belong to the system. A system can not determine another, e.g. a personal system can not determine a social system.

3 Modeling Practices as Coherent Social Systems

This section outlines a modeling approach for modeling socio-technical systems. The approach draws on the concept of practice and refers to activity systems as coherent social systems. As this paper rethinks the epistemological foundation of modeling socio-technical systems, the approach goes beyond specifying specific concepts and relations and addresses the meta-level of modeling. The (meta-) modeling approach is based on three major inputs: (1) distinguishing the meta-level categories `natural type` from `role type` to distinguish between an object and its role within a specific context [7], [17], (2) introducing a system-centered perspective to model a system of elements which reciprocally generate each other, and (3) integrating basic assumptions of activity theory in order to overcome shortcomings of workflow models which work as means-end-models. The structure of this section is as follows: First a basic comprehension of the formal terms natural type and role type is given, based on the work [6], [7], and [17]. Then, a system-centered (systemic) perspective is delineated, modeling socio-technical systems as coherent social systems [1]. Finally the meta-model of the system-centered role-based modeling approach, which reflects an activity-theoretical and system-centered perspective, is presented.

3.1 Meta-Level Categories Natural Type and Role Type

In the context of knowledge representation, meta-level categories are categories used to model the world, such as `concept`, `property`, `state`, `role`, `attribute`, and `relation`. Within this work, distinguishing the meta-level category `natural type` from `role type` is crucial, based on [7] who provides an ontological distinction. This distinction is based on the meta-properties `identity` and `rigidity` (table 1). [18] states that the definition of natural types matches the class construct of object-oriented modeling, as the definition of classes is outside the context of any relationships, and the instances keep their types for their lifetimes (identity). A type is a natural type if *belonging to the type is independent of being engaged in a relationship (except for, perhaps, a whole-part relation) and if an object cannot leave the extension of the type without loosing its identity*. [18]. Natural types relevant in modeling socio-technical systems and (knowledge) practices are e.g. `person` and `artifact` (such as `technology, services, information asset)`. Person here represents any meaning processing system (e.g. a group, an organization; according to [10]). The category of role type is as fundamental in object-oriented modeling as the category of natural types, classes, and relations [17]. Due to the fact that usually no difference is made between the concepts of natural types and role types, the concept of role types is relatively unknown. The concept of types normally represents both: natural types and role types. Due to a synopsis prepared by [17] the concept of role type does not play a role in most formal languages, including the logics (cp. modeling and the formal grounding of maths by Frege, 1848-1925), while it plays a major role in linguistics and semantics [5]: In linguistics there is a common theory of formal languages, integrating the role type as fundamental concept complementing the concepts of predicates and objects.

The difference between role types and natural types is in its contents. Syntax allows to work without distinguishing the concepts - but, semantically many problems arise from not drawing the difference between the concepts [17]. Husserl introduces the quality of Fundierung (en: foundation), cp. [17]. [6] (in the context of knowledge representation) specifies semantical and ontological rigidity. A concept is founded if none of its instances can exist alone: Each instance is related to another instance. A concept is semantically or ontologically rigid if an instance can not join and leave the extension of the concept without loosing its identity. If x *has the property of being an apple, it cannot lose this property without losing its identity (...). This observation goes back to Aristotelian essentialism (...).* [7]. [6] founds the concept of role type as an ontological concept and gives a formal definition assigning two conditions. Role types are those concepts which are founded and not semantically rigid. Natural types are those concepts which are semantically rigid and not founded. According to [7], the meta-property rigidity means: *A property P is rigid if, for each x, if P(x) is true in one possible world, then it is also true in all possible worlds. Person and location are rigid, while student and tall are not.* A role type specifies the behavior within a context - a behavior is a contract or relationship between two entities. A role type implies a specific relationship between instances filling the role. Role types require the instance to have an identity apart from its role type. Natural types do not imply a specific relationship with other types (except for whole-part relations). Natural types grant an instance its identity. The concept of role types allows describing the function an object fills within a specific context. [17] states that the standardization of the term *role* (role type) in modeling complements the meta-level categories type and relation. Instances of types can *fill roles*. The classical dichotomy type/relation is extended to the trilogy type/role/relation. [17] works out practical implications for its integration in object-oriented modeling and its representation in the modeling language UML ([13]). Introducing the concept of role types into object-oriented modeling makes possible dynamic modeling approaches. Role types are dependent from relations and context. Each instance of a certain natural type can fill different role types, called polymorphism [17]. Role types and natural types (in the context of object-oriented modeling [17] refers to natural types as classes) are interconnected by the `supports` relationship, specifying which classes support which roles [17]. The role type specifies the behavior instances of a natural type must provide in order to be able to fill the role. How the behavior is achieved is left up to the classes that support it. It depends on the classes' properties and qualities whether its instances can fill a role or not.

Table 1. Distinguishing the ontological concepts natural type from role type.

Natural Type	Role Type
Static	Dynamic
Semantically rigid: An instance of a class once and forever belongs to that class. It cannot change it without loosing its identity.	Not semantically rigid. Instances of natural types can fill, adopt and leave a role without loosing their identity [6].
Not founded	Founded. Role types are defined by context and relation.

Integrating the concept of role types in UML, the notation for role types must be specified. [17] recommends using the lollipop-notation, which in UML represents interfaces. In the following UML diagrams a rectangle indicates a natural type, a circle indicates a role type (fig. 2). The UML diagram specifies role types the instance of a natural type can fill. In specifying metadata, it is necessary to distinguish between static attributes (such as Dublin Core and vCard attributes), which are based on the natural type of a resource, and context- or role-dependent attributes which are based on the role type a resource fills. Natural types such as information assets and persons have context-independent static attributes. These static attributes are independent from the role a resource fills. Regarding an information asset, static attributes are taken from Dublin Core (Dublin Core Metadata Initiative, 2004). Persons are annotated with vCard attributes like vcard:FN (full name) and vcard:EMAIL. Besides static attributes, context-specific role-based attributes are attached to resources. Role-based attributes are specified according to a specific context.

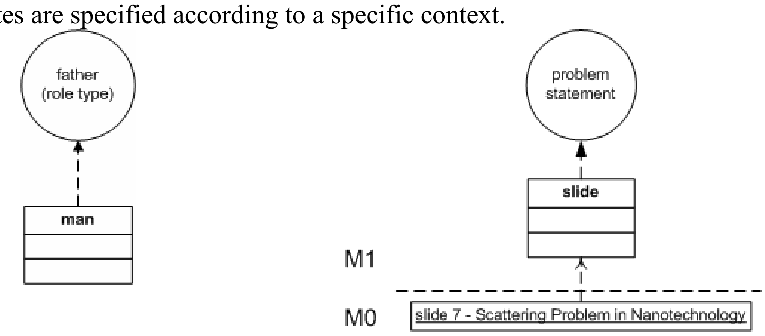

Fig. 2. Notation in UML. Man is a natural type, father is a role type. The role father can be adopted and dropped by instances of the class man. An instance of the natural type `slide` fills the role type `problem statement`. Not distinguishing the natural type from the role type mixes the resource and its function/role within a specific context (as e.g. in LOM's category Learning Resource Type).

The System-Centered Perspective - Modeling Coherent Social Systems

[7], [17] define the meta-level category role type as a binary relation. A role type is defined by its relation to another role type. In this section we complement this definition by a system-centered perspective and define the concept of role type as n-ary relation. We refer to role types as `role` and to natural types as `type`.

Roles within a system are related. They generate each other, as elements within an activity system generate each other reciprocally. For person related roles this means for example: there is no accused without a complainant, no father without a son or daughter. A person (natural type) filling a role within a system has expectations towards the other persons filling roles. The accused has specific expectations towards the judge. An instance of a natural type fills a role as soon as it moves into the system. In case of the natural type person, the concept of role types is intuitively understood (fig. 3). But also further natural types such as information asset (e.g. a picture), behavior, technology, service, etc. fill roles within diverse systems (fig. 4). In the same way activities are related and generated by one another within the system. Within the legal system (which serves as an example here) the type picture does not exist. But a picture which fills the role indication does exist in the legal system. This means: as soon as someone hands in a picture the judge will bring it into the system as indication (the picture filling the role indication) - or refuses to do so. Only filling the role indication (or another) the picture is part of the system. The same with the role evidence: only as the judge accepts an asset as evidence it becomes part of the system. It is not part of the system per se, but filling the role evidence. Types do not belong to a system but to its environment [10]. An instance of the natural type person which fills the role accused in the legal system fills the role father in the system family, each with specific intents, aspects, and purposes. Modeling coherent social systems, this work argues that roles are arbitrary n-ary relations. This is different from the definition of a role as a binary relation [7]. Whereas [7] interprets an unary predicate as a concept (class) and a binary predicate as a role, this work assumes a role as n-ary predicate. The relation `father-son` is insufficiently described by a binary relation as in a system the relation father-son is entirely affected by any other role represented in the system e.g. the role mother. The absence of an instance filling the role mother entirely affects the relation father–son.

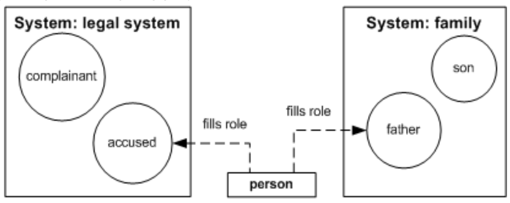

Fig. 3. A person (natural types) filling roles within different systems.

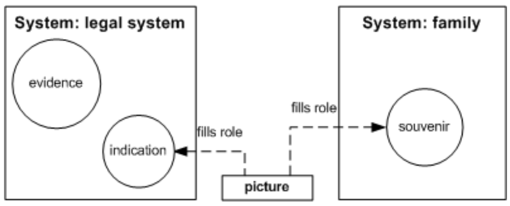

Fig. 4. A picture (natural types) filling roles within different systems.

Modeling an activity system requires a further level of abstraction as it has a (theoretical) foundation and underlying rationale (e.g. the concept of *Bildung* or the concept of *situated learning*). This foundation conceptualizes the system but is not formalized. The underlying rationale of the system is reflected by the *meta-type* in M2 (meta-level 2, fig. 5). The meta-meta-level category meta-type is crucial in modeling socio technical systems, as there always is an underlying rationale which is not formalized. The meta-type reflects central issues/culture/identity of the activity system.

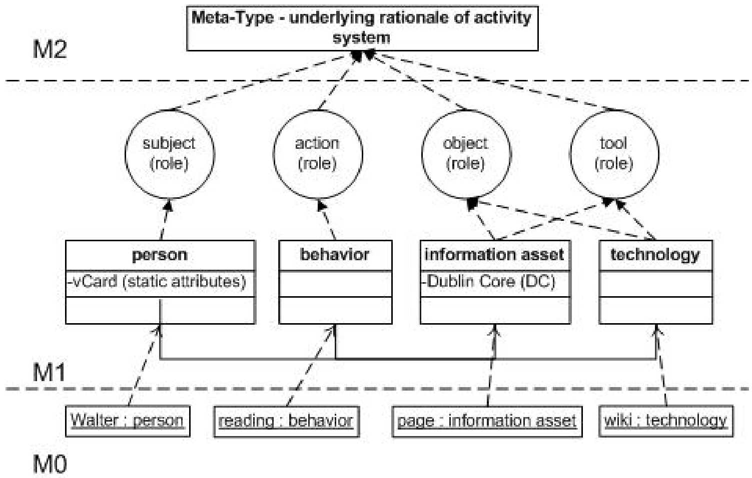

Fig. 5. A set of natural types (person, behavior, information asset, technology), a set of role types (subject, action, object, tool (M1)) and a meta-type (M2) is specified.

The role-based modeling approach allows modeling a natural type filling different roles within different activity systems. Tim, an instance of the natural type `person`, fills the role `manager` at workplace and the role `learner` within the executive MBA program. Interoperability between the activity systems and different contexts is given via the natural type. First, a set of role types and natural types is specified. Then, the roles are related. Typically the natural types filling roles are related in a cause-and-effect chain, forming a process-oriented workflow model. In the context of learning this means, a subject performs an activity to reach a predefined goal (cp. e.g. [8]. Learning is assumed to result from a chain of actions. Such a model would oversimplify learning for several reasons, cp. [14]. To avoid this, this work models `action` as an n-ary relation and `action` and `activity` are modeled on different levels of emergence (fig. 6). Thus the modeling approach addresses several key assumptions of Activity Theory and the Theory of Social Systems: (1) learning is contextualized, (2) activities can not be de-composed to several actions without loss of information - the relation between operations and actions as well as actions and activities is not an additive one, (3) the elements of a system generate each other, (4) learning can not be reduced to a chain of actions - it is not possible to simply decompose learning into a set of actions, (5) social systems are meaning processing systems - the difference between a social system and a group of persons is not a quantitative but a qualitative one. A final example might illustrate this: Taking into account Leontjew's [11] concept of activities, actions and operations, one and the same action is capable to be a component of different activities. An activity can not be decomposed to the actions it contains without loosing information. The action of reading is different depending on the activity the learner carries out (reading a problem statement in a setting of knowledge creation learning, or reading out loud in a setting of instructional design, e.g.).

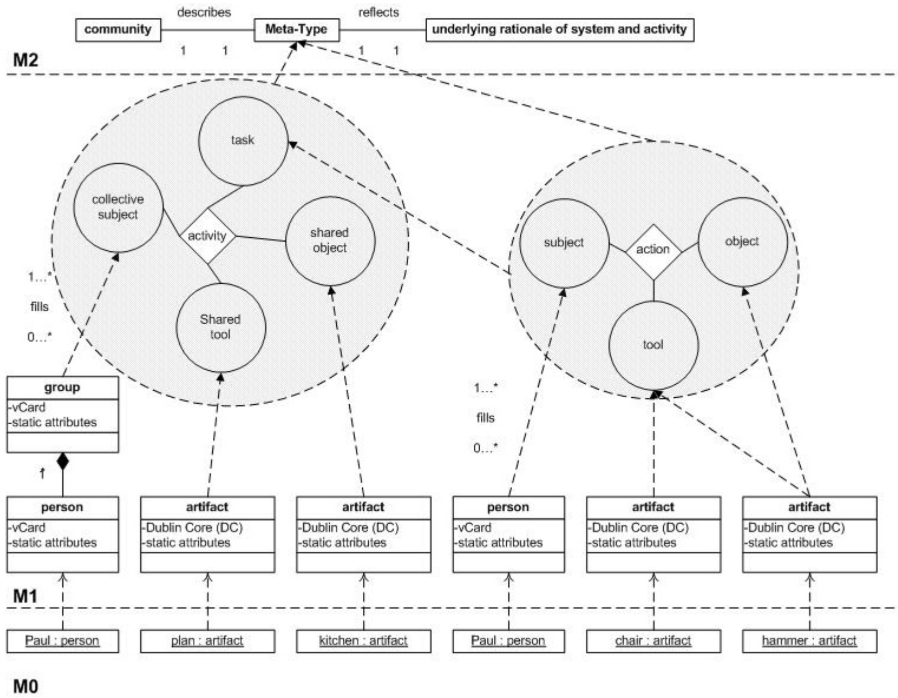

Fig. 6. The meta-model of a system-centered role-based modeling approach. According to activity theory and the theory of social systems, the elements within a system generate each other. Changing one element within the system also effects all the others.

4 Practical Relevance

The aim of this section is to elaborate on the practical implications of the meta-model introduced before. Thereby it is important to note that the model described so far is a meta-model providing the semantics but not the syntax of a respective modeling language. The aim is to demonstrate the general implications of the meta-model proposed. In order to do so, several concrete modeling problems related to learning and education will be discussed.

4.1 Decomposition of activities

The problem of decomposition of activities directly relates to the relationship between activities and actions. The question at stake is whether an activity can be broken down into a set of interrelated actions without loss of information. Common modeling languages such as the Unified Modeling Language [13], the Business Process Modeling Notation [12] and IMS Learning Design [8] are build on the assumption that such a decomposition is possible and hence equate the sequence of actions with the respective activity. Nevertheless this is problematic not only from a theoretical but also form a practical point of view. Given that the assumption would be true it would follow that the sequence of actions including the actors, artifacts and tools used would suffice to describe an activity. Consequently it would be possible to compare two activities by comparing the actions entailed. For example the IMS-LD Best Practice Guide [8] describes a problem based learning scenario as an arrangement of 17 actions, implying that differences between pedagogical scenarios are due to the organization of actions entailed. From this point of view it would not make a difference whether the students would have to solve a well- or ill-structured problem, whether it is a theoretical or practical problem, or whether there is a real customer interested in the outcome or not. All these differences cannot be modeled adequately and therefore result in misleading or even wrong comparisons across pedagogical scenarios in particular and practices in general. On the other hand, the meta-model introduced here can handle these differences as it models activities as entities of their own, which cannot be decomposed.

4.2 Equivalence of actions and its components

Another problem relates to the comparability of actions and its components, i.e. the question whether two actions are equivalent or not. From the practical point of view this question is of interest with regard to the modification of a pedagogical design or the implementation of a given design in another context. Both modification and re-implementation require knowing the constituting elements of the original solution in order to modify or transfer them intentionally. Modeling language that do not distinguish between role- and type-based attributes of the objects involved, such as the ones mentioned above, run into trouble when it comes to the equivalence of actions and its components. The problem is that they either generalize to natural-classes or that they mix up role- and type-based attributes. In both cases the misleading conclusions might be implied. For example in a given pedagogical design the students might be administered a multiple-choice test in order to assess their understanding of the topics addressed in the course. The aim of the multiple-choice test in this case is to provide an overall feedback whether the students understood the core concepts or if remedial activities are required. When modifying or adapting the pedagogical design it might become relevant to replace the multiple-choice test by another instrument and hence to know what is equivalent to this test. In case one generalizes to the natural-class the static attributes of the test, namely that it is a multiple-choice test focused on domain specific topics comes to the fore while its particular purpose is dropped. Consequently the test might be replaced by another test which is not designed to provide formative but summative information on students' performance. Even if no such generalization is made, it still remains unclear which attributes of the test are relevant and which not and hence might lead to the false conclusion that it is important to use a multiple-choice test while in fact any other instrument providing feedback on students understanding would be suitable.

Coupling of actions

The last problem to be discussed here relates to the modeling of interrelations between actions. Following the idea of hierarchical decomposition most of the current modeling languages treat activities and actions as self-contained entities related to other activities and actions via respective pre- and post-conditions. Consequently activities and actions are either organized sequentially or in parallel, while in the later case no direct dependency exists between the actions while being carried out. While these approach allows to depict the overall flow of actions and activities it ignores the fact that actions or activities are often coupled via the persons involved and the artifacts used. In other cases two actions might have to be coupled in order to work correctly. Even though giving a lecture and listening to the lecturer can be decomposed into two distinct actions the coupling of these actions is essential for the overall outcome. The mutual dependency of synchronous actions or activities is of vital importance for understanding the mode of operation. While these dependencies cannot be described adequately when action and activities are treated as self-contained entities, the meta-model introduced here overcomes this problem by allowing a person or artifact to fill different roles in the context of different actions and hence to couple them explicitly.

Further work is to be done in specifying a modeling language which is based on the meta-model presented in this paper. Modeling languages are needed to describe practices and socio technical systems in the field of computer-supported collaborative learning and computer-supported cooperative work.

Acknowledgments. The KP-Lab Integrated Project is sponsored under the 6th EU Framework Program for Research and Development. The authors are solely responsible for the content of this article. It does not represent the opinion of the KP-Lab consortium or the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

References

1. Allert, H.: Modeling Coherent Social Systems for Learning. Universität Hannover, Hannover (2005)
2. Allert, H., Richter, C.: Drawing on Practices for Modeling Socio-Technical Systems. Proceedings of the Computer Supported Learning Conference (CSCL'07). Kluwer Academic Publishers, Lancaster (2007) in press
3. Bardram, J. E.: Plans as Situated Action: an Activity Theory Approach to Workflow Systems. Proceedings of the 5th European Conference on Computer Supported Cooperative Work (ECSCW'97). Kluwer Academic Publishers, Lancaster (1997) 1732-1740
4. Bødker, S.: Activity Theory as a Challenge to Systems Design. In: H.E. Nissen, H. Klein, R. Hirschheim (eds.). Information Systems Research: Contemporary Approaches and Emergent Traditions. Elsevier, Amsterdam (1991) 551-564
5. Bühler, K.: Sprachtheorie - die Darstellungsform der Sprache. Gustav Fischer, Jena (1934)

6. Guarino, N.: Concepts, Attributes and Arbitrary Relations: Some linguistic and ontological criteria for structuring knowledge bases. Data & Knowledge Engineering, 8 (1992) 249-261
7. Guarino, N., Carrara, M., Giaretta, P.: An ontology of meta-level categories. In: D. J. E. Sandewall, P. Torasso (eds.): Principles of Knowledge Representation and reasoning: Proceedings of the fourth international conference. Morgan Kaufmann, San Mateo (1994) 270-280.
8. IMS Global Learning Consortium: IMS Learning Design Specification, Version 1.0 (2003), http://www.imsglobal. org/learningdesign/index.cfm.
9. Krieger, D.: Einführung in die allgemeine Systemtheorie. UTB, München (1998)
10. Luhmann, N.: Social Systems. Stanford University press, Stanford (1995)
11. Leontjew, A. N.: Activity, Consciousness, and Personality. Prentice Hall, Englewood Cliffs (1978)
12. Object Management Group: Business Process Modeling Notation (BPMN), Version 1.0 (2006), http://www.omg.org/cgi-bin/doc?dtc/2006-02-01
13. Object Management Group: Unified Modeling Language (UML), Version 2.0 (2005), http://www.omg.org/ technology/documents/formal/uml.htm
14. Scheunpflug, A.: Evolutionäre Didaktik: Unterricht aus system- und evolutions-theoretischer Perspektive. Beltz, Weinheim (2001)
15. Sowa, J.: Knowledge representation - logical, philosophical and computational foundations. Brooks/Cole, Pacific Grove (2000)
16. Stahl, G.: Meaning and Interpretation in Collaboration. In: B. Wasson, S. Ludvigsen, U. Hoppe (eds.): Designing for Change in Networked Environments. Kluwer, Dordrecht (2003) 523-532
17. Steimann, F.: Modellierung mit Rollen. Universität Hannover, Hannover (2000)
18. Steimann, F., Siberski, W., & Kühne, T.: Towards the Systematic use of Interfaces in JAVA Programming. Proceedings of the 2nd int. conf. on the principles and practices of programming in java PPPJ 2003, ACM, Kilkenny City (2003) 13-17
19. Vygotsky, L. S.: Mind in Society. Harvard University Press, Cambridge (1978)
20. Willke, H.: Systemtheorie. Fischer, Suttgart (1993)