Formalization of REA Ontology

Johannes Buder* Claudia Koschtial†
Carsten Felden‡

*TU-Bergakademie Freiberg, johannes.buder@bwl.tu-freiberg.de
†TU-Bergakademie Freiberg, claudia.koschtial@bwl.tu-freiberg.de
‡TU-Bergakademie Freiberg, carsten.felden@bwl.tu-freiberg.de
This paper is posted at AIS Electronic Library (AISeL).
http://aisel.aisnet.org/amcis2009/431
Formalization of REA Ontology

Johannes Buder
TU-Bergakademie Freiberg
Johannes.Buder@bwl.tu-freiberg.de

Claudia Koschtial
TU-Bergakademie Freiberg
Claudia.Koschtial@bwl.tu-freiberg.de

Carsten Felden
TU-Bergakademie Freiberg
Carsten.Felden@bwl.tu-freiberg.de

ABSTRACT

Current Enterprise Resource Planning (ERP) systems are based on double entry bookkeeping. This technique has several disadvantages concerning application neutral financial data storage. Therefore, the Resource Event Agent (REA) ontology was introduced to negotiate the current drawbacks. It describes concepts and its relationships semantically. Nevertheless, there exist a set of different REA models that reveal several lacks in context of formalization and specification. Although several REA approaches use formalized languages like the Unified Modeling Language (UML), they are too general to fulfill the Guidelines of Modeling. Due to these reasons, the paper adapts the SEMantic-based Planning Approach (SEMPA) for a formalized REA modeling. By annotating process activities semantically, the approach emphasizes on business control. Furthermore, process activities are linked together representing all valid solutions of a process flow representing behavioral aspects of REA. The goal is to present a formalized approach that enables the integration of diverse REA models.

Keywords

INTRODUCTION

Since the 15th century, double entry bookkeeping has dominated financial information by recording financial data on simple trade transactions (Vandenbossche and Wortmann, 2006). Nevertheless, financial transactions have become more complex and therefore, double entry bookkeeping reveals its drawbacks (McCarthy, 1982; ibid, 2006). By implying limitations in storing financial data, double entry bookkeeping cannot store data application neutral. Due to this reason, it does not provide any true and fair view on the performance of a company (Kirkegaard, 1997). In order to overcome these drawbacks, the Resource Event Agent [REA] accounting model was developed (McCarthy, 1982). But for various reasons, this approach was not adopted, yet. One reason is that its semantics are too unspecific in favor of a usage in accounting information systems (Scheller and Hruby, 2009). Moreover, the REA accounting model has changed and amplified over the time to an enterprise ontology (Hesselund, 2006). There exist various approaches to describe REA models (e.g. Verdaasdonk, 2003; Murthy and Wiggins, 2004; Batra and Sin, 2008). The multiplicity of different models leads to redundancies inside the REA ontology. Furthermore, there does not exist any approach that describes a process model to develop an ontology that fulfills ontology engineering requirements. These requirements are the semantically consistency, reusability and completeness of underlying process models, as well as the control flow structure (Heinrich, Bewernik, Heneberger and Krammer, 2008; Janisch, 2009). Moreover, the REA model is underspecified and insufficient formalized (Gaily and Poels, 2005). On the one hand, an ontology enables the communication of model artifacts, improves monitoring over process activities, and harmonizes the terminology (Gruber, 1993). The lack of specification and formalization of current REA models fail these goals. In order to negotiate these drawbacks, the paper’s goal is to create a flexible REA model that considers dynamic aspects and enables an automatic adaptation of extensions and changes. This is the basis improving the communication of financial data. For this reasons we are using the SEMPA approach to fulfill the requirements of ontology quality (Heinrich et al. 2008).

The course of the paper is as follows: the business need of REA will be described in the first place. In this context, we are going to explain the drawbacks of double entry bookkeeping within Enterprise Resource Planning [ERP] systems. Then the REA accounting model is presented. Due to its current inconsistencies, we are going to examine, if the REA model fulfills
CURRENT DRAWBACKS OF ERP SYSTEMS

Primarily, the REA model was developed in order to negotiate the drawbacks of accounting systems in a data base world. Current ERP systems track financial data in different modules (e.g. payroll, job costing, and order entry) and consolidate these financial data to meet legal demands for government agencies, creditors, and shareholders. Nevertheless, the problem of traditional ERP is to fit with purpose neutral accounting onto the organization database (McCarthy, 1982). Moreover, ERP general ledgers based on the double entry bookkeeping. There is no distinction between the accounting technique and the data model. Due to this reason, financial data are not recorded in their original form at the transaction level. In fact, there is no separation between the data model and the underlying business logic. Therefore, current ERP systems cannot provide purpose neutral accounting (Verdaasdonk, 2006). Consequently, ERP systems are unable to provide appropriate accounting data for various decisions making. Furthermore, they both do not consider ex-ante accounting information and knowledge about business processes (Vanderbossche & Wortmann, 2006; Verdaasdonk, 2003). Due to these reasons, understanding and facilitating financial decisions are difficult even for domain experts, because the standard process of identifying the addition of overhead cost obscure the real cost structure in ERP systems (Geerts and McCarthy, 1999; Riebel, 1994). Therefore, ERP systems have to deduce cost components and its underlying transactions in order to fulfill the requirement of cost accuracy. Today, there is a gap between running data storing, periodical planning, and financial ad-hoc analysis (Riebel, 1994). The concepts for negotiating this gap are not new (Goetz, 1939; Riebel, 1956; Back-Hock, 1995). By separating data environment and application, a purpose neutral data model is achieved that enables support for a broad spectrum of management issues (Verdaasdonk, 2003).

Regarding to the drawbacks of current ERP systems, McCarthy has provided the economic REA Accounting Model (McCarthy, 1982). Its goal is to achieve a high level of task accuracy providing a conceptual model of accounting systems that links business events with IS analysis and design (Dunn, Cherrington and Hollander, 2005). Therefore, it concerns core MIS applications like database applications, systems analyses, and design (Batra and Sin, 2008). The model orientates on the principles of the Basic Pecuniary Record (Goetz, 1939) and the German Grundrechnung (Schmalenbach, 1956). Both accounting approaches and REA capture financial data at the business event level, characterize business entries with semantic meaning, and are application independent. Beyond these facts, the REA model provides an alternative to the general ledger by representing business processes and data jointly (Verdaasdonk, 2006). Finally, we believe that the REA model has the potential to negotiate the current ERP drawbacks by storing business data, giving these data semantic meaning, and providing an application independent model.

THE REA ACCOUNTING MODEL

The REA Model is an abstraction which is describing economic exchanges within an enterprise. Its basis is the microeconomic theory of the firm (Borch and Stefansen, 2004; Jaquet, 2004). By modeling entities independently from its application, they can be composed in order to define business processes for the enterprise. The model itself consists of three core artifacts: Resource, Events, and Agents (McCarthy 1982). Later, the Commitment artifact was added. Moreover, these artifacts supplemented with a set of relationships that build an ontological basis for an accounting information system representation (Geerts and McCarthy, 2000; Hesselund, 2006) The Resource represents goods or services that can be bought and sold. Events are economic transactions, e.g. concrete acts of selling or buying, whereas Agents act as sellers or buyers of recourses. Agents are a representation of individuals, departments, or companies. The Commitment entity constitutes an obligation within an event, e.g. invoices have to be paid. These four core entities have diverse relationships and attributes. The relationship between the resource and the event entity explains the stock-flow relationship. A resource can be used, consumed, given, taken, or produced within an event. Therefore, the resource transformation is explained as stock-flow. The custody relationship concerns the responsibility of internal agents for specific resources. If commitments specify operational level phenomena directly, the reserve relationship will be used by connecting commitment with the actual phenomena. The materialization of a commitment into an event describes the executive relationship. It represents a reciprocal commitment between two agents. Both the accountability and the involvement relationship specify subtypes of the inside relationship of an agent and are self-explanatory (Batra and Sin, 2008). The reciprocal relationship is related to the commitment entity. It means that one commitment leads to a second commitment (e.g. shipment and invoice). The linkage relationship refers to the resource entity. By describing substitutable resource types or components, the linkage relationship avoids a part entity that represents also a resource (Hruby, 2006). Duality describes the complementary relationship between two events in order to complete a transaction. An event associated with an increment of a recourse must be paired with an event associated with a
decrement. Finally, the association relationship describes responsibilities between two internal agents, the cooperation between two external agents, or the assignment between internal and external agents (Geerts and McCarthy, 2000). The following figure expresses both concepts and their relationships.

![Figure 1. Components of the REA model (Batra and Sin 2008)](image)

The described concepts of the REA model provide patterns for full enterprise modeling. Furthermore, the concepts concentrate on the semantically description of economic exchanges. Whereas primarily the REA model was introduced as an accounting oriented metamodel, it has evolved to a full enterprise ontology (Hesselund, 2006). The next section describes the basics of an ontology and explain, why REA complies with ontology requirements.

**REA AS AN ONTOLOGY**

The term ontology is a composition of the Greek *ontos* (being) and *logos* (word). In a broader sense, an ontology provides a category system that account for a certain vision of the world. The meaning of ontologies has changed from a more epistemological position to the focus of representing knowledge, especially in context of IS. According to Sowa, an ontology is a product of a study of categories (Sowa, 2004). These categories may exist in some domain. Otherwise, Gruber supplies a definition concerning the IS focus. He defines an ontology as a “formal, explicit specification of a shared conceptualization” (Gruber, 1993). Conceptualization refers to an abstract model, whereas explicit means that elements have to be clearly defined. Formal indicates that the specifications are machine processable (ibid). This definition takes emphasis on the representation of knowledge of a domain that describes a set of objects and their relationships by a vocabulary (Breitman, Casanova and Truszkowski, 2007). In summary, ontologies represent a systematic and formalized description of a domain, represented in a logical language in order to achieve a commonly agreed upon understanding of a domain (Church and Smith, 2007; Gruber, 1993). Ontologies can serve for diverse purposes: On the one hand, they are used for the communication between people and computational systems or computational inference; on the other hand, they concern the organization of knowledge and its reuse (ibid).

In fact, within the literature, REA is described as ontology. But it is questionable, if REA is conform to the requirements of full enterprise ontologies and their implementation. Firstly, REA does not include concepts to define the context of business relation. Secondly, duality and reciprocity are modeled as one to one association, although they are many to many relations in practice (Blommenstein, 2006). Thirdly, concepts are not described at the detailed level. The missing specification leads to misinterpretation of the current REA model, e.g. what enumerations have the stock flow relationship? (Vandenbossche and Wortmann, 2006; Hruby and Kiehn, 2006). In addition, the REA ontology lacks of formal representation (Gailly and Poels, 2006; Guizzardi and Wagner, 2004). There are a number of isolated REA extensions which complement drawbacks of current REA model (e.g. Batra and Sin, 2008; Jaquet, 2006; Murthy and Casper, 2004). Due to this reasons, the first research question states as following: Does REA and its artifacts represent an ontology that match the requirements of the given ontology definitions by Gruber and Sowa?

The REA ontology represents economic exchanges and therefore, it provides a top-down decomposition of the enterprise value chain among diverse business events. The value chain itself consists of individual processes that are decomposed into
business process specifications. These process level models can be specified into task level models that specify both logical sequence and activities (Church and Smith, 2007). Otherwise, REA concentrates vertically in terms of entrepreneurial logic (value chain) and workflow detail. On the other hand, it focuses horizontally on terms of type and commitment images of enterprise economic phenomena (Geerts and McCarthy, 2000). Therefore, there is a differentiation between conceptualization of economic phenomena concerning the value chain and abstract phenomena. Finally, the REA ontology corresponds to Sowa’s ontology definition. In addition, REA supplies a knowledge representation by describing business events at different aggregation levels both functionally and semantically.

Concerning this aspect, REA does also comply with Gruber’s definition. Nevertheless, the REA ontology is underspecified and insufficient formalized. Due to this reason, the use of REA in real applications is still limited (Gailly and Poels, 2005; Gailly and Poels, 2006). Furthermore, REA describes only a data model that represents the static aspects of a system (Batra and Sin, 2008). Behavioral aspects have to be modeled separately leading to a lack of integration of structural and behavioral aspects (Murthy and Wiggins, 2004). In fact, there exist approaches to overcome these drawbacks by using object oriented notations like the UML (Batra and Sin, 2008; Borch, Jespersen, Linvald and Osterbye, 2003; Gailly and Poels 2005; Geerts, 2004; Murthy and Wiggins, 2004). But all these approaches have a lack of a formalized methodology in context of a quality ontology engineering in common. The problem of using UML is its generality. Although, object oriented methods offer a broad spectrum of unique notations for both static and dynamic views, the notations are also used for different purposes (Fieber, Huhn and Rumpf, 2008).

Due to this reason, current REA approaches do not fulfill requirements of model quality. In context of software quality, the term itself is defined as “the totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs” (ISO, 2001). Therefore, software quality depends on the quality of the software development process and of the underlying models (Balzert, 2008). With respect to a high quality model, we follow the Guidelines of Modelling (GOM) (Becker, Rosemann and Schuette, 1995). These principles are accuracy, relevance, efficiency, comparability, and systematic composition. Accuracy and relevance means adequacy in construction of a model. Considering the described drawbacks of current REA models, our second research question arises, how to create a REA model that fulfills the mentioned requirements in respect of formalization and quality?

**SEMPA PROCESS MODEL**

Regarding to the research question, the paper develops a formalized framework that ensures ontologic clarity, enables the redesign of process models, and meet the GOM requirements. This is reached by presenting a model based on the Semantic based Planning Approach (SEMPA) (Heinrich et al., 2008). We have chosen this approach, because in comparison with other semantically approaches like SHOP2 (Sirin, Parsia, Wu, Hendler and Nau, 2004) or GOLOG (McIlraith and Son, 2002), this approach comply with both GOM and the main requirements concerning the planning of semantic process models (Heinrich et al., 2008). They are eight main requirements. The first one deals with the composition of business process activities and their description. Secondly, a process activity has to be used multiply. Thirdly, a data type has to assign to every input and output parameter of a process activity. Fourthly, all input and output parameter are exclusive, even if there are composed. Moreover, every single aspect of a parameter has to be considered. By defining inference mechanisms for the semantic analyses of the parameters, each class and relation of the REA ontology is considered. The seventh requirement refers to the control flow, e.g. ExclusiveChoice, ParallelSplit, or Synchronization. Finally, all valid solutions of process models have to be modeled. A process model is valid, if from an initial state a defined target state will reached. All these requirements are necessary to create a model in order to process numeric and alphanumeric data of a process model. Moreover, it is possible to describe propositional and description logic in order to achieve a semantic integration (ibid). The SEMPA approach consists of four steps. Within the first one, the predecessors and successors relations of a process activity are identified. If there does not exist an ontology, yet, an ontology is created as a terminological bases for the process model. It describes concepts and their attributes of a business process. The next step deals with defining both input and output parameters of process activities within a process library. By defining process parameters, the meaning of each process activity becomes obvious. Based on the initial state, a backtracking traversing is realized in the third step. Its goal is to create a graph which describes the dependences between the different process activities. It is called Activity Relation Graph (ARG). Each node of this graph represents a parameter. The connections between them describe the predecessor and successor relations. Whereas the ARG does not contain the direct predecessor and successor relationships between different process activities, a forward tracking is executed. The result of the final step is the Activity State Graph (ASG) which represents a generic basis for a process model. The ASG is represented by an UML activity diagram. Moreover, this activity diagram refers to both static and dynamic aspects of a business process. Therefore, the result is a REA model that conforms to the generally accepted modeling principles, considers behavioral aspects of a business process and enables a semantic method in order to map diverse REA models.
DEVELOPING A BUSINESS PROCESS

This section illustrates our approach by designing the process “Sale”. Firstly, we have developed an ontology suitable for the REA accounting model. Its function is to unify the common terminology. A detail of the ontology is given in the figure below.

![Diagram of the REA Ontology](image)

Figure 2. Detail of the REA Ontology formalized by OntoStudio 2.1

The basic idea of the REA model is represented in the ontology by the relations and concepts. Agents are partners within the business itself who can be customers or suppliers. The resources can only be changed by contracts (events). The contract is the central concept of the ontology, because it forms the interface between agents and resources. Each of the concepts contains attributes. For example, the contract has a type, a state, and a value. The marrow of the ontology is that one can comprehend the relation given between the different entities of the REA model and anytime one can define exactly the context by checking all the attributes of an ontology. Another important attribute of the ontology is the representation of duality. Every increment implicates a decrement – every increase of the resource cash implicates a decrement of the resource article.

The next step deals with the enrichment of the process model with the metadata being presented in the ontology. We choose the process Sale to illustrate our approach. This process itself and the different relations are already completely REA-suitable defined by Hunka et al. (Hunka, Hucka, Kasic and Vymetal, 2008). We have defined the input and the output parameters for each activity in correspondence to the ontology. Therefore, we know exactly what circumstances are necessary to perform an action and the possible changes. The annotation enables the control of the process itself, because an activity can only be entered if all conditions add up. For the process activity Classify Customer the attribute contract has to be received and the resources have to be available. The process activity consists of two input parameters, Order and ContractItem. Firstly, the Order parameter describes the contract state by using two classes (contract and contractstate) and the attribute received. Secondly, the parameter ContractItem contains the classes contractItemAmount, contractItemQuantity and resourceArticleQuantity in order to check the contract item amount. If the contract item amount is more than 50.000, the contract state is sendable. Otherwise, it is not.

The process activity results in different states of the contract and in different customer classifications. Of course, the occurring result depends on the parameters of one action. The semantic annotation represents the possible results.
Figure 3. Annotation of Process Parameters

Within the following step, the process actions with their annotations are linked together in the ARG. The semantic annotations are split up according to cases. The ARG shows the relations between predecessors and successors of activities. At the end of this step, all admissible solutions are represented within the ARG. Therefore, it is possible to plan and analyze non-deterministic activities and to check the state of an event. Corresponding to REA, the annotations ensure that the contract is carried out successfully only if all resource increments and decrements are passed. The following figure represents an example taken from the sales event.
Within the last step, an UML activity diagram is created starting from the first activity from the ARG. Each use of a process activity leads to a new state. Hence, the amount of used process activities is reduced, because not every activity can be executed in a certain state. A process activity is executed, if all specified conditions matches within a state and do not violate a restriction in order to avoid infinitive loops. Moreover, within the activity diagram, a control flow is deduced. It represents sequences, exclusive choices, parallel splits, and simple merges. A sequence is existent, if one process activity depends on the antecessor. Exclusive choice describes partial dependences, whereas a parallel split enables parallel process activities. Finally, a simple merge is modeled by joining exclusive choices.

It is to see that the activity diagram represents the business process composition. The diagram is the basis for further specifications e.g. in Web Ontology Language (OWL). The described process model ensures the testing of completeness, temporal aspects, and validity. The testing does not execute automatically at the current state. Therefore, the process flow has to check manually (structured walk through). So, the approach ensures a high level of formalization enabling the presentation of relations between the different concepts.
Figure 5. UML Activity Diagram
CONCLUSION

The paper deals with the formalization of the REA ontology using the SEMPA approach. Current ERP systems lack of appropriate methods storing financial data application neutral. Therefore, it is impossible to provide any fair view on the company’s performance. In order to overcome these drawbacks, McCarthy has developed the REA accounting model. Over the years, the REA accounting model has evolved to a full enterprise ontology. By defining resources, events, and agents independently from its application, they can be composed to describe business processes within an enterprise. The REA concepts and their relationships describe business processes semantically. Nevertheless, the REA model is not adopted, yet. In fact, one reason is the popularity and acceptance of current methods within ERP systems, e.g. the general ledger. But on the other hand, the REA model itself lacks on formalization, specification, and ontological clarity. Initially, it is the goal of ontologies to explicate knowledge by specifying abstract models via harmonizing domain specific terminology. Whereas REA provides such a concept, it does not contain ontological clarity. Thus, our research question concerns the negotiation of missing model formalization and clarity.

There exist various approaches to build a REA ontology. However, they do not provide mechanisms validating enterprise schemata as well as business process chains. Furthermore, they use modeling languages that are too general. Due to these reasons, the model quality cannot be ensured. Hence, our contribution is the presentation of an approach that overcomes these drawbacks. By providing a formalized process model, a set of valid solutions of process activities is generated. On the one hand, the generation process permits the optimization of business processes; on the other hand, it is possible to represent and match relations between different REA ontologies. Moreover, the ability to match different REA ontologies is the basis for building a unique terminology. A unique terminology of REA models achieves Gruber’s intentions of a shared and explicit conceptualization. In context with the semantic enrichment of process activities, the communication of the REA model will be improved. Improving communication of a formalized ontology leads to interoperability and to an adoption of REA.

But it has to be stated that modeling with SEMPA is extensively. The semantic annotation enables the reuse of process activities, improves adaptivity to various applications, and ensures the usage of control flow structures. Our target is to develop a process model that works independently from any technology and generate UML activity diagrams automatically in order to make a contribution to REA adoption.

REFERENCES


