Concept integration precedes enterprise integration

Jeusfeld, M.A.; de Moor, A.

Published in:
Proceedings of the 34th HICSS

Publication date:
2001

Citation for published version (APA):
Jeusfeld, M. A., & de Moor, A. (2001). Concept integration precedes enterprise integration. In R. H. Sprague (Ed.), *Proceedings of the 34th HICSS* (pp. 1-10). IEEE Press.

Concept Integration Precedes Enterprise Integration

Manfred A. Jeusfeld, Aldo de Moor
Tilburg University, Infolab
Postbus 90153, 5000 LE Tilburg, The Netherlands
{*jeusfeld,ademoor*}*@kub.nl*

Abstract

The integration of enterprises in a vertical market is not solved but rather facilitated by information technology. One aspect is the coupling of heterogeneous information systems from the participating enterprises. However, before this integration can be tackled, the enterprises have to create a common set of concepts to discuss their cooperation. We call this the inter-organizational concept base and present a proposal on how to structure such a concept base and how to co-develop it by participants from various enterprises. Product ontologies are bundled into reference models for certain industry sectors and serve as a starting point for the discussion about concepts. The second ingredient are explicit representations of norms that describe who is supposed to participate in which part of the discussion process. The end result, the inter-organizational concept base, is the input for an inter-organizational workflow modeling to specify precisely the enterprise integration.

1 Introduction

Enterprise integration is a relatively new research area. Its goal is to investigate how the various units of an enterprise can be more tightly integrated by means of modeling information systems development. Two directions have been pursued. In the *enterprise modeling approach*, models have to represent the key entities of an enterprise, i.e. its organizational structure, its resources, its business processes, its subjects, its goals, its constraints etc. [3]. Such approaches have a descriptive nature. The enterprise model is mainly used to understand the enterprise and to support the development of information systems. In *workflow approaches*, the enterprise is seen as a network of agents whose collaboration is defined by a process or workflow model. The purpose of these approaches is to prescribe the behavior of actors in the enterprise [9].

The two approaches are merged within enterprise modeling tools like ARIS [12]. The models produced by ARIS are used within ERP systems like SAP R/3[1]. ARIS covers organization structure, informational entities, business processes (control aspect), and business functions. However, in addition to system models, process performance figures can be assigned to business functions. In the process design stage, models are created and partially tested by simulation. If satisfactory, they are mapped to the operational system, e.g. an enterprise resource planning system or a workflow system. A process monitoring service is used to detect bottlenecks. Each business process has a responsible process owner who supervises the performance. The business processes are enacted as workflows which integrate the existing information systems of the enterprise. This is roughly the vision of ARIS. It stresses optimization of processes where the optimization goals are defined by the business process owner, e.g. the department head. For cross-organizational process integration, this approach falls short: We can no longer assume that there is a single process owner. Instead, stakeholders from different companies have to agree on any change that affects their interest. But how can they know that their interests are affected by a change? The main goal of this paper is to sketch a method to deal with these obstacles systematically. We focus on the following aspects. First, *common terminology* has to be co-developed and co-evolved. That standardizes the language used in communication, but more importantly, it is a prerequisite to the design of integrated information systems. Second, the development of the common terminology shall be guided by *explicit norms* which encode the rights and obligations of each stakeholder of the integrated information system.

As an illustrating example we investigate an Internet-based system which is currently developed for supporting

This work was supported in part by the Commission of the European Communities under Esprit contract 26.895 MEMO (Mediating and Monitoring Electronic Commerce).

[1] see [8] for a good presentation of business engineering with such models

early stages in business to business electronic commerce [10]. These early stages are *searching* (for business partners, product information, etc.) and *negotiating* (about possible contracts). If a contract is made, then the MEMO system supports the execution of the contract via an EDIFACT-based workflow management system. The use of the system is limited to member companies of a specific industrial sector. Such companies can and publish their company profiles and product catalogs to the other member companies. Some of the member companies can provide additional business information like trade rules, credibility of companies, and product ontologies to mention a few.

1.1 Problem definition

The problem setting is visualized by the lifecycle in Figure 1. It is a refinement of the ARIS lifecycle [12] but it adds multiple stakeholders instead of a single *process owner*. A group of stakeholders is responsible for defining the inter-organizational concept base of any kind, e.g. an ontology. From that, an enactable workflow model is generated and enacted (e.g. by a workflow management system). Stakeholders observe the enactment and decide about changes to the common enterprise model. But: *How can a fair decision process be organized when stakeholders are potential competitors?* In the case of stakeholders with equal rights, the fairness rules have to be adopted by the group of stakeholders. We call rules which guide the behavior of members of a group *social norms*. They circumscribe which behavior is regarded as acceptable.

Socio-technical specification methods such as Soft Systems Methodology and ETHICS strongly involve the various stakeholders in the specification process. However, some drawbacks are that they provide little concrete guidance in detailed IS requirements elicitation [17]. Furthermore, these methods are not easy to integrate in a continuous, evolutionary specification process as needed in virtual communities. A very important drawback, finally, is that the mechanisms for selecting the stakeholders to involve in a *particular* change are very crude, often involving too many stakeholders or none at all.

Apparently, any stakeholder can observe problems and feel the need for a change of the system since the interest of her enterprise is affected[2]. The challenge is that such changes also affect partner stakeholders. Hence, a discussion [11] about change can be expected. For such a discussion to be well-structured and focused, two aspects need to be paid attention to. First, there must be certain rules of discourse. Second and more relevant in this paper, it must be ensured that exactly the right subset of stakeholders is invited to participate in any of these discussion. The term 'exactly' is precise here. The rules upon which the participants are invited are the result of an earlier 'meta' discussion among the stakeholders themselves which is then enacted when a system change is due. A fuzzy interpretation of whom to invite into a discussion is counter-productive for a domain where commercial interests are dominating. An inter-organizational system is only acceptable, if the participants are sure that their rights are strictly enforced. To ensure an acceptable discussion process, the social norms guiding these discussant selection processes need to be represented formally and used to invite the appropriate users in a concept re-definition discussion.

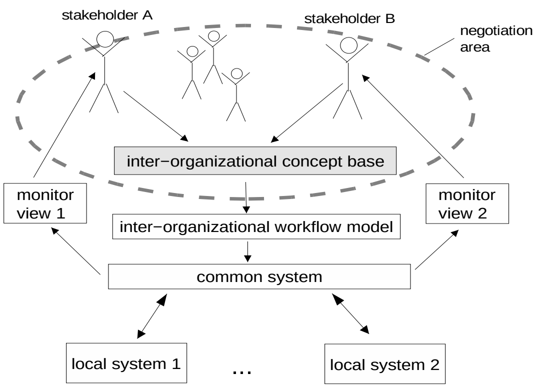

Figure 1: Lifecycle of common enterprise models

In this paper, we focus on the concept definition, i.e. the creation of a system of interrelated concepts including their definition which is then the basis to create an inter-organizational system used by multiple enterprises. We exclude the discussion of the workflow processes that are to be supported by the inter-organizational system. The reason for this is that we have identified concept creation as a important sub-activity that is undertaken within industrial interest groups as explained in section 2. The support for discussing changes to the system is however open for any change type, i.e. also changes to the workflow model and changes in the implementation. This has been demonstrated for the case of managing the review process of an electronic law journal [1].

The object-oriented approach promotes the co-development of data structures and operations. It was designed for system developers who start from a set of agreed requirements. It is unclear whether the object-oriented approach is suitable for the requirements analysis of inter-organizational systems where the conflicting interests of the participating enterprises are dominating. The goal of the paper is to present a method of co-development of ontologies which ensures that the rights of the participating enterprises are made explicit by so-called *social norms*, i.e. rules which encode how changes to the co-developed object (the ontology) can be discussed in the group.

[2] A change principally refers to any part of the common system. In the lifecycle model, the change occurs in the system model, in figure 1 the concept base plus the workflow model.

Our research method is technology-driven. Our hypothesis is that a system which makes the social norms explicit will lead to a more efficient discussion process with less breakdowns. The system has already been applied in other domains, specifically the domain of scientific journals where the rights of editors, authors, and reviewers were encoded and used. The results of these experiments were so encouraging that we believe that other application areas can also benefit from the approach. Enterprise integration is such an area: The representatives of the enterprises share a common goal, e.g. to integrate their information processing in order to save transaction costs. On the other hand, they expose their enterprise to their competition and require some level of right protection. The innovation of our approach is that we do not propose a general set of rules to protect rights. Instead, our system allows the participants to define both their rights (the so-called *action norms*) and the rules for disussing changes to the rights (the so-called *definition norms*). The work here is in the context of an electronic commerce project that has just begun its pilot studies. Hence, empirical results on the usefulness of our approach cannot yet be presented.

The rest of the paper is organized as follows. Section 2 presents the example of the Dutch construction industry which is in the process of setting up integrated information systems for their business. A common product ontology (i.e. a system of concepts) is used to integrate existing information systems of enterprises in order to exchange information. The product ontologies define the way how an enterprise can classify its products. The discussion process about changes to the product ontologies is the focus of this paper. Section 3 then shows the role that formal representations of social norms can play in selecting participants for a concept re-definition discussion. We claim that our approach leads to more acceptable changes to the common product ontologies which are the basis of a common electronic commerce platform for the industry sector.

2 Ontologies in a B2B E-Commerce system

The MEMO project [10] has the goal to facilitate early stages in business-to-business electronic commerce. The central idea is to develop a so-called *broker system* to which companies can register and export their business data into. The MEMO project started in 1999 and will last until mid 2001. Partners include a large bank, two universities (including ourselves), a software house, a consulting company, an Internet service provider, and two chambers of commerce. The partners are distributed between Spain, the Netherlands, and Germany. Recently, the first prototype has been released and pilot studies are starting by end 2000.

We concentrate in this paper on the business data management, i.e. how to change the content of the systems database which subsumes company profiles, product profiles, business ontologies, Member companies can search for business data, find potential partner companies, and then negotiate about a contract with selected partner companies.

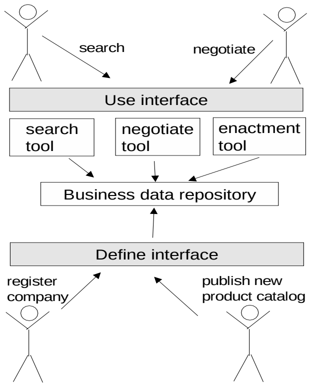

Figure 2: The interfaces to the MEMO system.

Figure 2 shows the two interface of the MEMO system. The *use interface* allows users from member companies to execute the main business functions supported by the MEMO system. The *define interface* provides the facilities to add new companies as members and to modify the systems business database, the so-called business data repository. The issue of this paper is about the define interface of the system, i.e. how to make the co-evolution of the product ontologies in the business data repository more agreeable among the stakeholders.

Subsequently, we elaborate what role product ontologies play in business-to-business electronic commerce. Essentially, they circumscribe how users of the system can advertise their products.

2.1 The Dutch Construction Industry

The broker system is specialized for a certain vertical market. We consider here the Dutch construction industry. It is subdivided vertically as follows[3]. *Producers* are companies which create any kind of material used in the construction industry, e.g. tiles. *Wholesalers* package such products into new products, e.g. tiles including the right glue and provide a geographically distributed buffer storage for products. *Architects* plan construction projects and specify products or product types to be used. Finally, *contractors* execute a construction project plan. The market is international and highly standardized. National and and European laws constrain the products to be used and the construction methods. For example, walls in a building must guarantee certain degrees of fire resistance. Companies and other

[3] We only present a simplified view of the construction industry. Functions like transport, financing, insurance etc. are left out to keep the presentation short.

stakeholders in the market are nationally organized in *interest groups*. Those groups propose standards and advise their members on how to deal with the standards.

The emergence of the Internet has a great influence on this market since it allows a quicker and more customized information exchange. So, an electronic commerce system is a logical step. The fast developments have led to considerable turbulences among the companies. For example, wholesalers fear that they lose their mediator role when a contractor can directly order products from the producers. Currently, the contractors are very dependent on the wholesalers because they simply do not know where to get the products from directly. Another fear is that an electronic commerce system might result in advantages to certain companies because they are more visible in the electronic system. A simple example is the ordering of company names in a list: a company appearing first in a long list is better visible than a company at the end. So, the broker system has to be fair in a certain sense.

The particular issue for this paper is the co-development of the common product ontology that member companies use to classify their product descriptions. So, the inter-organizational concept base cf. figure 1 is in this case the collection of common product ontologies. The product ontologies that we discovered in the construction industry are role-specific but not company-specific. The product ontologies for the different roles co-exist without interference. It is the responsibility of the supplier company to classify its products into those ontologies that are used by her customers.

2.2 Product Ontologies

Product ontologies are networks of concepts which denote product categories used in the industry. The construction industry is so well-standardized that practically all products can be assigned to a specific place in the product ontology. Since the market is vertically organized, specialized ontologies are developed by the interest groups. For example, architects use an ontology which names product categories according to their function. A tile is a product which covers the floor. So, it is an incarnation of the concept *floor cover*. The structure of the ontology determines the search for suitable products (and their suppliers). An architect uses the concept 'floor cover' to find suitable solutions for her floor covering task. It can be argued that a supplier is very much interested to have its products classified rather close to these ontology terms. The interesting point is that the ontologies are owned by the interest groups. Hence, the members of the interest groups are co-designing them as a service for all members. On the other hand, *product profiles* (product sheets) are published locally by the suppliers only. They may or may not standardize the data structures. Indeed, the current situation shows very little standardization on product profiles.

Communication with user groups in the Dutch construction industry revealed that different stakeholders (see above classification) use different concepts which may or may not overlap. Hence, two different users can find the same product by referring to different search terms. Consequently, multiple product ontologies, i.e. hierarchies of business terms have to be supported at the same time. The product ontology can be considered to be part of a *user profile*: it determines the vocabulary that a user employs to find products. Subsequently, we propose a generic ontology schema that is used to represent ontologies in the broker system. It is encoded in the ConceptBase repository system [5] which offers a conceptual modeling environment which was used in enterprise integration earlier [7].

An *ontology* is defined here as a set of concepts and other ontology elements such as lexicals and strings, which are interrelated in a semantic network. Each concept also features a natural language definition that can be looked up. The same concept can have multiple denotations (lexicals in different languages). Attributes of concepts, e.g. the size of a door, are also considered as concepts. A *product catalog* is a list of product descriptions, usually from a single supplier. The way how products are described depends on the product category (identifiable with the concepts of the product ontology). Some products are described by physical properties (size, heat resistance, weight, geometry) while others are described more by the way how the products can be used to solve a task.

2.3 Product Ontologies and Data Representation

This diversity creates a problem for product data representation. In fact, each company has the interest to describe its product in a way that stresses its *distinctive features*. On the other hand, companies have the interest to describe products in a way they they can be found by the right customers. So, a company has an interest that the common product ontology contains just the right concepts to describe its own products. This is an inherent conflict since competing companies have not the goal to have the competitor's product better classified than their own. Hence, the challenge is to *co-develop the product ontologies in such a way that the interests of all stakeholders are preserved.* Existing business-to-business systems like [2] use a very general schema for product description:

```
(EANcode, supplier code, description,
product-group, last update)
```

The EANcode identifies a product using an international coding scheme. The EANcode does not characterize the product. It identifies the country, the supplier and the product of that supplier. The supplier code is an internal product identifier that the supplier uses independently of the EAN-code. The description is plain text about the product. The product group is the name of the product category for which a given product is an example of. Finally, a date field is used to store the time of the last update of a product record.

Such a simple product data structure is not suitable for the purposes of MEMO for the following reasons. First, a product can only be grouped into a single product group; in MEMO, multiple product ontologies for different user types (contractor, agent, architect, wholesaler, etc.) are incorporated and thus require the ability to group the same product into multiple product groups. Second, a purely textual description is only supporting keyword based search.

To achieve a more detailed product data structure without loosing the generality of the approach, a closer look into existing strategies to represent product data is required. We consider two examples for supplier-defined product descriptions. All descriptions are directed to potential buyers of the product.

The first example is from a supplier for facade panels [15]. The product called 'Trespa Meteon' is described by around 20 attributes which are mostly numerical. The attributes are sorted into categories: physical properties, optical properties, mechanical properties, thermal properties, chemical properties, and fire behavior. The physical properties are subdivided into specific mass (value example: +/-m 1400 kg/m^3 cf. ISO R1183-87), dimensional stability, water absorption, vapour diffusion coefficient, and coefficient of thermal expansion. Optical properties has just one attribute: color stability hours (value example: 4-5 (3000 hrs; Xenon test) Grey scale cf. ISO 105 A02-87). Fire behavior has 4 attributes, namely the fire behavior norms fulfilled for four countries. Such a product description is suitable for buyers who need to evaluate whether the given product fulfill certain specifications. Typically, architects shall be interested in such descriptions. The second example is from a supplier of roof windows [16]. A product description merely consists of a picture and about 5 lines of text describing the way of opening the window, the opening degrees, and locking mechanism. Such a description is close to the simple product data structure proposed by EC-Gate.

The construction sector is characterized by a relatively high level of organization, close cooperation between partner companies in project consortia, a high number of product suppliers, contractors, and other commercial partners like architects. The standardization in this industry is pursued by non-profit organizations. One of these organizations is HCP-EDIBOUW, a Dutch organization with the goal to enable electronic business in the construction industry. A relevant document from this organization is the "Branchemodel Elektronische Communicatie" [4]. Among others, it defines a so-called product sheet specifying relevant features to describe a product.

The values for the attributes are typically textual. Attributes from the product form category may involve numerical values. Performance numbers are augmentable by measurements norms, e.g. specified by an ISO code. The categories are suitable for the construction industry. Other business areas, e.g. the insurance business, would greatly differ. The principle of having a number of categories and for each category a number of attributes is however generic.

A product catalog has a certain structure, normally a table structure. Products are supplied by companies and have a product profile. The remaining issue to be solved is the mapping of external product catalogs to the above data structure and the product ontologies (compare figure 3). The product profile has to be accompanied by classification instructions which specify the meaning of the attributes according to the product ontology, more specifically the classification of product profile fields into concept attributes. Note that the ontology is co-developed by the members of the industry's interest group while the product descriptions and their data structures are completely controlled by the product supplier.

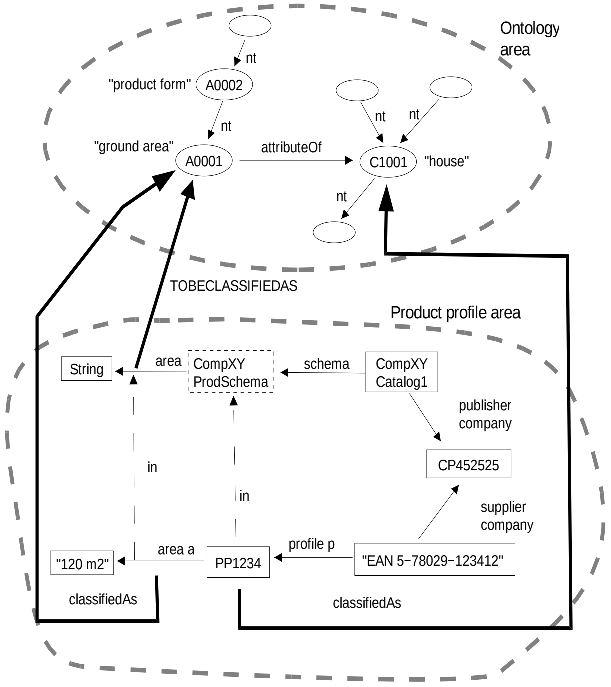

Figure 3: Product profiles classified into an ontology.

The upper part in figure 3 shows an excerpt of a product ontology. The lower part shows a product catalog schema and one example product `EAN 5-78029-123412` with a product profile `PP1234`[4]. It is associated to a company

[4] In principle, the same product can have multiple profiles to address specific information needs by different customer groups.

CP452525 and has a size attribute. The product is classified into a product concept of the ontology and the size field into an appropriate concept attribute of the ontology. The graphical notation is used here since it matches the internal data representation in the MEMO business data repository ConceptBase. The classification of products and their attributes is done via a rule mechanism in ConceptBase. The attribute classification is based on a schema analysis whereas the product classification is based on concept codes attached to product codes. The provision of necessary information about the classification is the responsibility of the company publishing a product catalog. The ontology schema which we use is treating concept attributes as special concepts (see [6] for more details on the technical implementation). This allows the MEMO search engine to offer search terms not only based on concepts like *floor cover* but also on concept attributes like *fire resistance*. For example, a user can search for all product descriptions related to *floor covers* that contain information about *fire resistance*. The attribute classification detaches the ontological search term from the product data structures which are heterogeneous and not unified at all. By this separation, we claim to have overcome the problem of data integration which has crippled earlier enterprise integration approaches. The multiple product ontologies do not integrate enterprise data. Instead they index them and make them accessible to a search engine that is pre-loaded with the product ontology of the user of the common system. It has to be said however, the our classification approach does not support the coupling of automated data processing tools of the companies.

A product can be classified into multiple concepts of an ontology. It has product profiles (possibly more than one for different markets) which associate descriptive attributes (fields) to a product. A product ontology is designed to be user-role-specific, ie. an architect has a different ontology from a wholesaler. Products relevant to both user-roles are classified into concepts of their respective ontologies.

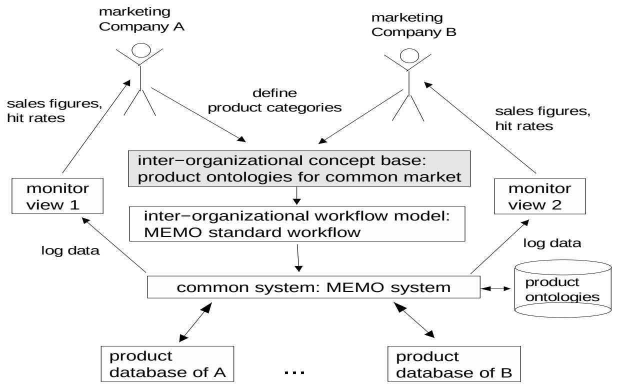

Figure 4: The lifecycle of product ontologies in MEMO.

Figure 4 specializes the original diagram of the problem definition (figure 1). The common concept definition is here limited to the co-development of product ontologies. Instead of a dedicated cross-enterprise workflow model, the standard MEMO workflow is prescribed. It basically supports the search for products, and the negotiation about contracts[5]. Finally, the common system is the MEMO broker system. It communicates with database systems hosting the product catalogs. They are classified into the co-developed product ontologies which are an integral part of the system itself. If a stakeholder of a company finds the performance figures of the common system unsatisfactory for her company, then she may propose a re-organization of the product categories in order to make her products more visible or more precisely focussed to customer search requests. Such a change is subject to a well-defined change protocol discussed in the subsequent section.

The above case study is a good example for a system that needs norm-guided evolution. Stakeholders (i.e. members of the broker system) are jointly responsible for evolving the product ontologies. Since there are multiple ontologies, different sub-groups are responsible for the concept definition. For example, the architect ontology is a matter of the architect interest group. In the MEMO B2B electronic commerce system, the product ontologies determine the accessibility of product profiles. The MEMO search engine allows the users to select search terms from their role-specific product ontology and returns the products which are classified into the concepts. As a consequence, member companies have a premier interest in product ontologies which are well-suited to index their products. If that is not the case for a company, then this company has a breakdown: it has an interest to start a discussion about changing the product ontology.

A second aspect is international trade. The European countries have national interest groups of companies. The Dutch product ontology for architects is not at all identical to the English counterpart. So, to facilitate cross-border trade, the two interest groups have to join forces and interrelate their ontologies. The integrated ontology can then be used to find suppliers for products independently from their country. Naturally, fairness is extremely important in such an integration. If one interest group dictates its own ontology to the partner, then the companies in the dictating interest group have a huge advantage since their products are ideally accessible to the combined marked. To prevent such problems, norms need to be defined that allow for the precise control of which ontological terms and relationships can be defined by whom. One approach to such norm-guided ontology evolution is worked out in the next

[5] Realistically, the negotiation should be regarded as industry-specific as well. Hence, a co-development of negotiation protocols should be considered. We leave this aspect however to the reader.

section.

3 Norms and Concept Definition

In the previous sections, we have outlined the important role that concept redefinition plays in enterprise integration, the complex ontological elements and relationships that need to be captured in these ontologies, and the many stakeholders involved. It seems hard to make a generic approach that can deal with all these variables. However, some assumptions can be made now when looking at how to focus and structure the concept redefinition discussions:

- The format of the discussions does not need to be formalized. Different combinations of stakeholders use different, often informal discussion approaches, ranging from informal face-to-face meetings to electronically mediated, newsgroup-like discussions.
- There are many different stakeholders involved in concept definition processes. Ontological elements may need to be discussed by widely varying subsets of stakeholders, depending on the kind of element, the ontology or ontologies in which it is included, etc.
- Different stakeholders have different definition rights. Some stakeholders only need to be informed of a concept definition modification (e.g. an architect will only need to be informed of a definition change made by the producers interest group in the product category ’floor tile’, while the wholesalers interest group needs to be asked for approval of the definition change, since their product packages depend on this definition).
- Formal norms need to prescribe which stakeholders to include in which definition change process. These norms can be derived from the informal norms already prevalent in the industry. Once formalized, they can be used to set up electronic discussion fora for particular definition changes, to invite the right participants, and to ensure that the interests of the various stakeholders are guaranteed. In this way, concept redefinition processes will become more acceptable, take less time, and will facilitate changes in other models, such as the enterprise and workflow models.
- Product ontologies are highly hierarchical constructs, often with many layers of details. Norms need to recognize the existence of these type hierarchies, so that redefinition behaviour can be specified at exactly the right level of detail.

Changes in norms and operational concepts are of a different order: the changes of these norms occur at the meta-level. They describe who should be involved in the change processes of concepts, and thus represent the concept definition decision making structure. The number of changes in these ’composition norms’ will in general be considerably less than the number of changes in the operational concepts. The norms are defined by the community itself. Members identify which kinds of concepts may, must, or may not be defined by which actors in the network. If adequate typologies of virtual communities in the domain are developed, reference models can be used to guide the definition of these norms. For instance, in a journal publishing community, there can be the (standard) norm that authors may not be involved in changes in the editorial process definitions. Norms can conflict. To deal with norm conflicts, a norm conflict resolution mechanism has been incorporated in our method, which takes into account both conflicts related to the different deontic effects of norms, and different levels of genericity of the norms.

There are three kinds of concept *definition processes*: *creations* of new ontological elements, *modifications* of existing elements, or *terminations* of obsolete elements.

Each *concept definition discussion* has three stages: some stakeholder *initiates* it, a group of stakeholders *executes* it, and one or more stakeholders need to *evaluate* the outcome before it is finalized. The mechanism for norm-guided knowledge-definition support is the one developed in the RENISYS method, that was specifically developed for the facilitation of legitimate user-driven specification processes of network information systems for virtual communities [1]. This approach has also been us for the evolution of workflow patterns for virtual communities [18].

In the subsequent section, the RENISYS system specification method is outlined. We then show how this method can be used to ensure that concept definition discussions are legitimate, i.e. conforming to the norms.

3.1 The RENISYS Method

The RENISYS (**RE**search **N**etwork **I**nformation **SY**stem **S**pecification) method facilitates the legitimate user-driven specification process. It supports the handling of breakdowns in the collaborative work of virtual professional communities. The method allows individual users who have become aware of a problem with either the way their work is organized, or with the support provided by the enabling technologies, to formulate their problems in terms of problematic *knowledge definitions*.

The method then determines which other users are to be involved in the resolution of these definitions. To this purpose, the *composition norms* that regulate the acceptable specification (i.e. concept definition) behaviour of actors (or stakeholders) in the community play an important role. An example of such a norm would be that wholesalers are permitted to create new product catalogs. To use norms, their deontic effect must be known. The deontic effect of a norm denotes whether the behaviour regulated by the norm is either permitted, required, or forbidden. The method calculates the *resultant deontic effect* of the complete set of composition norms that apply to a particular user and the specification process required to change the definition. In this way, it knows which users to involve in the *conversation for specification* (i.e. the concept definition discussion) in which the problematic knowledge definition can be legitimately changed. Additionally, or alternatively, a *discourse process* can be started in which users can critically examine background assumptions that determine the meaning of the various knowledge definitions making up the system specifications.

In the method, knowledge definitions are represented and reasoned about using conceptual graph theory [13]. One of the useful properties of this theory is that it creates implicit generalization hierarchies of graphs. This has the great advantage that properties of different sets of definitions can be concisely represented. Another advantage is that conceptual graphs can be easily mapped to (pseudo)-natural language constructs, thus allowing for more effective interactions between method and users. Dynamic deontic logic [19] is used to handle composition norm conflicts and calculate the authorizations of users involved in a particular conversation for specification. To model the moves that users can make within a conversation for specification, a Specification Process Model was developed, which is a variation of Van Reijswoud's Transaction Process Model [14]. This model links the speech acts that lead to a successful transaction with the speech acts necessary for the discussion of validity claims, and with those acts required for the critical discourse of background assumptions in the sense of Habermas's theory of communicative action. However, since we just stated that we do not intend to formalize the support for concept definition discussions in the construction industry, we do not further address this here.

3.1.1 Knowledge Representation

In RENISYS, four types of knowledge definitions are distinguished, of which two are relevant here: composition norms and type definitions[6].

Composition norms are meta-norms that determine the acceptable specification behaviour of community members. They include permitted, required, and forbidden compositions. The composition norm example mentioned above, which concerned a *permitted composition*, is represented as follows:

$$\begin{array}{l}[\mathtt{Perm\_Comp} : [\mathtt{Wholesaler}] \leftarrow (\mathtt{Agnt}) \leftarrow [\mathtt{Control}] \rightarrow (\mathtt{Obj}) - \\ \quad [\mathtt{Create\_Type}] \rightarrow (\mathtt{Rslt}) \rightarrow [\mathtt{Type} : [\mathtt{Product\_Catalog}]]].\end{array}$$

This norm is a very generic one, saying that any wholesaler is allowed to create any kind of product catalog. Translating the formal representation: it is *permitted* for any wholesaler *actor* to *control* (i.e. initiate, execute, and evaluate) the *creation* of product catalog *types*. However, more realistically, instead of this one generic norm there would likely need to be more specific norms for different kinds of wholesalers and product catalogs.

This norm is an example of a *privilege*. To safeguard the interests of stakeholders, two other categories of norms, *responsibilities* and *prohibitions* are also needed. The first category describes who must be involved in a particular specification process, the second category who may not be involved. An example of the former would be the required composition that wholesalers *must* be involved in the creation of new concept types (replace in the above norm the Perm_Comp label by Req_Comp).

The first categories describes who must be involved in a particular specification process, the second category who may not be involved. An example of the latter would be that no company is allowed to define the product profile of another company:

$$\begin{array}{l}[\mathtt{Forb\_Comp} : [\mathtt{Company} : *\mathtt{x}] \leftarrow (\mathtt{Agnt}) \leftarrow [\mathtt{Control}] \rightarrow (\mathtt{Obj}) - \\ \quad [\mathtt{Create\_Type}] \rightarrow (\mathtt{Rslt}) - \\ \qquad [\mathtt{Type} : [\mathtt{Product\_Profile}] \leftarrow (\mathtt{Poss}) - \\ \qquad [\mathtt{Company} : *\mathtt{y}]]]. \text{ with } *\mathtt{x} = *\mathtt{y}\end{array}$$

This *forbidden composition* says that no company x may change a product profile definition of another company y if x is not equal to y.

Type definitions define the meaning of network concepts. These definitions have two functions in our concept definition approach: they allow the structure of ontological definitions themselves to be captured, as well as (at the meta-level) the definition of the role and meaning of the concept definition processes. An example of the latter is that a concept discussion involves one or more stakeholders, requires a concept change proposal and problematic concept as input, and results in a changed concept definition.

$$\begin{array}{l}[\mathtt{Type} : [\mathtt{Concept\_Disc} : *\mathtt{x}] \rightarrow (\mathtt{Def}) \rightarrow [\mathtt{Discussion} : *\mathtt{x}] - \\ \quad (\mathtt{Matr}) \rightarrow [\mathtt{Change\_Proposal}] \\ \quad (\mathtt{Matr}) \rightarrow [\mathtt{Probl\_Concept} : *\mathtt{y}] \\ \quad (\mathtt{Rslt}) \rightarrow [\mathtt{Changed\_Concept} : *\mathtt{y}]].\end{array}$$

RENISYS contains a number of ontologies. A core change process ontology describes the top level concepts needed to define more specialized concepts. Community-specific concepts are stored in domain ontologies, e.g. the product ontologies. Concepts from the core process and domain ontologies can then be used to define the norms. This is a very brief introduction of the RENISYS method. Space is lacking to describe it in more depth (see [1] for more details).

3.2 Making Acceptable Concept Definition Changes

To ensure that concept definition changes are not only meaningful but also acceptable to the community of stakeholders as a whole, we now show how RENISYS can be used to ensure that only legitimate such changes can be

[6] Knowledge definitions are in conceptual graph notation, which we assume to be familiar to the reader. The syntax of the knowledge definition categories is explained in [1], and is not repeated here.

made. Assume the following (partial) RENISYS type hierarchy has already been defined[7]. The hierarchy is in conceptual graph syntax. Each concept that has subtypes is followed by a >-sign, its subtypes are then at the next level of indentation.

```
Entity >
  Actor >
    Architect
    Producer
    Producer_IntGroup
    Wholesaler
    Wholesaler_IntGroup
  Object >
    Product_Category >
      Floor_Tile
  Process
    Control
      Init
      Exec
      Eval
    Specify >
      Modify_Type >
        Modify_Concept
```

Each RENISYS specification process (e.g. type creation) is considered to consist of three *compositions*: the *initiation*, *execution*, and *evaluation* of the knowledge definition change process that is the objective of the specification process. For all three compositions of the requested specification process RENISYS calculates, for all users in the network, who is permitted or required to participate in the composition.

To illustrate, let us take the problem of the modification of a product category. We want to capture that individual producers may propose changes in a product category, that the producer interest group is then required to develop the proposed changes, and that the wholesaler interest group must either approve or reject the proposed changes. This knowledge is then represented in, respectively, the following composition norms:

$$\begin{array}{l}[\mathtt{Perm\_Comp}:[\mathtt{Producer}] \leftarrow (\mathtt{Agnt}) \leftarrow [\mathtt{Init}] \rightarrow (\mathtt{Obj}) - \\ \quad [\mathtt{Modify\_Concept}] \rightarrow (\mathtt{Rslt}) - \\ \qquad [\mathtt{Type}:[\mathtt{Product\_Category}]]].\end{array}$$

$$\begin{array}{l}[\mathtt{Req\_Comp}:[\mathtt{Producer\_IntGroup}] \leftarrow (\mathtt{Agnt}) \leftarrow [\mathtt{Exec}] \rightarrow (\mathtt{Obj}) - \\ \quad [\mathtt{Modify\_Concept}] \rightarrow (\mathtt{Rslt}) - \\ \qquad [\mathtt{Type}:[\mathtt{Product\_Category}]]].\end{array}$$

$$\begin{array}{l}[\mathtt{Req\_Comp}:[\mathtt{Wholesaler\_IntGroup}] \leftarrow (\mathtt{Agnt}) \leftarrow [\mathtt{Eval}] \rightarrow (\mathtt{Obj}) - \\ \quad [\mathtt{Modify\_Concept}] \rightarrow (\mathtt{Rslt}) - \\ \qquad [\mathtt{Type}:[\mathtt{Product\_Category}]]].\end{array}$$

Note again that these norms are still very primitive, for example, in the sense that no subdivisions are made for different product interest groups being responsible for different product categories. This demonstrates even more the need for an automated approach such as RENISYS in dealing with these complex norm patterns and potential conflicts.

Let us now take the modification of a floor tile concept definition as the *active specification process*. For the calculation of which users to invite in each of the three *active compositions*, RENISYS uses two functions, of which the semantics have been described in [1].

The function $\mathbf{D_{CN\_APPL}}(\mathit{user,comp})$ calculates which composition norms apply to user *user* for active composition *comp*. Applied to the example, this function could provide the following results (assuming that John is a representative of the producer interest group):

$$\begin{array}{l}\mathbf{D_{CN\_APPL}}(\mathbf{John}, \\ [\mathtt{Exec}] \rightarrow (\mathtt{Obj}) \rightarrow [\mathtt{Modify\_Concept}:[\mathtt{Type}:[\mathtt{Floor\_Tile}]]]) = \\ [\mathtt{Req\_Comp}:[\mathtt{Producer\_IntGroup}] \leftarrow (\mathtt{Agnt}) \leftarrow [\mathtt{Exec}] \rightarrow (\mathtt{Obj}) - \\ \qquad [\mathtt{Modify\_Concept}] \rightarrow (\mathtt{Rslt}) \rightarrow [\mathtt{Type}:[\mathtt{Product\_Category}]]].\end{array}$$

Thus, in this case, only one composition norm is retrieved, which says that John, in his capacity as product interest group member is required to modify the product category definition, once prompted by, for instance, an architect. Based on this set of applicable norms, the *resultant deontic effect* is calculated by the function $\mathbf{de_r}$ This function deals with potential norm conflicts by applying norm priorities to the norms in the set. If applied to the example, this function would return the following result:

$$\begin{array}{l}\mathbf{de_r}(\mathbf{D_{CN\_APPL}}(\mathbf{John}, \\ \quad [\mathtt{Exec}] \rightarrow (\mathtt{Obj}) \rightarrow [\mathtt{Modify\_Concept}:[\mathtt{Type}:[\mathtt{Floor\_Tile}]]]) = \\ \quad = \mathit{Req}\end{array}$$

In the case of the example, this calculation is easy, since only one norm applies. However, in more realistic scenarios, dozens of norms could apply, making formal support indispensable. Thus, we know now that John is required to be involved in the modification of product category modifications.

4 Conclusions

This paper had the goal to investigate how multiple stakeholders can be supported in their discussion about the development of a cross-enterprise information system. As a testing case, the set-up of an business-to-business electronic broker system for the Dutch construction industry has been considered. The case study revealed that a crucial item for creating the broker system is the *fair* co-development of product ontologies for the market segment. The product ontologies constrain the way how companies can market their products in the common broker system.

The result of our paper is a formal model of ontology co-development guided by explicit social norms. The involvement of stakeholders in the co-development is stated in formal norms which are employed in determining which

[7] A detailed description of the core process ontology underlying the RENISYS type hierarchy is given in [1].

stakeholders to include a a discussion process when a change to a product ontology element is proposed. The formal approach to the *selection* of the stakeholders to get involved in a particular concept definition discussion is a decisive factor for the success of cross-enterprise systems. Any such system which fails to convince its members that they are treated fairly is likely to fail.

The group of members (i.e. stakeholders) have complete control of which stakeholders to involve in particular concept definition processes by so-called composition norms. Thus, different concept types can be redefined by different subsets of stakeholders. To accomplish this, using an automated legitimate user-driven specification approach such as the RENISYS method has several advantages, such as:

- All stakeholders who should be involved will be automatically invited, without the risk of being left out.
- Participants who should not be involved in a particular change process can be kept out, for example by defining explicit *forbidden compositions* norms.
- A better subdivision of redefinition responsibilities can be made, thus reducing the workload and promoting involvement.

Existing approaches fall short in considering the explicit representation of social norms. That is acceptable in a single-enterprise environment where the hierarchical organization of the enterprise supplies the legitimacy of changes. In the case of cross-enterprise systems, this is no longer true. Naturally, such systems can also be developed without the RENISYS method. Then, the rights of the members are enforced by norms that are informally agreed upon. We claim however, that automation as offered by RENISYS makes decision processes more complete, consistent, and efficient and makes cross-enterprise information systems more trustworthy. To test these claims, a robust version of a prototype RENISYS tool is being developed, which can be used in projects such as the MEMO project. Whether RENISYS really improves the decision processes is an open question that has to be answered by subsequent empirical research. Our approach demands extra work in the definition of the social norms which might be rejected by users. It may also be that the issue of granularity of the social norms prevents the construction of the norms. An earlier case study from the area of electronic journal makes us however confident that the approach is suitable here as well.

References

[1] A. De Moor. *Empowering the User: A Method for the Legitimate User-Driven Specification of Network Information Systems*. PhD thesis, Tilburg University, The Netherlands, 1999.

[2] EC-Gate. EC-Gate home page. Online www.ecgate.nl, Jan 2000.

[3] M. Fox and M. Grüninger. Ontologies for enterprise integration. In M. Brodie, M. Jarke, and M. Papazoglou, editors, *Proceedings 2nd Conference on Cooperative Information Systems, 1994, Toronto, Canada*, pages 82–89, 1994.

[4] HCP-Edibouw. Branchemodel elektronische communicatie. HCP-EDIBOUW, Driebergen, Netherlands, November 1998.

[5] M. Jarke, R. Gallersdörfer, M. Jeusfeld, and M. Staudt. ConceptBase - a deductive object base for meta data management. *Journal of Intelligent Information Systems*, 4(2):167–192, March 1995.

[6] M. Jeusfeld. Business data structures for B2B commerce. Technical Report ITRS-006, INFOLAB, Tilburg University, The Netherlands, 2000. Online infolab.kub.nl/pub/.

[7] M. Jeusfeld and M. Jarke. Enterprise integration by market-driven schema evolution. *International Journal on Cuncurrent Engineering Research and Applications*, 4(2), September 1996.

[8] G. Keller, A. Lietschulte, and T. Curran. Business engineering mit den r/3-referenzmodellen. In A.-W. Scheer and M. Nüttgens, editors, *Electronic Business Engineering, Proceedings Intl. Conf. Wirtschaftsinformatik, Saarbrücken, Germany*, pages 397–423. Physica-Verlag, 1999.

[9] E. Kendall, M. Malkoun, and C. Jiang. A methodology for developing agent based systems for enterprose integration. In *Proceedings IFIG Working Conference on Architectures for Enterprise Integration, November 1995, Queensland, Australia*, 1995.

[10] MEMO-Consortium. MEMO home page. Online www.abnamro.com/memo/, Feb 2000.

[11] W. Robinson and V. Volkov. Supporting the negotiation life cycle. *Communications of the ACM*, 41(5):95–102, 1998.

[12] A.-W. Scheer. Aris. In P. Bernus, K. Mertins, and G. Schmidt, editors, *Handbook on Architectures of Information Systems*, pages 541–565. Springer-Verlag, 1998.

[13] J. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, 1984.

[14] A. Steuten and V. Van Reijswoud. The interpretation of business communication: The application of functional grammar and the transaction process model. In *LAP96*, pages 77–91, 1996.

[15] Trespa. Trespa home page. Online www.trespa.nl, Jan 2000.

[16] Velux. Velux home page. Online www.velux.nl, Jan 2000.

[17] R. Vidgen. Stakeholders, soft systems and technology: Separation and mediation in the analysis of information system requirements. *Information Systems Journal*, 7:21–46, 1997.

[18] H. Weigand, A. De Moor, and W. Van den Heuvel. Supporting the evolution of workflow patterns for virtual communities. In *HICSS'33*, 2000.

[19] H. Weigand, E. Verharen, and F. Dignum. Integrated semantics for information and communication systems. In R. Meersman and L. Mark, editors, *Database Application Semantics*. Chapman & Hall, 1997.