SPINNING A CORPORATE SEMANTIC WEB FOR PRODUCT ENGINEERING

Uwe Keller, Stijn Heymans

Digitial Enterprise Research Institute (DERI), University of Innsbruck, Technikerstrasse 21a, Innsbruck, Austria

Alois Reitbauer

ProFactor Produktionsforschungs GmbH, Im Stadtgut A2, Steyr-Gleink, Austria

Michael Neswal

ATENSOR Engineering and Technology Systems GmbH & CoKG, Im Stadtgut A2, Steyr-Gleink, Austria

Abstract: We propose a novel approach towards a generic engineering support environment that combines Semantic Web technologies, Semantic Desktops, Group Memory Systems and Multi-Agent Systems to overcome the problems of current systems for supporting individuals in collaborative engineering processes. In particular, we aim at transferring the principles underlying the Semantic Web into an enterprise. This paper provides a motivation for our research, identifies requirements and outlines the proposed solution.

1 INTRODUCTION

Engineering of products is a complex and ubiquitous task; it occurs within enterprises across all industries and faces products of increasing complexity. As shown in Fig. 1, already for the process of designing a single product, involved artifacts are numerous and highly inter-dependent. Information about design elements (or parts) is created and managed by different people using their favorite tool. People can take very different roles in the process (e.g. mechanical engineer, software developer, product designer, project manager, financial controller etc.) and therefore generate information about parts of the product that reflect their specific roles. The result is an extensive use of various data formats to represent certain information about the single parts of the product. At the same time, there is neither a single uniform data format that could suitably serve all the people involved in the process, nor is it possible to impose the use of such a uniform, generic format since the used tools are not natively designed to serve an integrated, process-wide environment. Hence, traditional database systems are not applicable for data management in this setting (Franklin et al., 2005).

Design decisions for certain parts of a product must be based on all relevant information. This in-
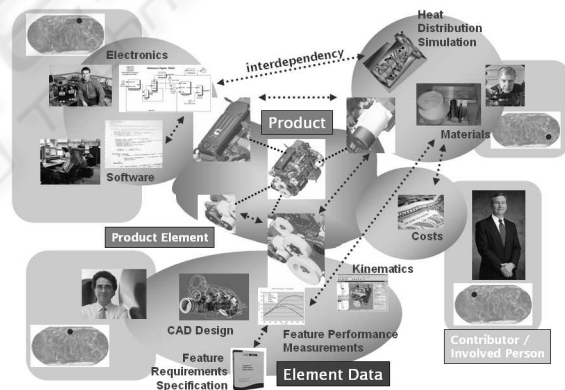

Figure 1: A Typical Scenario when Designing a Product.

cludes information about all other parts that are in some way directly or indirectly related to the element under consideration. A decision on how an element has to be designed might affect (or be constrained by) all these other parts: for instance, if there is a need to change the diameter and material of a screw for cost reasons, all regions around the respective drill wholes where this screw is supposed to be used as well as all physical connections to other parts of the product have to be checked to see if there would be any subsequent problems (e.g. geometric, mechanical, physical ones etc.) with the new de-

sign. Design in this respect can be understood as high-dimensional multi-parameter optimization problem with constraints stemming from different areas (such as mechanical or or geometric restrictions). The product design team altogether tries to find an optimal solution to the given constraints. Search in the solution space is *knowledge-driven* and requires *intensive communication and collaboration*:

Information in general is distributed across different legacy systems that are rarely integrated or even inter-operable. Therefore, it is not easy for individuals to identify and to get needed information for a specific task. It is often necessary to talk to a series of people to eventually find out the desired information. Furthermore, design decisions can often not be taken by individuals alone, but must be taken by a group of experts. Typically, those people are not located at the same site, but work at different places, even in different time zones. Face-to-face communication is therefore not always possible and time-lags to identify relevant information or to discuss design issues for resolving a concrete problem may arise.

In fact, the overall situation is even more complicated in a real-world scenario, since companies typically invent various products and maintain product variants or versions (in re-design processes) over time. In particular, when re-designing a product, a lot of documentation on the single parts of the product is already available and needs to be taken into account. Especially, at this stage it is very valuable to know about former design decisions. Therefore, efficient management and support of the knowledge needs of people involved in a design process is crucial and getting more and more importance nowadays.

For knowledge-intensive, collaborative processes, information systems used within the industry today are insufficient for providing efficient and holistic support during a process as well as across different processes. Especially, it is up to individuals to (manually) create and manage their own information space to increase the efficiency of their work process. This space is neither explicitly represented (and hence not documented) nor managed by a dedicated system. It cannot be communicated and not be shared with others. Furthermore, support systems are usually passive, i.e. the user has to repeatedly ask for information, rather than getting notified about relevant information whenever it is available. This shifts the burden of information gathering to the end-user rather than into a "smart" information space and therefore absorbs a lot of useful attention of the end-user from the actual problem solving process. This situation is depicted in Fig. 2.

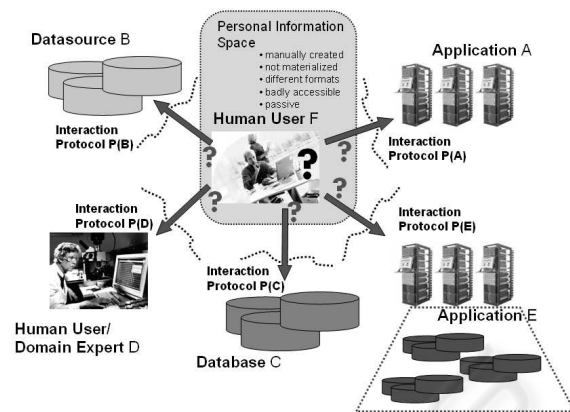For this reason, we propose a novel approach to-

Figure 2: The Perspective of an Individual in the Process.

wards developing a generic engineering support environment that combines ideas underlying the Semantic Web, Semantic Desktops, Group Memory Systems and Multi-Agent Systems into a coherent environment to overcome the problems and improves the current situation for individuals involved in engineering process (see Fig. 2) towards the situation shown in Fig. 4. Whereas in this paper, we restrict ourselves on the presentation of the problem context, the identification of requirements for a desired support environment and the derivation of the conceptual ingredients of our solution, more technical details and a discussion on economic relevance can be found in (Keller et al., 2007).

2 MAIN CHALLENGES

From our discussion of the problem context in Section 1, we derive the following main challenges that engineering support environments should properly address: (i) **Explicit representation and management of information spaces**: individuals in the engineering process are knowledge workers. They work within their personal information space. A prerequisite for finding relevant information inside this space is the availability of an explicit representation of such a space. Explicit representation and management by a dedicated tool is therefore necessary and desirable. (ii) **Organization and access of information spaces according to suitable mental models (or world views)**: Information spaces can get large and finding the right information can become a bottleneck. Creating some structure within the space, allows to deal with the information more efficiently. But suitable structures depend on the individuals using a space. Efficient access of information is facilitated by structures that correspond to an individual's "natural" un-

derstanding of the domain (i.e. world view or domain model). Hence, (a) domain models are needed to structure information spaces, (b) the structure of personal information spaces should be customizable for individuals and (c) these models should be machine-processable to enable machine-support when accessing information.

(iii) **Support for the maintenance of various information spaces & distinction between private and shared spaces**: Information spaces of individuals are distinct and should not interfere (if not explicitly stated). Usually, there is information that is specific to an individual and not intended to be shared with others. Further, there is a lot of information that is meant to be shared with others for collaborative tasks. Individuals are therefore interested in including such shared information into their personal information space whenever it is relevant to them, i.e. on a demand basis. Hence, (a) information spaces can be shared or private, and (b) structured by controlled overlapping for collaboration purposes.

(iv) **Pro-active support**: information spaces can become very large over time. Therefore, even when being structured properly, an individual might not be able to keep track of the (global) state of his information space at a specific moment in time and therefore miss relevant information that comes into being (i.e. from a shared source) or only detect the information unnecessarily late. Instead of requiring an individual to explicitly ask the system for the current state of the information space at any point in time (to explore the space), a pro-active environment is desirable, that informs individuals about (small) changes of the current state of their information space. Such notifications should be optional (i.e. applied on a on-demand basis), selective and specified by the client. We consider this elementary form of pro-activeness as a minimal requirement. Beyond it, more complex forms of pro-activeness might be useful. In general, this will require detailed task-specific (and perhaps company or product specific knowledge) which is not reusable across companies. Therefore, the respective pro-active support functionality will then have to be encapsulated inside a dedicated software component (instead of being a generic functionality supported by the basic system).

(v) **Task-centered support**: At any moment in time during the process, individuals are concerned with a particular, well-defined task. This implies, that only parts of the overall personal (and global) information space are relevant. Therefore, it is desirable that all irrelevant information is faded out, since (a) individuals have limited cognitive capacity, (b) they should focus as intensively as possible on their task (not being dis-

tracted by other things at that moment). Furthermore, restricting the overall information space to a relevant subset at a particular moment in time facilitate scalability of algorithms that are concerned with accessing the domain models (e.g. ontology reasoning). Therefore, tasks should be used as a central means for scoping within the system, whenever this is possible.

(vii) The system should **support the documentation of design decisions** (that have been taken during a design process throughout time) to generate a corporate memory: in fact, this aspect has been identified as one of the most useful features of support for engineering tasks in (Gruber and Russell, 1994).

(viii) The **system can not replace existing tools**, but rather has to act as a glue between existing ones: people are used to work with specific tools. Often these tools are highly specialized to support a particular task (but not the overall process) and have been developed with a lot of intellectual and monetary effort. Hence, **various forms of legacy data** (e.g existing documents, entries in databases) **have to be integrated** into information spaces. These data sources often have fundamentally different nature (e.g. file systems vs. databases), which must be accessed by different interaction protocols and are distributed over various the world.

3 RELATED WORK

In the following, we briefly overview some related work and concepts that have inspired our approach and that have similar goals.

Semantic Desktops. Semantic Desktops (Sauermann et al., 2005) are a first step towards bringing the Semantic Web on a personal computer: the underlying idea is to use of ontologies, metadata annotations, and Semantic Web protocols on desktop computers to enable integration of desktop applications and the Web, and therefore a much more focused and integrated personal information management as well as focused information distribution and collaboration on the Web beyond sending emails. Recently, (Decker and Frank, 2004) envisioned the concept of a *Networked* or *Social Semantic Desktops* as the ultimate result of a convergence of three very active recent research fields: Peer-to-Peer Computing, Social Networking and Semantic Web. Essentially, Social Semantic Desktops extend the idea of Semantic Desktops by a strong collaborative dimension based on a highly decentralized infrastructure.

Our proposal can be seen as a specific instantiation and extension of Semantic Desktops, where we (i) add a task-specific dimension (e.g. support is strictly

based on an explicit representation of a current task to be performed), (ii) target at a specific domain and therefore aim at providing more domain-specific support (e.g. an engineering rational framework), and (iii) use a specific set of technologies to realize the system, that do not have to be the standard technologies used within the Semantic Desktop Community (e.g. using a Semantic Wiki as a communication / collaboration channel).

Organizational Memory Systems. Organizational memory systems (OMS) (Abecker et al., 1998; Dieng, 2000) have been proposed as a general approach to enable integration of dispersed and unstructured organizational knowledge by enhancing its access, dissemination and reuse amongst the members of an organization and the organization's information systems. More recently, (Vasconcelos et al., 2000) proposed to narrow organizational memory systems to the concept of *group memory systems* (GMS): systems to manage heterogeneous and distributed knowledge embedded in business process activities. Therefore, GMS can be understood as a specialized version of OMS that deal with knowledge at a smaller scale. The system proposed in (Vasconcelos et al., 2000) focuses on capturing and sharing knowledge about internal competencies, in particular on human "knowledge sources". They are understood as information systems in the classical sense, i.e. passive knowledge stores. In contrast, the environment proposed here (although sharing many objectives with OMS and GMS) has a strong pro-active nature (besides providing a passive corporate memory). A dedicated agent layer allows to add desired domain-specific support functionality. Furthermore, we aim at explicit task-oriented support for individuals in an engineering process. At the same time, our use case requires the instantiation of the agent layer for a specific functionality, namely an engineering rational framework.

In summary, our system in a way tries to combine the ideas underlying all these approaches into a coherent system and to tailor them towards the specific desiderata of product engineering. The idea of scoping is central here. Therefore, it embodies a novel approach towards support for engineering processes that is based on solid existing work.

4 PROPOSED SOLUTION

So far, we have argued that current tools to support engineering processes are not sufficient, identified main reasons and extracted the major challenges that need to be addressed by an environment that aims at providing holistic support within and across engineering

processes. In this section, we outline our proposal for an environment, that combines the ideas underlying Semantic Web, Semantic Desktops, Group Memory Systems, and Agent-based Systems in order to tackle the challenges discussed in Sec. 2. In this paper, we focus on motivating and identifying the main elements and conceptual ideas underlying our system (Sec. 4.1) and discuss their combination to a conceptual system architecture (Sec. 4.2).

4.1 Design Principles

A review of the requirements from Sec. 2, leads to the following conceptual elements of a holistic support environment: Requirements (i) *Explicit representation and management of information spaces* and (ii) *Organization and access of information spaces according to suitable mental models (or world views)* ((a) and (c)) motivate the use of ontologies (Staab and Studer, 2004), as formalized, machine-processable representation of domain models. Ontologies are the fundamental semantic data model underlying the system. Requirement (ii)(b) can be achieved considering ontology networks (i.e. a collection of ontologies that are interrelated by suitable mappings, e.g. (Haase and Motik, 2005)) instead of a single upper-level ontology. The formulation of mappings is up to the user as well as dedicated administrators of the corporate knowledge space. Clearly, suitable tools should simplify the definition of mappings as much as possible (e.g. by means of simple GUI gestures, such as the Drag-and-Drop metaphor). Clients always operate on their personal information space through their own ontology. Already pre-defined (default) ontologies in an enterprise could serve as a simplifying starting point for the development of a personal ontology. Requirement (iii) *Support for the maintenance of various information spaces & distinction between private and shared spaces* motivates the development of a dedicated ontology management component, which is capable of identifying ontologies, of managing multiple ontologies and of sharing ontologies amongst a group of people. The latter might involve concurrent changes of the ontology. Conceptually, the component provides multiple spaces as semantic communication and collaboration channels between all other components of the system. Fine-grained access-control and the support of (partial) imports of ontologies (via mappings) together allow to construct hierarchies of (partially) overlapping information spaces and together cover the desiderata (iii)(a) and (b). We address requirement (iv) *Pro-active support* by an ontology management component that provides publish-subscribe capabilities on changes of ontologies. The most relevant changes are the creation or deletion of

instances in an ontology, however, changes at the terminological level should be supported too (e.g. the change of the description of a particular class). Furthermore, the use of an agent-infrastructure allows to encapsulated and integrate more specific and domain-dependent support functionality into the environment. An example of such a functionality is an engineering rational framework. Requirement (v) *Task-centered support* is covered by a dedicated domain model: the explicit representation of possible tasks of individuals within an engineering process and the relation of these tasks to people, to resources and other activities. For any individual, a specification of the current task should be available at any point in time. Subsequently, all components and agents in the environment will exploit this information. Requirement (vii) *documentation of design decisions* is resolved on the agent layer by means of a suitable set of agents that collaboratively implement an engineering rational framework. Finally, requirement (viii) *No replacement of existing tools* and *integration of existing legacy systems* is taken into account by the use of an agent-based architecture and a specific type of adapter components (called *Semantic Facades*) that provide a semantic perspective on the data residing in a legacy data source. Semantic Facades for standard data sources (such as relation databases, file systems or web servers) and data formats (e.g. PDF, spreadsheet formats, XML, RDF) must be provided by the environment. All actors outside of the environment (e.g. human users, legacy systems or remote support environments) are integrated via dedicated agents. Agents exploit ontologies for communication. Communication between agents can happen in two ways: synchronously by means of a standard message-based paradigm, as well as asynchronously (and decoupled in time and reference) by means of a (semantic) space. Instead of hiding knowledge spaces of individuals inside an agent at the agent layer, we strive for an explicit dedicated ontology management component (based on a distributed infrastructure) that can be accessed outside the agent-framework as well.

4.2 System Architecture

Conceptually, the architecture of the proposed generic support system consists of three layers (see Fig. 3), that successively abstract from the actual technical infrastructure available to support individuals today:

(i) The *Web Layer* addresses the problem of how to retrieve data objects stores in physical data sources. It implements the Web identification scheme for naming resources and allows to retrieve data elements given their URL. Technical knowledge about underlying (heterogeneous) interaction protocols, distribu-
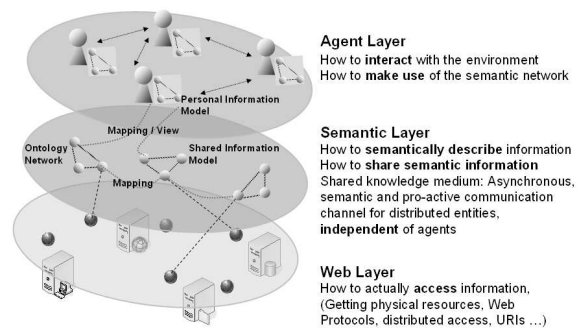

Figure 3: Conceptual Layering of the Support Environment.

tion of resources and alike are hidden from the upper layers. (ii) The *Semantic Layer* essentially provides a semantic perspective (i.e. a semantic network) on the collection of resources that are available inside the environment. Semantic networks are rooted in ontologies. This layer provides capabilities to manage (multiple) information spaces (i.e. ontologies) and exploits Semantic Facades to integrate semantic networks from legacy data sources. The semantic layer can be seen as a fundamental infrastructure component for semantic-enabled collaboration based on a very simple conceptual abstraction, namely ontologies. (iii) The *Agent Layer* provides advanced services to the end-user. Here, we specifically aim at the implementation of an engineering framework, i.e. a software system that allows people to document and efficiently retrieve design decisions that have been taken during engineering processers. Besides the engineering rational framework, many further services are desirable in practice. They can be integrated in a modular way at the agent layer. A concrete example is a system for the management of corporate competency (Vasconcelos et al., 2000).

5 CONCLUSIONS

To enhance the support for collaborative, concurrent engineering processes in distributed teams, we propose a novel approach to create engineering support systems. Our approach combines concepts underlying the Semantic Web, Semantic Desktops, Group Memory Systems and Agents into a coherent engineering support environment. A particularly important use case for the proposed system is the realization of an agent-based *engineering rationale framework*, i.e. a system that helps engineers to document and retrieve design decisions for parts of a product. The proposed environment is able to provide a more effective and holistic support for distributed, concurrent engineering processes, with the following features: (i)
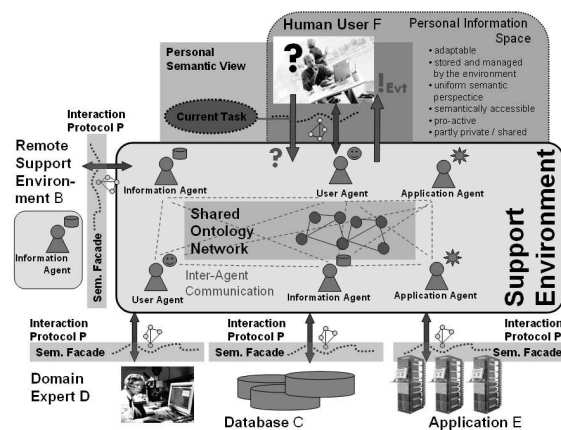
Figure 4: Using a Semantic-based Environment for Dedicated Human-centered Support.

decreased complexity of finding and interacting with the right information, (ii) semantic-based organization and access of information spaces, (iii) personalized and customizable organization of the information space for every individual, (iv) decreased complexity of identifying the right people, (v) pro-active and (vi) task-centered support for individuals during the process, (vii) documentation of design decisions throughout time to generate a corporate memory. The proposed system changes the situation that individuals have to face within the process from the one that is shown in Fig. 2 to the situation that is illustrated in Fig. 4. Conceptually, the system transfers the principles underlying the Semantic Web architecture into enterprises. One fundamental principle thereby is to not replace the existing tools and systems, but rather to build an agent-based environment which integrates them gradually and allows stepwise enrichment of the overall process support beyond the capabilities of the single tools. Economically, our approach is expected to provide as measurable advantages (i) a reduction of development time and the time-to-market (especially for re-design processes, which become increasingly important), (ii) reduction of costs, while at the same time providing (iii) improved process quality and quality of the resulting design, and (iv) the construction and management of a cooperate knowledge base. The latter aspect is especially interesting for companies to address the problem of loosing knowledge of experienced people leaving the company, since a lot of knowledge is not stored explicitly in any IT system inside a company. These information spaces exist mostly in people's minds today. For new staff a cooperate knowledge base reduces the effort required to become familiar with projects and all relevant information for their daily business. We implemented our ideas in a first prototype. The prototype is functionally not complete yet. It focuses on the use

and sharing of ontologies between users. Concerning the legacy data sources, the prototype is capable of integrating information from various file systems. Future work is the extension of the prototype to support event subscription and notification mechanisms for information spaces, extended support of important legacy data sources (e.g. various popular document formats, relational databases), as well as the implementation of an agent-based engineering rationale framework.

ACKNOWLEDGEMENTS

REFERENCES

Abecker, A., Bernardi, A., Hinkelmann, K., Kühn, O., and Sintek, M. (1998). Toward a technology for organizational memories. *IEEE Intell. Systems*, 13(3):40–48.

Decker, S. and Frank, M. R. (2004). The Networked Semantic Desktop. In *WWW Workshop on Applic. Design, Develop. and Implem. Issues in the Semantic Web*.

Dieng, R. (2000). Guest editor's introduction: Knowledge management and the internet. *IEEE Intelligent Systems*, 15(3):14–17.

Franklin, M. J., Halevy, A. Y., and Maier, D. (2005). From Databases to Dataspaces: a New Abstraction for Information Management. *SIGMOD Record*, 34(4):27–33.

Gruber, T. R. and Russell, D. M. (1994). Generative design rationale: Beyond the record and replay paradigm.

Haase, P. and Motik, B. (2005). A mapping system for the integration of OWL-DL ontologies. In *Proceed. of the Intern. ACM Workshop on Interoperability of Heterogen. Information Systems (IHIS'05)*, pages 9–16.

Keller, U., Reitbauer, A., and Mungenast, R. (2007). Principles of SEnSEful Engineering Support Systems. TechReport at http://www.uwekeller.net/publications.html DERI TR 2007-01-07.

Sauermann, L., Bernardi, A., and Dengel, A. (2005). Overview and outlook on the semantic desktop. In *Proceedings of the Workshop on The Semantic Desktop at the ISWC 2005 Conference*.

Staab, S. and Studer, R., editors (2004). *Handbook on Ontologies*. International Handbooks on Information Systems. Springer.

Vasconcelos, J., Kimble, C., Gouveia, F., and Kudenko, D. (2000). A Group Memory System for Corporate Knowledge Management: An Ontological Approach. In *Proceedings of 1st European Conference on Knowledge Management (ECKM)*, pages 91–99.