

FINDING MANUFACTURING EXPERTISE USING ONTOLOGIES AND COOPERATIVE AGENTS

Olga Nabuco¹, Mauro F. Koyama¹, Francisco Edeneziano D. Pereira¹ and Khalil Drira²

¹ *Centro de Pesquisas Renato Archer, 13081-970 Campinas, SP, Brazil;* ² *LAAS-CNRS 7, Av. Colonel Roche 31077 Toulouse, France*

Abstract: The new product development process, in its early phases, congregates knowledge from diverse domains and sources, spread among specialists that usually are not aware of each other's research interests. This work presents a tool designed to make the establishment of new communities or teams easier. The tool organizes their informal vocabulary into formal domain categories, in order to correlate them, and advises its users of potential networks they could belong to. An approach based on ontologies and machine learning is used to structure and make knowledge acquisition / retrieval, together with agents that cooperate and coordinate the access to the knowledge. The tool automatically updates its knowledge base relieving the user from tasks she/he is unfamiliar with. Initial experiments demonstrated the potential of this kind of tool. A test case is being developed encompassing members of a scientific network in the manufacturing area. Results will be analyzed envisaging its broader use and application to different knowledge domains.

Key words: knowledge sharing tools, ontology, knowledge acquisition, agents, manufacturing recommendation system.

1. INTRODUCTION

The design and conception phase of a product congregates knowledge from various domains. This knowledge is spread among people who have their own jargon and are possibly not aware of each other's vocabulary, even if they belong to the same technological chain. In order to truly cooperate

people need to share a vocabulary, even minimal, so that they can exchange information that have the same meaning for each other.

This work addresses the cooperation among specialists mainly in the early phase of peer recognition. It introduces Sharing Engineering Information and Knowledge - SHEIK Tool - a system based on software agents that proactively retrieves and presents contextualized information based on user's vocabulary.

The approach was to develop a distributed architecture that permits to find manufacturing expertise by means of specialists recommendation in selected domains of manufacturing, in a way that can lead to establishment of new development teams. The early phases of product development were targeted, as it is the time a new community can emerge.

This article is organized as follows: section 2 discusses cooperation in manufacturing virtual enterprise; section 3 presents a functional description of SHEIK system; section 4 describes an initial prototype evaluation; section 5 shows the importance of knowledge and ontologies to recommendation systems; section 6 concludes the paper.

2. COOPERATION IN A MANUFACTURING VIRTUAL ENTERPRISE

Manufacturing is an area that is facing drastic transformations due especially to the production paradigm change: from mass to customized production. Such change entails transformations in the productive processes, that are increasingly automated, as well as in the organizational processes that are changing from functional to business process view.

Product development is a key process for industrial enterprises. A survey carried out by Harmsen et al. (2000) in 513 Danish industrial companies showed that managers ranked the perceived importance of product development competence in fourth place in a 10 items list; sales, market responsiveness, production management were ranked number one to three.

Concurrent Engineering (CE) is the current paradigm for product development. Its purpose is to overlap product development phases. Willaert et al. (1998) discuss CE and shows that it can not lead to maximum benefits if it were considered alone, being necessary to consider it in a wider context called Collaborative Engineering.

Early phases of product development are those ones in which concepts are being studied, searching for fitness for the development problem in hand. In these phases the specialists may not have a complete idea for solutions, they are screening concepts against their criteria. This is a moment when

specialists profit from their social network, asking questions as who knows about the product and who knows others that know about it.

3. SHARING ENGINEERING INFORMATION AND KNOWLEDGE - SHEIK - OVERVIEW

SHEIK is a tool for enhancing cooperation among members (potential or actual) of a community of practice (competence network) in a particular knowledge domain. The idea is that specialists could connect themselves to the network (local or global) and subscribe to a recommendation service the output of which would be specialists' profiles, containing their technical areas of interest, mapped from an ontology related to their main knowledge domain.

The purpose of this tool is to ease community establishment. Users can recognize potential peers from answers delivered from system and they can contact these potential peers to verify their skills and will to cooperate.

After communities are set up, SHEIK will serve as a maintenance tool and as a warehouse of knowledge and specialists' profiles. Also, it is a known fact that information needs of specialists vary in time so SHEIK could aid by recommending new profiles as needed.

3.1 Sheik requirements

The SHEIK requirements were presented in (Nabuco et al., 2001), together with a requirements modeling framework which is shortly summarized as follows: it encompasses three generic requirements (or dimensions) – cooperation, coordination and communication – and two views – system (technological) view and user (human) view.

Knowledge is the central element in all dimensions. **Cooperation** among team's members needs those members to share part of their knowledge. **Coordination** mutually identifies producers and consumers of information/knowledge. **Communication** locates and clarifies the path among producers and consumers.

Views are related to gather physical system requirements. **System view** considers technological and economical constraints, allowing a feasible solution to be defined; **user view** regards how to facilitate interaction between user and system.

To meet these requirements it was designed a technological solution that combines multi-agent systems and ontologies. Characteristics as system distribution and autonomy to act on behalf of its users were some of the

reasons that led to the multi-agent solution. Enhancing the multi-agent systems capabilities ontologies were added, in which the core recommender system is based on and, machine-learning algorithms, performing the “intelligent” side of the agents. The Sheik multi-agent system has three functional elements: Multi-Agent System (MAS), Sheik meta-model, and Intelligent Modules.

3.2 Sheik multi-agent system

As shown by Figure 1, the Multi-Agent System is formed by two types of agents: Sheik - an interface agent that runs in each user’s machine, monitoring user activities in a timely basis, and Erudite - a matchmaker agent that interfaces with a Knowledge Base (KB) whose knowledge is related to a specific (sub) domain of Manufacturing.

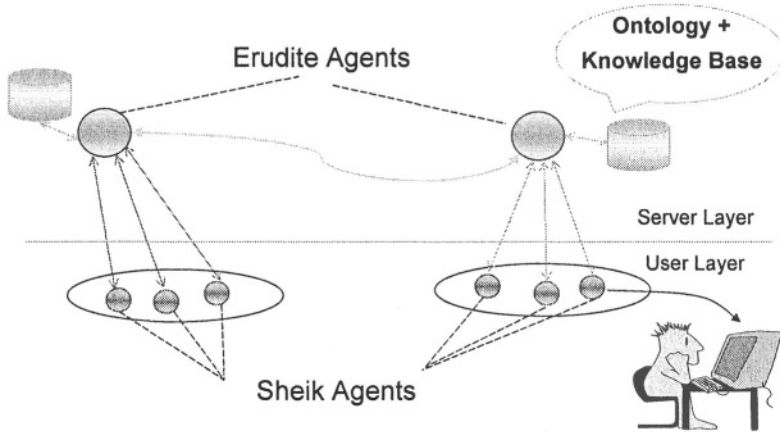


Figure 1. SHEIK Architecture

SHEIK architecture (see Figure 1) is organized into two layers: User Layer, composed of a colony of Sheik agents each one running in a user’s machine and Server Layer, composed of a federation of Erudite agents running on server machines.

Each Sheik agent automatically updates users’ profiles information relative to its user’s knowledge domain, asks the Erudite agent for a recommendation containing specialists in domains related (or coincident) to his own domain. After Sheik had received a response, it e-mails its content to his user.

Each Erudite agent is related to a particular knowledge domain and contains an associated ontology relating person’s profiles to a knowledge domain. Erudite agents, upon receiving a request from one Sheik agent,

perform a search in its local KB, and if they find data about the referred knowledge domains they respond immediately to the Sheik agent. In case data is not found locally by an Erudite agent, it queries other Erudite agents to get pertinent data and if successful it forwards the data (knowledge domains and profiles) to the requesting Sheik agent.

The system was designed considering an open system perspective, so it could be easily replicated, and capable of executing in heterogeneous environments. As part of this philosophy the reutilization of existing tools and code was emphasized, as a manner of shortening development cycle and collaborating in others efforts in disciplines related to the project.

It was used a MAS prototyping tool (MASPT), created at “Renato Archer” Research Center – CenPRA. MASPT is based on a MAS Development Methodology (MASDM) that was developed for dealing with the modeling and development of architectures for shop floor control (Koyama, 2001). Results from the requirements phase (Nabuco et al., 2001), based on the GAIA methodology for agent-oriented analysis and design (Wooldridge et al., 1999), were used as a starting point for the design, that proceeded using MASDM.

Today, a set of systems for developing MAS exists (www.fipa.org/resources/livesystems.html). At the time the detailed design (2002) was done system’s designers considered existent systems inadequate for the project development because designers required the projected system to be lightweight and low-cost. Some of the existing systems required either a RMI / CORBA server or compliance to FIPA or MASIF specifications, adding unnecessary complexity to the system. Also features for Java code generation, that could ease rapid prototyping were lacking or were not effective at that time, so designers opted for using CenPRA’s MASPT that generates a MAS using *ad hoc* performatives declared by users and exchanged as Java serialized objects in a TCP-IP environment.

3.3 Sheik intelligent modules and meta-model

Intelligent Modules are used to accomplish required agents’ functionalities. Figure 2 shows the basic models for both the Erudite agent and the Sheik agent. KA means Knowledge Acquisition and KB means Knowledge Base, the Intelligent Modules for each agent. This approach of making intelligent modules explicit provides great flexibility to the system that can be upgraded by changing them.

Sheik agent was designed to act on behalf of its user, acquiring data on his/her interests and consulting the Erudite agent to verify who are the possible peers (specialists with compatible / similar profiles) to consult, aiming at cooperation. To perform Sheik’s KA functionality, the Kea tool

(Witten et al., 1999) was selected. It is based on a bayesian networks approach for keyword extraction and analysis and is available in Java.

The Erudite agent is responsible for answering requests from Sheik agents on behalf of their users. It must use its KB functionality. A KB based on ontology was selected; such an approach allows one to relate specialist's profiles (characterized by keywords, and working areas) to knowledge domains (for instance, manufacturing, software engineering, artificial intelligence).

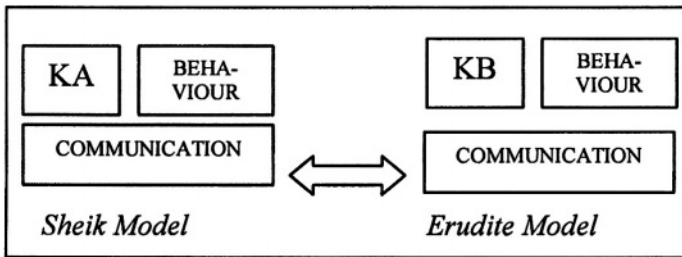


Figure 2. Sheik & Erudite models

To implement KB functionality, the Protégé System (Gennari et al., 2001) was selected as it is both an ontology editor and a knowledge base editor. It is available in Java as well and has been extended, having a large community of users and plug-ins developed (<http://protege.stanford.edu>).

An ontology can be considered as having been formed as a combination of a taxonomy and a set of rules. Such a set can be conceptualized through a knowledge meta-model.

A meta-model, named Sheik meta-model, that relates a manufacturing taxonomy (IMTI, 2000) and a conceptual class named Profiles (person's profiles) was developed to be used in SHEIK. Those elements were entered into Protégé, Figure 3, and the resulting knowledge base was populated with initial data concerning mainly profiles from researchers working on the SHEIK project.

4. PROTOTYPE EVALUATION

The SHEIK prototype, using Kea tool and Protégé KB, was tested in a research environment composed of two laboratories: one at CenPRA and the other at Campinas State University - Unicamp, School of Mechanical Engineering. These laboratories are located in the same city but physically separated (10 Km). They are connected to the Internet and have Manufacturing as their main research area.

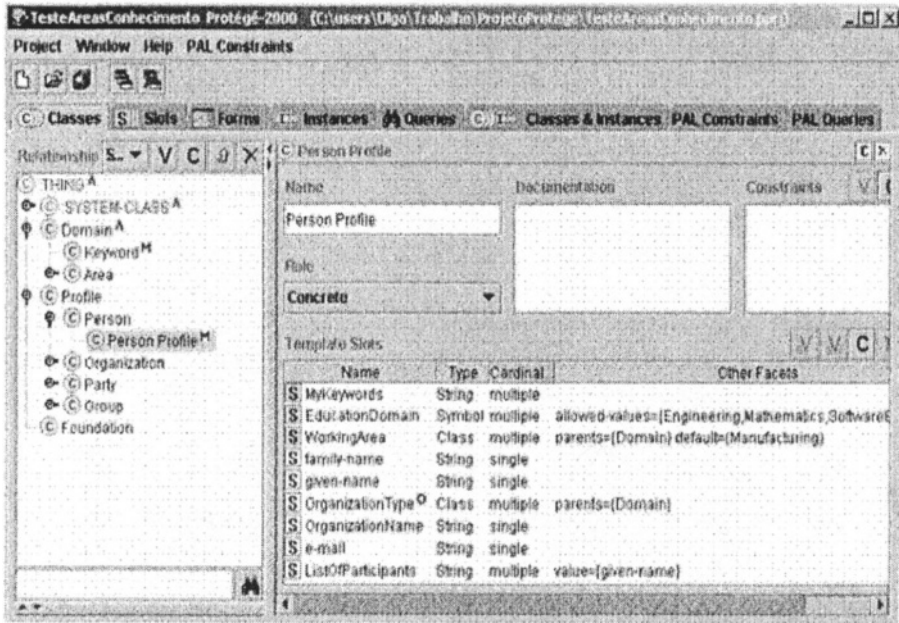


Figure 3. Sheik meta-model mapped into Protégé.

An Erudite agent was installed at Unicamp, containing the related Protégé system and acting as a server in the Internet. Researchers’ profiles were manually populated in order to avoid the cold-start problem.

Sheik agents were installed at CenPRA. They gathered local user’s data by means of their KA functionality. Local presentation of the data was done using an optional (Sheik agents do not use it) external module, as shown in Figure 4. It was tested the Sheik’s ability to request related researcher’s profiles from Erudite.

Figure 4 (left side) shows an user profile, defined locally at user’s machine. In the future, it can be used to automatically feed the KB when the user subscribes the Erudite services, meaning the system could be capable of fully automatic start-up and operation. Figure 4 (right side) shows the output of Kea tool. These data are combined with local user’s profile and sent to Erudite agent.

Two aspects were verified in the evaluation: if the system met the specifications and the users’ acceptance.

The specification constrained Sheik agent to be light in terms of memory use and memory occupation, profiling the users and “learning” with them. Learning means that the agent searches new keywords increasing its accuracy as time passes by. Communication with Erudite agent was made using an *ad hoc* protocol. Erudite used the Protégé Axiomatic Language

application program interface to query the database about keywords and the people related to them. Considering the specifications it was concluded that the system worked properly, according to the expected behavior.

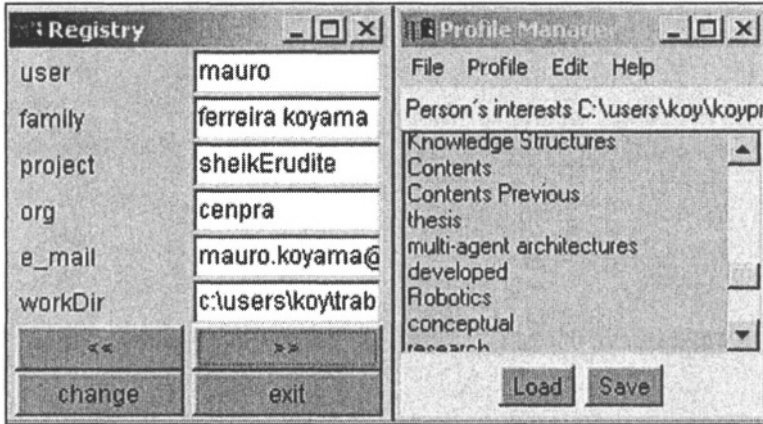


Figure 4. Sheik's user interface.

Regarding users' acceptance the system was considered well-behaved. Users disliked Kea's characteristic of acquiring keywords only from text files. is unable to gather information from other sources than text files. After this evaluation, other format conversions (PDF to TXT, PS to TXT and DOC to TXT) were added to the Sheik agent.

To date no system has been found in the literature with similar objectives and structure and having enough published information to make an effective benchmarking effort possible. Table 1 summarizes systems listed in section 5. The table shows that SHEIK is a recommendation system based on multi-agent system, knowledge sharing and ontologies that recommends specialists with compatible profiles.

5. RELATED SYSTEMS

During an innovation process, people inside teams need to exchange information and share knowledge. There is an evolution in the way knowledge exchange is viewed (McLure and Faraj, 2000), from "knowledge as object" to "knowledge embedded in people" and finally as "knowledge embedded in the community" view.

Recommendation Systems search and retrieve information according to users' needs; they can be specialized on users profile or on the user's instantaneous interests (e.g., when users browse the web). They can be

classified as collaborative filtering with and without content analysis and knowledge sharing. This kind of system is usually based on artificial intelligence techniques such as multi-agent systems, machine learning algorithms and ontologies. These techniques can be used separately or combined in different ways to find information and deliver it to the people who need it.

Table 1. Recommendation Systems comparison

	Collaborative filtering		Knowledge sharing	Multi-agent	Recommends	Uses ontology
	With content analysis	Without content analysis				
Yenta	y			y	Scientific papers	
GroupLens	y				Usenet news	
Referral Web	y			y	Scientists	
Phoaks		y			Scientists	y
OntoShare			y		Opinions	y
QuickStep			y	y	Web pages	y
OntoCoPI			y		Communities of practice	y
Sheik			y	y	Similar profiles	y

Collaborative filtering with content analysis is based on information from trusted people which the system recognizes and also recommends. Examples of this kind of system are: The GroupLens Research Project (Konstan, 1997), The ReferralWeb (Kautz, 1997) and Yenta (Foner, 1999).

Collaborative filtering without content analysis examines the meta-information and classifies it according to the user's current context. It can recommend multimedia information that otherwise could be too complex to be analyzed. PHOAKS (Terveen et al., 1997) is an example of a system that associates scientific profiles by chaining bibliographic references with no comprehension of the article's contents.

Shared Knowledge Systems aim to construct a knowledge base according to a previous specialists know-how or best practice memories. Ontologies are well-suited for knowledge sharing as they offer a formal base for describing terminology in a knowledge domain (Gruber, 1993; McGuinness, 2001). Examples of recommendation systems that use ontologies are: OntoShare (Davies et al., 2002), OntoCoPI - Ontology Communities of

Practice Identifier (Alani et al., 2002) and the Quickstep system (Middleton et al., 2002).

6. CONCLUSION

SHEIK integrates many concepts from distributed systems and artificial intelligence having as target product development in manufacturing that is a strongly team-based process. The literature review revealed a lack of systems as SHEIK which supports teams establishment in the early phases of product development and with the cross-functional profiles of the specialists involved in this process.

In the initial evaluation, despite the fact that the set of users was small, it was verified that the SHEIK prototype worked satisfactorily. Technically, the concept's feasibility was verified. An agreement is in course between SHEIK developers and one Brazilian non-governmental organization, in the manufacturing area, encompassing around 100 researchers, to setup an scalable experiment.

The problem of interoperation with other systems was not studied at an application level, but this was not a priority as, to date, there are no standards defining ontological descriptions of manufacturing systems and subsystems and as the idea was toward team establishment where the principle of minimal ontological commitment applies.

The underlying MAS can be upgraded to be compatible with emerging agents communication protocols standards, if necessary, preserving the intelligent modules (knowledge base and knowledge acquisition).

Validations of the used tools were done. Both Kea tool from Waikato University, New Zealand, and Protégé from Stanford University School of Medicine, USA, proved useful.

It is expected that the new test-case being set will enable unveiling new and existing problems and will permit to enhance SHEIK and put it to reliable use in the manufacturing.

REFERENCES

- Alani, H., O'Hara, K. and Shadbolt, N. ONTOCOPI: methods and tools for identifying communities of practice' Proceedings of 2002 World Computer Congress, Intelligent Information Processing Stream (Montreal) 2002.
- Davies, J., Dukes, A. and Stonkus, A. OntoShare: Using ontologies for knowledge sharing. Proceedings of the www2002 International Workshop on the Semantic Web. Hawaii, May 7, 2002.

- Foner, L. N.. Political Artifacts and Personal Privacy: The Yenta Multi-Agent Distributed Matchmaking System. Boston: Media Arts and Sciences, Massachusetts Institute of Technology, 1999.129p. PhD Thesis.
- J. Gennari, M. A. Musen, R. W. Ferguson, W. E. Grosso, M. Crubézy, H. Eriksson, N. F. Noy. The Evolution of Protégé: An Environment for Knowledge-Based Systems Development. *International Journal of Human-Computer Studies*, in press, <http://faculty.washington.edu/gennari/#pubs>.
- Gruber, T. R., Towards principles for the design of ontologies used for knowledge sharing, *International Journal of Human-Computer Studies*, v.43 n.5-6, p. 907-928, Nov./Dec. 1995
- Harmsen, H., Grunert, K. G., and Bove, K., Company Competencies as a Network: the Role of Product Development, *Journal of Production Innovation Management*, pp 194-207, v. 17, 2000
- IMTI - Integrated Manufacturing Technology Initiative. 21st Century, Manufacturing Taxonomy A Framework for Manufacturing Technology Knowledge Management. V1.0. Dec. 2000, <http://www.IMTI21.org>.
- Kautz, H., Selman, B., Shah, M.. The hidden web. *American association for artificial intelligence*. Summer 1997. p 27-36.
- Konstan, Joseph A., Miller, B. N., Hellocker, J.L., Gordon, L. P., Riedl, J. Applying collaborative filtering to Usenet News. *Communications of ACM*. Vol 40. N°3. March 1997.
- Koyama, Mauro F., Arquitetura de Supervisão e Controle de Chão de Fábrica baseada em componentes genéricos, School of Mechanical Engineering, State University of Campinas, Brazil, PhD Thesis, 2001.
- McGuinness, D. L., Ontologies and online commerce. *IEEE Intelligent Systems*. V. 16. N° 1 p. 8-14.2001.
- Middleton, S., De Roure, D., Shadbolt, N. Capturing Knowledge of user preferences: ontologies in recommender systems. *Proceeding of First International Conference on Knowledge Capture, KCAP2001*.
- McLure M. W., Faraj, S., "It is what one does": why people participate and help others in electronic communities of practice. *Journal of Strategic Information Systems*. N°. 9, p. 155-173.2000.
- Nabuco, O., Drira, K., Dantas E., A Layered Design Model for Knowledge and Information Sharing Cooperative Systems, in *Proceedings of tenth IEEE International Workshops of Enabling Technologies: Infrastructure for Collaborative Enterprises*, p. 305-310, MIT, Cambridge, Massachusetts, USA, June 2001.
- Terveen, L., Hill, W., Amento, B., McDonald, D., Creter, J. PHOAKS: A system for sharing recommendations. *Communications of the ACM*. Vol. 40, N. 3. March 1997.
- Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C. and Nevill-Manning, C.G., "Kea: practical automatic keyphrase extraction, " *Proc Digital Libraries '99*. 254-265, Berkeley, CA, USA, August 1999.
- Willaert, Stephan S.A., Graaf, Rob de, and Minderhoud, S., Collaborative engineering: A case study of Concurrent Engineering in a wider context, *Journal of Engineering and Technology Management*, JET-M, v. 15, pp. 87-109, 1998
- Wooldridge, M., Jennings, N., Kinny, D. A methodology for agent-oriented analysis and design. *Proc. 3rd Int. Conf. On Autonomous Agents*, pp 68-76, Seattle, WA, 1999.