KMi Technical Report

KMi–TR–94

KNOWLEDGE MEDIA

KMi

INSTITUTE

Knowledge Management in a Distributed Organisation

Martin Dzbor, Jan Paralic, Marek Paralic

Knowledge Media Institute
The Open University

May 2000

1 KNOWLEDGE MANAGEMENT IN A DISTRIBUTED ORGANISATION

Dzbor, M.[§] – Paralic, J.[‡] – Paralic, M.[†]

[§] *Knowledge Media Institute, The Open University, United Kingdom (M.Dzbor@open.ac.uk)*
[‡] *Dept. of Cybernetics & AI, Technical University of Kosice, Slovakia (Paralic@tuke.sk)*
[†] *Dept. of Computer Science, Technical University of Kosice, Slovakia (ParalicM@tuke.sk)*

It has become very important for advanced organisations to make the best use of information gathered from various document sources inside companies and from external sources like the Internet. There are many technologies under development, which address knowledge discovery. On the other hand, there is a lack of efficient technologies focused on organising and sharing of existing knowledge. In this paper we introduce the research in scope of KnowWeb[i] (EC funded project). We focus our attention on two important issues – (i) how to capture tacit, contextual knowledge that is connected to the documents and (ii) how to support knowledge management in geographically distributed organisations through up-to-date communication and AI technologies

1. INTRODUCTION

Knowledge has become an important asset in a modern enterprise. Straightforward and fast access to knowledge possessed by its employees may significantly influence the competitiveness of an enterprise. A new research discipline – *knowledge management*, has emerged when knowledge was recognised as an asset. Knowledge management (KM) generally deals with several phases of knowledge life cycle (Abecker, Bernardi *et al.*, 1998): identification, acquisition, development, dissemination, use and preservation of enterprise's knowledge. In this paper we look briefly at the identification and acquisition. The emphasis is given on the preservation and further development of knowledge especially in distributed organisations, which possess some specifics that must be born in mind when designing a KM system.

The companies are keen to design and implement an effective KM system because, "*an organisation's knowledge walks out of the door every night – and it might never come back.*"[ii] Employees and knowledge gained during their engagement are often lost in the dynamic business environment. Those who stay with the company are often unaware of critical resources that remain hidden in the vast repositories. Most knowledge in a company is thus forgotten in a relatively short time after it was invented. A large portion of this knowledge is tacit and typically stored in the employees' minds but in one way or another, it is connected with various documents published and archived within a company.

Organisational learning and personnel training are two of possible remedies to an acute problem with the preservation of knowledge in a modern company. However, the globalisation and geographical distribution of many companies bring into focus

additional issues that do not appear so visibly in a centralised enterprise. In addition to knowledge preservation in the individual 'branch' offices, in distributed companies the need emerges to share and re-use knowledge between the geographical locations. As O'Leary claims, KM is difficult when participants are in the same location; geographical dispersion only makes it harder because many traditional knowledge flows (e.g. face to face communication) are simply not present (O'Leary, 1998). Selected issues with KM are discussed in sections 2 and 6 (implementation).

In this paper we present our research on knowledge management tools funded by European Commission under Esprit Programme, project 29065 'KnowWeb – Web in Support of KM in Company'. Our tool is has three distinguishing features:

- Conceptual knowledge modelling clearly structuring knowledge bases,
- Model-based knowledge and information retrieval, and
- Agent support for distributed knowledge bases and repositories.

Knowledge modelling approach assumes there is a conceptual description of an application domain, so called *domain model*, describing such concepts as company's activities, projects, customers, employees, etc., as well as relations among them. The main purpose of the model is to use a set of pre-defined concepts to express knowledge in concern to a particular information source. More details about knowledge modelling background are provided in sections 4 and 5.

2. SELECTED ISSUES WITH KNOWLEDGE MANAGEMENT

Traditional approach to KM suggests a central knowledge repository, whence employees may retrieve important knowledge and make good decisions (Bank, 1996). This approach is correct when connecting people with knowledge; however, is not exhaustive because it does not link a chunk of knowledge to other relevant chunks. In addition, these KM systems often store only raw data without particular context. Although data from databases and from various textual documents is a prime source of knowledge, neither of them is sufficient when standalone (O'Leary, 1998).

Documents[iii] are a common source of information; they partially provide the *content* of knowledge base entries. But content reflects only a fraction of knowledge that is encoded in a document, unfortunately a very small one. More important fraction is provided by the *context* in which particular documents were created. Context may be expressed and codified in form of links between different documents and between the documents and other information sources. And this is the major source of troubles with KM. While the well-defined content may be easily captured and encoded, the context remains unnoticed, 'well-hidden' inside employees' minds.

The main purpose of any KM tool is thus capturing and dissemination of both aspects of knowledge. In our approach we assume this process consists of two steps; first, the content is prepared in a common text processor and second, the author annotates the chunks of content to the selected 'reference points'. To make the contextual links unambiguous throughout a company *all contributors to a 'corporate memory' must inevitably share* the *reference points*. If this requirement is neglected in the development and implementation of KM system, the resulting knowledge repository will be created 'ad-hoc', which may introduce undesired ambiguity. Eventually, it may bring more harms than benefits.

We mentioned above that geographical distribution of modern companies makes the management of corporate knowledge difficult because many traditional communication channels do not exist among knowledge sources and recipients. The rapid development of network technologies introduces new media that have a potential to lift the 'communication barriers' in geographically dispersed companies. Internet and corporate intranets are at the base of many successful AI techniques that may be advantageously utilised in KM. World Wide Web itself is even considered as one of the most powerful KM tools (O'Leary, 1998). Intelligent and mobile agents are nowadays able to act 'on behalf' of their owners, manipulate vast information repositories, *communicate* with each other and *exchange* valuable knowledge assets.

3. ORGANISATIONAL MEMORIES AS TOOLS FOR KM

Individuals represent a prime location where intellectual resources of an organisation are located (the others are organisational culture, organisational structures, etc). Organisations are indeed concerned with preserving knowledge within an organisational setting. Mechanism of *organisational memory* (OM) aims at capturing and storing knowledge within an organisation and simultaneously making it retrievable and reusable. At a macro-level, organisational memory corresponds to organisational knowledge with persistence. At a micro-level, information seeking by some organisational member within an organisational context can be considered as the process of finding the right 'piece' of information or knowledge from an organisational memory in the right time (Ackerman, 1994).

OM is essentially a *hybrid technology* (Abecker, Bernardi *et al.*, 1998) that involves both human participants and various supporting technologies. The role of humans in this partnership is to solve ill-defined problems and derive innovative and creative solutions using knowledge that is preserved and made available by supporting technologies. Such an idea supersedes traditional expert and decision support systems that were acting more autonomously but were lacking many essential human capabilities in problem solving. OM is thus a typical example of an 'intelligent' assisting computer- (network-) based tool.

In co-operation with the industrial partners in KnowWeb project we identified several requirements that are crucial (unfortunately not sufficient) for the success of OM and intelligent knowledge management systems in practice. Our experience roughly corresponds to that of other researchers (Kühn & Abecker, 1997) and enhances it further:

- *Preservation and systematic organisation of knowledge from different sources* – knowledge in a company is typically dispersed in many sources (e.g. text-based reports, contacts, best practices, e-mails, etc.). OM should preserve this knowledge, simplify access to it, and allow easy, user-friendly navigation even for non-specialists in up-to-date communication technologies.
- *Low initial costs for OM introduction* – companies do not want to spend too much time identifying and formalising their knowledge, rather they prefer techniques that are exploitable 'immediately' and use available knowledge sources (e.g. contacts, e-mails, etc.) without mundane transformation and other 'preprocessing burden'. The existing company staff should manage the development of OM and its use, if possible.

- *Integration with existing environments and tools* – the idea of KM system to be introduced in a company was accepted easier when OM was designed as an enhancement of an existing technology within a company (for instance, tailored contact management tool or Point-Of-Sale applications in our case).
- *Active retrieval and presentation of relevant knowledge* – access to knowledge stored inside OM should be simple; it is advantageous if the retrieval of knowledge chunk is made 'pro-actively', not waiting for a prompt from the user. To make retrieval more user-friendly it is not sufficient to make it keyword-based; approximate and analogous search techniques were used in case of KnowWeb.
- *Dissemination and distributed OM management* – centralised approach to OM is limited in the amount of services provided; often setting higher staffing demands. This is particularly visible in small enterprises with a few employees (as in our case) where a dedicated 'knowledge manager' in each branch may come quite costly. OM should mirror the organisation itself, i.e. being distributed and using available networks such as Internet instead of dedicated phone lines.

4. MODEL-BASED APPROACH TO KM AND OM

An approach addressing the requirements from the previous section developed in the KnowWeb project is introduced below. Earlier in the document the need for a clear and unambiguous 'reference point' was emphasised if a company wants to capture not only the content of the documents but also their context. Effective capture and re-use of *tacit* (M.Polanyi in Sweiby, 1999), contextual knowledge may be achieved utilising a well-structured '*common and shared vocabulary*'. Such a vocabulary is known as a *common ontology* (Gruber, 1993; Chandrasekaran *et al.*, 1999) – another recently emerged AI technology addressing the issue of knowledge sharing.

Ontology is seen as a set of definitions of context-specific knowledge representation primitives consisting of domain-dependent classes, relations, functions, and object constants. A popular view on ontologies is one that understands them as representation vocabularies specialised to a problem domain or subject matter. For instance, let us consider an organisation doing planning. Plans are composed of steps, which in different cases form slightly different sequences with domain- and task-specific dependencies. If one wants to use the same plan, for a different purpose, s/he has to fully understand the meaning of each step, each notion, relation or dependency in the current plan before re-using and adapting it. If there is a consensus on what particular 'terms' in a description of a plan mean, one can adapt the plan for his/her particular purposes. S/he is thus able to re-use and share knowledge, which is encoded in the specific plan document thanks to understanding the plan description.

The above-mentioned common understanding of terms used in a particular domain (their meaning may be domain-specific) results in the definition of *concepts*. Concepts differ in their character and role: *classes*, *relations, functions* or *procedures*, *objects*, *variables* or *constants*. These primitive constructs can be represented differently in different applications but have the same meaning for human participants in OM – i.e. when one wants to communicate with somebody else, s/he can do it using constructs from the ontology of shared concepts. The concepts usually create a complicated structure through various relations but even a complex structure covers only a narrow world (e.g. a company and its business).

A set of concepts together with attributes and relations (hierarchical and other) among them forms a *domain model* that is used for two primary purposes: (i) annotation of documents and (ii) retrieval of annotated documents. In this paper only the former one is discussed more in-depth in the following section, which refers to a model shown in Figure 1 as a small sample of a company-specific domain model.
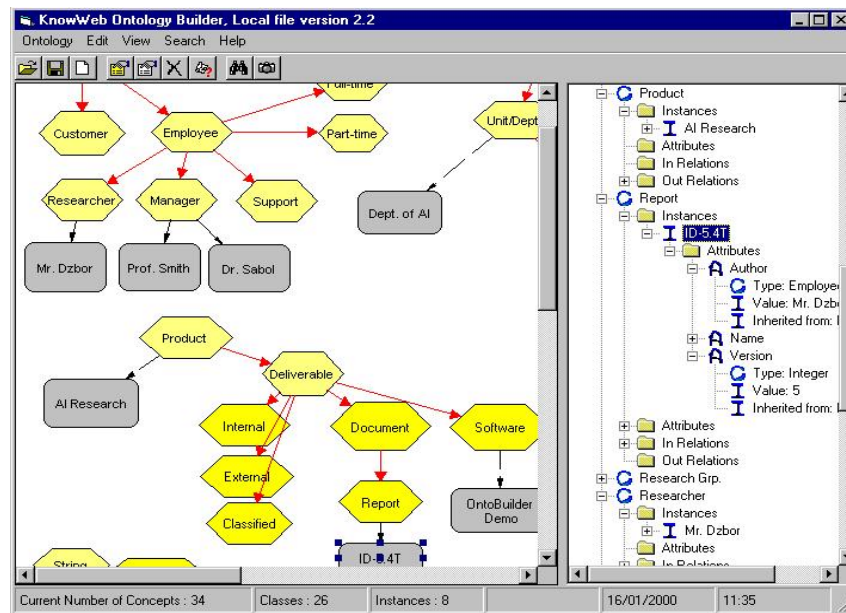

Figure 1 – Sample of a domain model showing two representation forms.

5. MODEL-BASED CONCEPTUALISATION OF DOCUMENT CONTENT

The KnowWeb tool enables authors to store their background knowledge together with the documents attaching the relevant concepts from ontologies – i.e. documents are stored with their context. Context can be attached to the document as a whole or to a specific part(s) called text fragment, that is a continuous part of text in a document (e.g. a sentence, paragraph). Currently, KnowWeb tool is able to process MS Word documents where no restrictions are given on text fragments and HTML documents where text fragment must not cross HTML tags. To cope with the documents the system provides a set of tools differing in their functionality but together supporting users to manage knowledge in a company easily and user-friendly.

First, in order to put a document to OM, it is necessary to attach context knowledge to it in the form of a *conceptual description* (CD) of a document. By conceptual description is meant a set of links between a document (marked text fragments) and concepts in a domain model. CD enables to refer to knowledge contained in the document in the future and make it easily accessible. The links can be created manually by selecting a text fragment and linking it to the domain model. Linking can be done directly to ontology concepts or to a template containing a given set of concepts. Association links are of 'many-to-many' type; i.e. it is legal to link a document (text fragment) to several concepts and contrary, a concept to more documents.

When a document with its description is ready (after manual or semi-automated linking inside the KnowWeb or after receiving a document from outside), both can be incorporated into OM represented by a KnowWeb server. Another possibility is to store a conceptual description of a document without storing the actual document (it is located elsewhere in a distributed company or on Internet). If an author is not satisfied with the conceptual description of a document in OM, s/he may modify it and subsequently upload the modified versions of document and/or description.

The aim of storing a document in OM is to access the right knowledge in right time or situation. To express requirements on documents to be retrieved from OM, the user has to formulate a query. For a query formulation s/he can use concepts (or their attributes) from the domain model. Concepts can be structured using various (e.g. logical) operators. In general, the concepts specified in the query are used to search conceptual descriptions of the documents, not the documents themselves!

The *Conceptualisation tool (CT)* consists of three modules – *DocView*, *OntoView*, and *TemplateEditor* (see Figure 2). CT serves as an 'envelope' for the modules, and deals with two types of documents. First, it is possible to open a document (located in an underlying EDMS – web server, file system). This may be linked to the chosen concepts in the domain model and its copy stored in the local KnowWeb server. Two kinds of association links are relevant in this case:

- association links between the whole document and a specific concept (or more concepts derived by a template) from the domain model;
- association links between a marked part of the document (e.g. a paragraph) and one or more concepts from the domain model

A similar operation may be done for a document that is already stored in local KnowWeb server (it may be also partly linked to some concepts in domain model). Existing association links can be edited/removed and/or new links can be added.

Another type is an HTML document not to be stored locally (e.g. documents accessed via remote retrieval function can be located anywhere on Internet). Documents of this kind are known as *referenced documents*, because only their URLs are known and serve as references to them. For this source, only the association links may be defined between the whole document and concepts from domain model.

In order to create association links, the drag-and-drop functionality exists in *CT*, which is working in both directions, i.e. from *OntoView* (or *TemplateEditor*) to *DocView*, and in opposite direction. It means, the user 'grabs' a concept in *OntoView*, drags and drops it on a highlighted text fragment (or vice-versa).

The purpose of the *DocView* module is to preview the documents and highlight the text fragments linked to the domain model (to a specified concept). Module supports definition and modification to the annotation structure of a document kept in form of a set of text fragments. Assignment of specific attributes to the fragments is also possible. The analysis of pilot applications by our industrial partners (e.g. application in the retail sector) identified a need for 'automated' conceptualisation of documents with rigid structure (e.g. daily reports in a retail chain generated in each shop). This need is addressed by the 'quiet mode' of the *CT* operation –documents are automatically linked to the predefined concepts without any visual component.

TemplateEditor is a support tool for both 'quiet' and interactive modes of *CT* operation. Its purpose is to group concepts that are often used together when linking documents (or text fragments). Linking a document (text fragment) to a template means that association links from the document (text fragment) to all concepts that

appear after processing a particular template are generated. *TemplateEditor* provides means for defining and utilising 'macro' (or batch) operations on the documents.
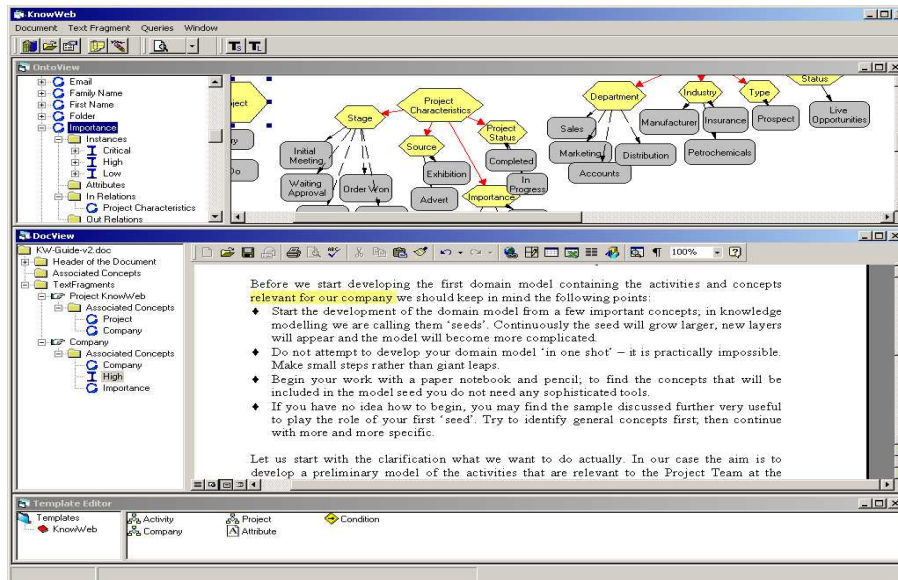

Figure 2 – KnowWeb client GUI for the conceptual description of documents.

Generally, a document has (possibly empty) header representing various properties of a document. For example, information about the document name, date and time of its creation, authors, comments, etc. The set of attributes applicable to a header definition is application-dependent. Some properties may be compulsory while others are optional. Templates support two kinds of concept reference:

- A concept *referenced directly* (only target concept must be given);
- A concept *referenced by a document property* (i.e. the name of a concept is given by the value of a document property in a header). The property name to be taken into account must be given, as well as the class under which the target concept with the same name will be looked for.

The former type is useful for both manual and automated modes. When using a template, a couple of association links can be defined in one step (for each concept in a template one association link is created). The latter type is tailored to the quiet mode of CT operation; i.e. automated linking. In this case, the target concepts are given indirectly and depend on the values of properties in the document header.

6. AGENT-BASED SUPPORT FOR DISTRIBUTED OM

As mentioned earlier, one of the most important requirements for success of OM is a support for the distributed environment of a company. The OM should be flexible enough to fit different company network settings and also should remain open to the external sources such as Internet. The state-of-the-art solution to the first problem lies in the utilisation of the distributed objects. The most popular architectures for

supporting distributed objects are *CORBA* (Common Object Request Broker Architecture) and *DCOM* (Distributed Component Object Model) (Pritchard, 1999).

Objects in these architectures capture the high-level (business) logic of a distributed application and are accessible for processes outside the computer running them. Present implementations of these standards use common communication protocols (e.g. TCP/IP). In the KnowWeb prototype a 3-tiered architecture is utilised:

- data source tier represented by the relational database,
- middle tier (called 'Server Front-End') that offers services for synchronised and secure access to document store and OM, and finally
- KnowWeb clients supporting the user-friendly usage of the toolkit.

The document store and OM functionality is mapped to the KnowWeb server. Server key services are (i) manipulation with the domain model state, (ii) provision of associations and documents, and (iii) retrieval in the document or ontology space. All these operations are implemented as manipulations upon relational databases.

Suggested scheme fulfils the flexibility requirement in the corporate intranet settings and partially addresses the openness to the external sources. KnowWeb server is usually separated on a local area network, contains a domain model and typically serves many clients. KnowWeb client can communicate with more than one KnowWeb server, potentially outside the intranet scope. However, the accessibility of KnowWeb servers outside the fast intranet networks assumes a reliable connection with guaranteed throughput, especially to upload new documents into OM and retrieve the existing ones. Unfortunately, this is not the case with most portable computers or branch offices connected via a low-speed dial-up connection.

To solve the major problem with the throughput the mobile agent-based solution was proposed. Mobile agent (MA) technology is a promising paradigm that supports application design for dynamically changeable, networked environment with distributed information and computation resources. The most significant features of MAs are their *autonomy* and *mobility* (Rothermel *et al.*, 1997). MAs are autonomous because of their capability to decide what locations in a computer network they visit and what actions they take once being there. This ability is given in the source code of MA (implicit behaviour) or by the agent's itinerary set dynamically (explicit order). A location is the basic environment for the execution of MAs' code and therefore an abstraction of an underlying computer network and operating system.

Usual benefits of MA are (i) reduced network load, (ii) overcome of the network latency, (iii) encapsulation of different protocols, (iv) asynchronous and autonomous execution, and (v) natural heterogeneity (Harrison *et al.*, 1995). We claim that robust and scaleable OM systems can profit from these features; especially in the companies with a world-wide and distributed structure. The distributed architecture of KnowWeb system and information flow are shown in Figure 3.

Dedicated mobile agents can do the most critical operations such as retrieval and gathering of documents at non-intranet servers. In the retrieval, client formulates the query locally and either sends a searching agent directly or demands its sending from a local KnowWeb server. The situation with gathering an existing document (accessible through other KnowWeb server) is similar. The main advantage is the client's possibility to get off-line during operation execution. By the next re-connect the ordered information is presented to the user. This approach brings savings on the communication between distributed KnowWeb servers in a distributed OM.

To work with MAs a *Mobile Agent Environment* (MAE) must be available on each concerned computer (client or server). MAE offers the following functionality: (i) creation of MA with a unique identity, (ii) transport of MA, (iii) sending a message to MA (possibly on another host), (iv) getting the status information about any MA, and (v) destroying MA. MAE used in KnowWeb project is an in-house design and implementation at the Technical University of Kosice (Paralic, 2000).
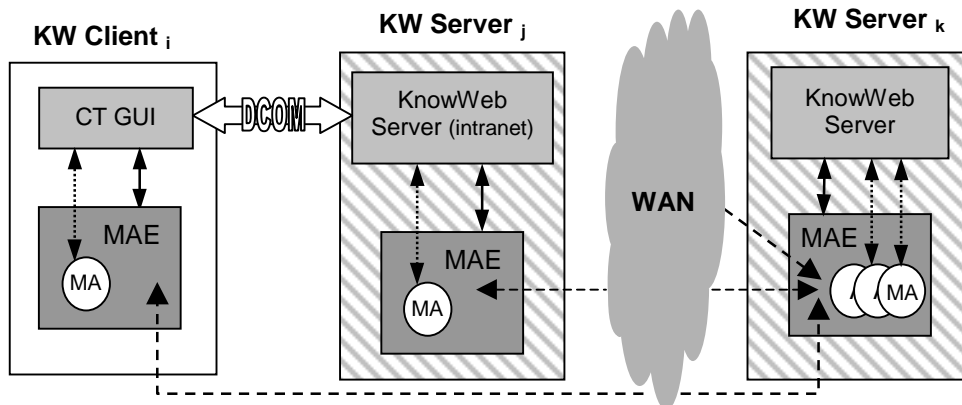

Figure 3 – Mobile-agent based distributed architecture of KnowWeb system.

7. CONCLUDING REMARKS

In this paper we present a few issues with the design and implementation of OM in distributed companies. Based on our experience from pilot applications realised in the KnowWeb research project we identified several requirements on the viability and success of OM in a real-world company. We stress the importance of two factors: (i) preservation and dissemination of knowledge content and context, and (ii) distributed and scaleable architecture for OM suitable for decentralised companies.

Our efforts are to design a tool capable to capture content of the documents and context, in which they were created. The resulting tool has access to a domain model describing conceptual terms and relationships specific for a particular company. Concepts in a domain model are used to capture the context of a document; a whole document or a chunk in HTML or Word document could be 'annotated'. Two basic modes of annotation are provided – *manual* (using tailored GUI) and *semi-automated* (using predefined templates). A sophisticated retrieval engine can retrieve the annotated documents. We achieved the sophistication by an approximate search, retrieving the documents based on their context rather than content or name. KnowWeb tool is able to work with MS Word documents directly, thus satisfying the demand to incorporate OM into existing corporate environment.

Mobile agents are used to satisfy the requirements on the management of a *distributed OM*. Agents search OM on the KnowWeb servers in other branches, retrieve and 'fetch' desired documents, update domain models and document annotation throughout the distributed network of a company. They cut the costs in working 'off-line'; clients only connect to the network to give an order and get results. In the meantime their work off-line avoids additional costs on maintaining the connection.

Generally, we may conclude that the tool developed within the KnowWeb project satisfies all requirements resulting from the pilot studies in heterogeneous enterprises as specified in section 3. Because of the heterogeneity of our pilot studies we believe that the requirements have a broader scope of validity and could be successfully used also in future research in the design and development of OM. Our next research should be oriented towards a higher flexibility and scalability of OM. It includes but is not limited to the investigation of methods for designing richly structured domain models and joining several more or less similar models in order to create a conceptualisation that is much closer to the reality. Other research strand may investigate capturing knowledge from other sources (e.g. documents not covered in our current research, databases, 'public' knowledge sources on Internet, etc.)

8. ACKNOWLEDGEMENTS

9. REFERENCES

Abecker, A.; Bernardi, A.; Hinkelmann, K.; Kühn, O. & Sintek, M. (1998): *Toward a Technology for Organizational Memories*, IEEE Intelligent Systems, **13**, May/June, p.40-48.

Ackerman, M.S. (1994): *Augmenting the Organizational Memory: A Field Study of Answer Garden*, Proceedings of the ACM Conference on Computer Supported Co-operative Work, p.243-252.

Bank, D. (1996): *Know It Alls*, The Wall Street Journal, Nov.18, p.R28.

Chandrasekaran, B.; Josephson, J.R. & Benjamins, V.R. (1999): *What Are Ontologies and Why Do We Need Them*, IEEE Intelligent Systems, **14**, p.20-26.

Gruber, T.R. (1993): *A Translation approach to Portable Ontology Specifications*, Knowledge Acquisition, **5**, No. 2, USA.

Harrison, C.G.; Chess, D.M. & Kershenbaum, A. (1995): *Mobile Agents: Are they a good idea?*, Research Report, IBM Research Division.

Kühn, O. & Abecker, A. (1997): *Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges*, Journal of Universal Computer Science, **3**(8), p.923-954.

Malhotra, Y. (1997): *Knowledge Management in Inquiring Organisations*, Proceedings of the 3rd Americas Conference on Information Systems, p.293-295.

O'Leary, D.E. (1998): *Knowledge Management Systems: Converting and Connecting*, IEEE Intelligent Systems, **13**, May/June, p.30-33.

Paralic, M. (2000): *Mobile agents based on Concurrent Constraint Programming*, to appear in the Joint Modular Languages Conference 2000, September, Zürich, Switzerland.

Pritchard, J. (1999): *COM and CORBA Side by Side: Architectures, Strategies, and Implementations*, Addison-Wesley.

Rothermel, K.; Hohl, F. & Radouniklis, N. (1997): *Mobile Agent Systems: What is Missing?*, Proc. of International Working Conference on Distributed Application and Interoperable Systems DAIS'97.

Sveiby, K.E. (1997): *Tacit Knowledge*, WWW page, URL http://www.sveiby.com.au/Polanyi.html, current at 7th April 1999.

---

[i] KnowWeb (Esprit #29065) – more information available on http://knowweb.fei.tuke.sk or upon request.

[ii] Kevin Abley (Cap Gemini) in OR Society Conference on Knowledge Management, London, Nov 1998.

[iii] The term 'document' throughout this paper refers to any 'storage place' based on text. We performed our experiments mostly with hypertext (HTML) and Microsoft® Word (DOC) documents.