On the convergence of core technologies for knowledge management and organisational memories: ontologies and experience factories

Yannis Kalfoglou[1]

Abstract. In this paper we argue for the convergence of core technologies for knowledge management and organisational memories. Most of the work reported in the literature regards knowledge management and organisational memories as intertwined areas. However, the technologies used to implement and support them are not treated in the same fashion. Usually, they are conceived, developed, and deployed separately. This prevents us from fully exploiting their strength. We identify two such technologies in this paper: ontologies and experience factories, originating in these different communities. We elaborate on their strengths as potential core technologies and show, through an example case, how their convergence could be of mutual benefit. We generalise the approach and speculate on the impact of such convergence in the broader context of knowledge management and organisational memories.

1 Motivation

Nowadays, we are witnessing increasing interest in knowledge management(hereafter, KM), and organisational memories(hereafter, OMs). The former refers to: "the formal management of knowledge for facilitating creation, access, and reuse of knowledge"([35]). The latter aims to provide the means for storing, retrieving and distributing knowledge from an organisation's repositories. The goal of OMs is to preserve knowledge whereas KM aims to develop and deploy knowledge. Clearly these two areas are intertwined and centred upon the enhancement of an organisation's competitiveness by improving the way it manages its knowledge.

There are various ways of implementing KM systems in an organisation as well as technologies for supporting OMs development. Ontologies and experience factories have been studied as the means to support KM and OMs, respectively. Ontologies were originally investigated in the Artificial Intelligence(hereafter, AI) community as a way to represent consensual knowledge about a domain of interest in reusable and sharable formats. On the other hand, experience factories, studied in the Software Engineering(hereafter, SE) community, have been explored as the means to promote and manage experiences collected throughout the life-cycle of a software project. Recently, both they have begun to applied in KM and OMs as core technologies to support their implementation. O'Leary investigates the role of ontologies in KM systems and argues that: "a KM system depends on ontologies to facilitate communication between its multiple users and links between multiple knowledge bases"([35]). Similarly, experience factories and their constituents, experience bases, have been applied as an OM to support exchange of experiences(see, for example, [6]).

Despite the fact that KM and OMs are intertwined the technologies that are employed to implement them are developed and applied separately. For example, most of applications of ontologies do not mention nor use any kind of organisational framework(i.e., in the form of an OM), but merely are deployed in order to achieve knowledge sharing and reuse benefits. In the same line, experience factories amid their successful implementations as OMs, do not employ particular types of ontologies, or even when they do these are often distant from the ontologies reported in the AI literature. This prevent us from exploiting the full potential of these two core technologies.

In this paper we argue for their convergence in order to exploit their potential more fully: knowledge sharing and reuse achieved by deploying ontologies could improve the development and deployment of KM, whereas the presence of an OM, in the form of an experience factory, could allow us better to organise, characterise, and store ontologies.

O'Leary argued in [36] for the need to "convert individual knowledge to group-available knowledge". This converting process requires knowledge sharing and collection in a form that can be generated and reused. The author refers to this as the *knowledge harvesting*, which "must identify knowledge that it desirable to share, worth converting, and usable by others"([36]). In this paper we paraphrase the above definition in the context of ontologies as follows:
Ontology harvesting must identify ontologies that are desirable to share, worth converting, and usable by others.

We argue that this *harvesting* process can be accomplished by employing OMs technologies, such as experience factories.

In section 2 we review the role of ontologies in KM and support our claim that they are a core technology to achieve KM goals which we identify in that section. Experience factories could effectively support OMs as we describe in section 3. These two technologies, however, should complement each other and we elaborate on an example case in section 4 where we employed experience factories to support ontologies deployment. We generalise the approach and speculate on the convergence of ontologies and experience factories in section 5. We conclude the paper in section 6.

2 Ontologies in support of KM

Many researchers are investigating the role of ontologies in KM. Recently, the reports of O'Leary in [36] and of Benjamins and colleagues in [11], summarise the contributions made and give us an

[1] University of Edinburgh, School of Artificial Intelligence, Division of Informatics, 80 South Bridge, Edinburgh, EH1 1HN, Scotland, email: yannisk@dai.ed.ac.uk

insight for their potential. O'Leary describes uses of ontologies in KM systems with example cases drawn from the private sector, in particular consulting firms([36]). In [11], Benjamins and colleagues argue for an ontology-based KM which results in an intelligent access to knowledge assets as they shown in an example case of KM in a virtual organisation.

We briefly recapitulate here the conclusions drawn in these reports and we also mention additional efforts from the ontology community to achieve effective KM. First, we identify areas of KM where ontologies are suitable for exploitation. In a review of KM systems([35]), O'Leary identified three factors that lead organisations to use KM. These were classified as *environmental pressures* imposed by the increasingly competitive global market place, *technological advancements* arising from recent developments in Internet technology, and the ability to *create valuable information* by converting individually available knowledge into group or organisationally available knowledge. Whereas the *environmental pressures* and *technological advancements* give an organisation the reason and the means to pursue KM activities, *creating valuable information* is the goal of KM.

The latter is achieved by the *converting and connecting* processes as identified in [36]. These are summarised as follows: convert (i)individual to group knowledge, (ii)data to knowledge, (iii)text to knowledge, and connect (iv)people to knowledge, (v)knowledge to knowledge, (vi)people to people, and (vii)knowledge to people. We argue that ontologies are present in most of these processes, either by playing a major role or by supplying the supporting infrastructure that helps an organisation to implement them. In the following paragraph we mention indicative examples from the ontology research literature to justify this claim. A complete listing is out of the scope of this paper which is to investigate the potential of convergence of KM and OMs technologies. However, we point the interested reader to field reviews in [44],[13],[22], and [25].

In particular, ontologies provide part of the infrastructure for conversion processes(i to iii as listed above) and play a major role in the connection activities(iv to vii as listed above). Conversion processes (i) seem to benefit more from the presence of ontologies as this is the underlying principle in their construction. Methodological([44]) and collaborative approaches([40],[10]) in ontology building, convert individual to group knowledge in the form of an ontology. Processes (ii) and (iii) use other AI technology like data and text mining techniques with ontologies being the guide to the 'right' data or text repository([16]). Ontologies are more active in the connecting processes. Process (iv) is concerned with the so called, 'pull' technology, which aims at pulling knowledge residing in vast repositories to people. The means which used to pull that knowledge are, mainly, search engines and intelligent agents. Examples of ontology use in this area are given in [32] and [24]. Process (v) actually highlights the main contribution of ontologies: enabling communication and interoperability between systems. The best way to cite indicative work here is to point to reviews and collections such as [44] and [23]. Process (vi) is not directly related to ontologies as it is more concerned with technological means such as Intranets. However, we should mention the work on collaboration and discussion aided by ontologies([39]). In contrast with process (iv), process (vii) is concerned with 'push' technology. Means to achieve this are designated systems that focus on content and push knowledge to the user instead of waiting for the user to pull out that knowledge. As in (iv), ontologies play a major role here since they are concerned with content and semantically enriched information. Example uses are described in [19] and [16].

We now return to the conclusions drawn in the two reports mentioned above([36] and [11]). In [36], O'Leary argued for a number of factors that drive the need for ontologies in KM. In brief, these are: formation of discussion groups on particular topics of interests, improved search capabilities by semantically-enriched query/answering facilities, filtering facilities to capture the desired knowledge, reusability of artefacts, and enabling communication between different systems and/or people by using an interlingua. In the other report([11]), Benjamins and colleagues argued that four basic activities of KM, namely, knowledge gathering, organisation and structuring, refinement, and distribution could be effectively supported by ontologies as they shown in their report with the use of ontologies in each of these activities.

3 Experience factories in support of OMs

While in the area of KM, as we saw in the previous section, ontologies have being acknowledged by many as a core technology to support KM, there is no such distinguishable technology in OMs. A reason for this may be the variety of activities that OMs are concerned with. For example, Reimer defines OMs as: "the means by which knowledge from the past is brought to bear on present activities, thus resulting in higher or lower levels of organisational effectiveness"([37]). Its purpose, Reimer continues, is to "ensure coherence between organisational units, coherence between cooperating companies, secure knowledge, and provide knowledge". There is no single technology that can accommodate all those activities and recent implementations combine several. For example, Abecker and colleagues identified potential research areas that contribute to OMs implementation([1]).

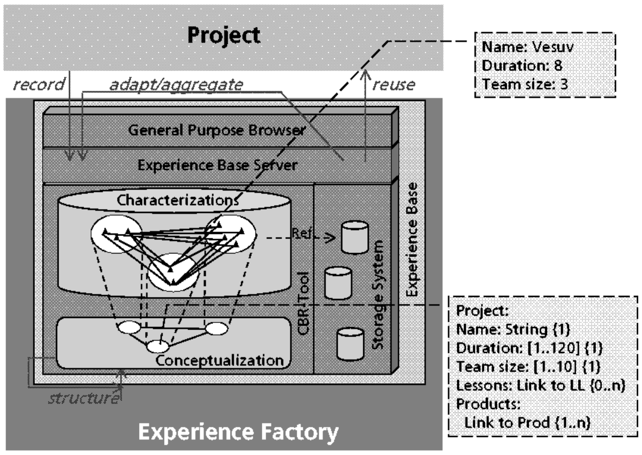

Figure 1. Main experience factory tasks and experience base architecture(adopted from [4]).

However, there is an area of research, stemming from the SE community, which is being promoted as a potential candidate for implementing OMs. This is the area of experience factories[2]. These were first investigated in the context of the TAME project([9]), and further generalised in the early nineties([8]) as the means to promote reuse of "all-kinds" of artefacts in an organisation. In figure 1, we illustrate

[2] In the SE literature, it is also referred to as "experience management" or "experienceware".

the main experience factory tasks and an experience base architecture, taken directly from [4]. The core of an experience factory is the experience base which acts as the OM. The key idea is to install an OM to support exchange of all kinds of experiences in the life cycle of a software project. The main focus of an experience factory is to support 'learning from experience' on a technology-independent organisational level. An experience factory stores the collected experiences in an experience base. In [5], it was argued that Case-Based Reasoning(hereafter, CBR) plays an important role in the experience factory paradigm. As CBR provides both the technology and a methodology for 'learning from experience' in the context of case-based knowledge systems([3]), it was natural to use it for implementing continuous learning in an experience factory style. In [4], Althoff and colleagues deployed CBR to manage the retrieval and adaptation of experiences in an experience factory.

The value of a repository of cases is also acknowledged by KM researchers as O'Leary writes with respect to the conversion of individual to group available knowledge: "Although individuals might have generic knowledge to contribute, case histories are particularly robust"([35]). Such repositories of cases are suitable for applying CBR techniques which provide the means for retrieval and adaptation.

As in KM systems, ontologies also found in OMs implementations, as for example in [2]. Reimer elaborates on a generic architecture for OM which "[. . .]provides content description that uses a domain ontology to refer to such knowledge areas as employee skills, company regulations, office tasks, organisational structure, products, marketing channels, etc."([37]). Liao and colleagues describe the use of ontologies for knowledge retrieval in OMs([31]). Recently, work has begun to integrate ontologies and experience factories, mainly in the SE community ([41]). In the next section, we argue for the benefits of such convergence via an example case in the area of ontology verification which we originally presented in [27].

4 Ontologies and Experience Factories

Most of research in ontologies is focussed on development issues and we have witnessed products and tools such as methodologies(i.e., [44], [20]) and guidelines([21]) for building ontologies, online environments([18],[40],[17]) for collaborative construction, agent-based systems([7]) for supporting the selection task, and generic ontologies([30]) to be used as foundations for domain-specific ones. However, there is a dearth of tools to support ontology deployment and maintenance.

To deploy ontologies correctly we need not only reuse-oriented tools and technologies, as for example in the HPKB([15]) and IBROW([19]) projects. It is also necessary to record and organise our experiences in having applied them in order to improve future ontology deployment. It is hard to gain this sort of experience from the literature because few cases of comprehensive ontology reuse and deployment on a large scale are reported([43], [12], [46]). Even those which are reported do not normally discuss the hidden assumptions and tradeoffs identified during testing. This section explains how OMs technologies, like experience factories, may address this issue.

We argue that experience factories can be useful in ontology development and deployment as a way of managing the experiences collected from various agents participating in ontology building and usage. This brings, crudely speaking, OMs technology into KM core technology - ontologies - which in turn has important advantages for the organisation that employs KM activities. We elaborate on these in section 5 while here we focus on a simple architecture we built to make the idea more concrete. The architecture, which is centred upon the notion of ontology verification, is given diagrammatically in figure 2.

The left-hand side of figure 2, depicts the task of verification. In particular, we are interested in verification of ontologies at the application level([28]). That is, we verify that ontological constructs are not misused by applications that adhere to an ontology. After applying our verification mechanism we accumulate, temporarily, the results in an experience base. These are code-testing results and we regard them as experiences. The experience base is then imported by an experiences editing tool which allows for further additions and modification of the description of existing experiences. It allows us to customise the experiences to the particular project as it provides a way of expressing information usually not obtainable through code-testing. We then select the experiences we want to validate and send them to a designated tool for verifying their correctness with respect to test results. This tool embodies the verification mechanism we deploy in the first step but here we apply it to verify the correctness of the results themselves. After the selected experiences have been validated we store them in the final experience base to be part of the experience factory.

This cycle can be repeated as many times as we wish in the same or other ontologies to collect and manage the knowledge accumulated during verification and testing. Ultimately, this will result in an experience factory of ontology verification and testing that can be deployed in similar projects in order to facilitate ontology use. It is important to mention that the similarity of projects to benefit from this sort of experience factory dictates the ability to transfer experiences. How easily can we identify similar projects in order to transfer previous experiences? An answer to this question is given by the ontology characterisation framework mentioned later in this section.

In [27] we applied this approach in an example case where we deployed and verified an existing ontology at the application level. The case is explained thoroughly in the aforementioned paper, while here we briefly describe by following the architecture given above. The ontology which we used, is the PhysSys ontology documented in [12]. It is a formal ontology based upon system dynamics theory and comprises of 7 different ontologies related via an inclusion lattice. We deployed two of these, the *mereology* and *topology* ontology in an exemplar application provided by the PhysSys developers themselves. We used the verification mechanism we invented([26]) to check the correctness of *mereological* and *topological* definitions used in the exemplar application. We devised and introduced artificial errors in the application in order to demonstrate the usage of the verification mechanism. Surprisingly, we also detected errors occurred in the original ontologies. We accumulated and verified all the discrepancies found, by storing them as experiences in the final experience base mentioned above. The size of such experience base varies in accordance with the project being applied. In our prototypical case we created a small experience base of a dozen of test cases.

Apart from this specific information on testing we also wanted to store additional information with regard to the ontologies used which will allow us to retrieve and adapt the experiences in different settings. We didn't dealt closely with retrieval and adaptation techniques as we intend to adopt those made available by the underlying technology, the CBR method(an example of which is described in [47]). This extra information has already been identified as 'ontology characterisations' and an early attempt to operationalise them is given by Uschold in [42]. Moreover, in [45], Uschold and Jasper describe possible scenarios where these characterisations could be of help in understanding ontology applications. As Uschold points

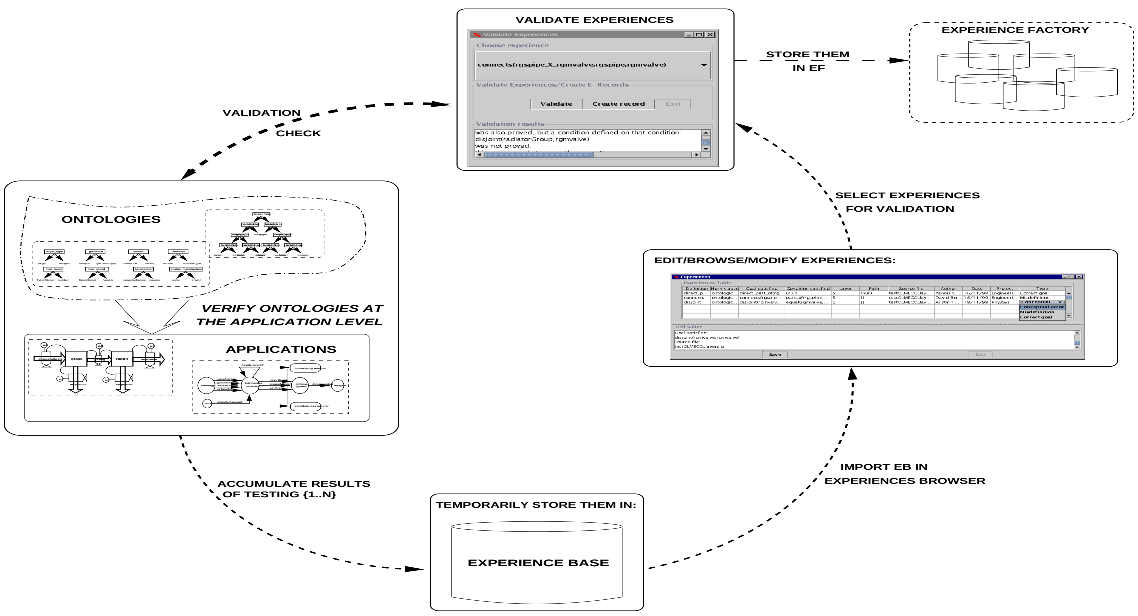

Figure 2. Experience-based architecture to support ontology verification(adopted from [27]).

out in [42]: "if this kind of description was provided for all applications, we would quickly get an overview of the state-of-the-art of ontology application, and a way to compare applications. It would be relatively easy to see which techniques have been used to apply particular kinds of ontologies in specific contexts". We found the goal of this research similar to our aim, which is to provide the means to organise and collect experiences gained during ontology testing and verification. Therefore, we found it practical to instantiate the proposed framework for characterising ontologies in our verification and testing scenario.

To make our case more precise we list below instantiations of the proposed framework tailored to our scenario. Each item represents a category, followed by its instantiation in the particular context. We also include a short explanation of each category in a parenthesis following its name.

- `Purpose`:*(the purpose of the ontology)* In our tests, both mereology and topology provide the building blocks for PHYSSYS ontology. They formalise generic mereotopological relations as described in the literature(i.e., [38] and [14]);
- `Representation languages and paradigms`:*(Ontolingua? Description Logics? Prolog? Clips? XML?)* The mereology and topology ontology were implemented in Ontolingua, we translated them in the implementation language we use: Prolog;
- `Meaning and formality`:*(to what extent and how formal is the specification of the meaning of each term?)* The Ontolingua versions of mereology and topology provide primitive terms with axioms restricting their use by placing constraints on relationships between types of entities. The implemented versions we used(in Prolog), include this information along with application specific constructs;
- `Subject Matter`:*(is it domain-specific or general?)* Both mereology and topology provide generic relations and axioms to be used in the PHYSSYS ontologies set;
- `Scale`:*(how big is the ontology?)* Both ontologies are quite small, each of them formalises less than 10 relations;
- `Development`:*(the degree to which the application is specified, developed and/or fielded)* The mereology and topology ontologies are research prototypes but the PHYSSYS which includes them has been used in the context of the OLMECO project([12]);
- `Conceptual architecture`:*(what are the main components in the ontology application, and how do they relate to each other)* A single research prototype system was used, the *hospital heating system*, implemented in Prolog, which includes ontological constructs;
- `Mechanisms and techniques`:*(what specific mechanisms and techniques were invoked to make use of the ontology)* We deployed a multi-layered architecture([28]) to embed ontological constructs in the application. We translated ontological constructs to the target implementation language: Prolog, and translated ontological axioms to a designated constraints format used to detect discrepancies with respect to misuse of ontological constructs([26]);
- `Implementation platform`:*(the particular implementation platform and context)* The ontologies are described textually in the literature and were also written in Ontolingua syntax,which we translated in Prolog. The testbed application was also implemented in Prolog. The tests were executed through a `Java` front-end.

The role of these characterisations is to provide ontology-specific information which can be used for organising and retrieving experiences on testing in similar ontology deployment efforts.

Recall figure 1 from section 3 where we presented the main experience factory tasks as realised in [4]. Here we reproduce that figure along with its instantiations in the context of our case: ontology verification.

As illustrated in figure 3, we have added information obtainable from our experiences with the ontology verification task. These are

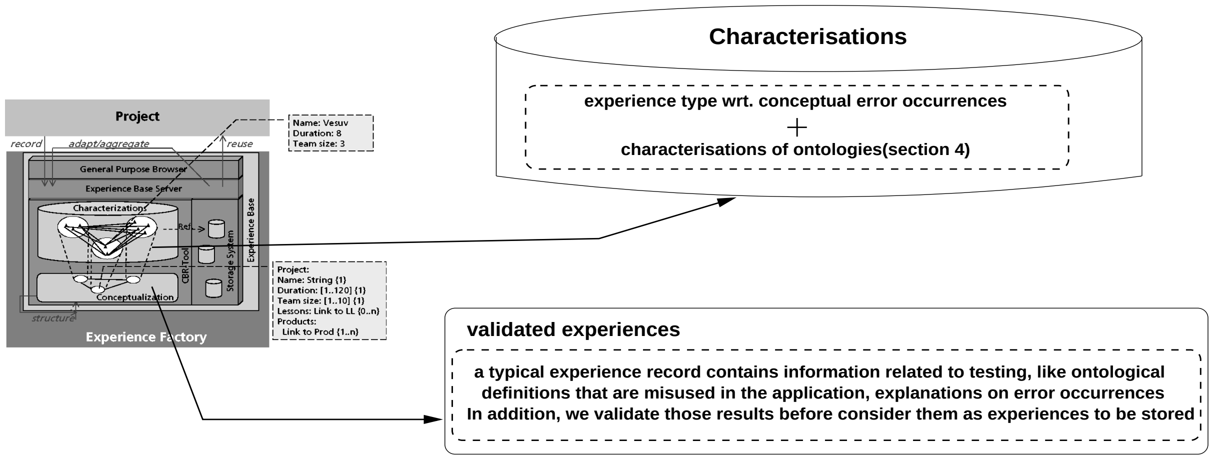

Figure 3. The experience factory instantiated with the ontology verification scenario(adopted from [27]).

results from code-testing, along with additional information such as explanations for error occurrences and are contained in an experience record which we validate before we store it in the experience base. In the characterisation phase we can give generic information, like the experience types with respect to conceptual error occurrences. In addition, at this phase we also provide links to the ontology characterisations described above, which are edited and linked to the relevant experiences through a designated experience table(described in [27]). This helps us to browse through the collected experiences and ultimately supports the idea of “learning from experience” which is an acknowledged need in ontology deployment.

5 OMs in KM

The development and use of KM ontologies hides important caveats. As O’Leary reports in [35]: “Each consulting firm we have been examining has built or is building its own ontologies. Because these enterprise ontologies are so costly to develop and maintain and are constantly changing, ontology or taxonomy issues are emerging as some of the most important problems in knowledge management”. O’Leary analyses further these problems from the KBSs development point of view in [34]. In this section we elaborate on our vision of “bringing OMs technology into KM core technology: ontologies”, which could, potentially, alleviate the situation. To explain the idea we illustrate the approach in figure 4.

On the left hand side of figure 4, we place within a box surrounded by a dash-lined border the OMs technologies. At the bottom of that box we place a potential technology for supporting OMs implementation: experience factories. The diagram included there is actually a reduction of figure 2 presented in section 4. It denotes an example use of experience factories in the area of ontology verification as we described in that section. There can be other technologies for implementing OMs. We point the interested reader to [1] were Abecker and colleagues elaborate on potential technologies for OMs implementation.

On the right hand side of figure 4, we illustrate the main KM tasks and activities. We identify four main KM tasks: *acquiring*, *analysing*, *using*, and *preserving* knowledge. We argue that these tasks are accomplished by activities which are supported by ontologies. In particular, the knowledge acquisition task, is accomplished by *identifying* activities which are supported by ontologies. This results in the application area of information extraction and/or content-matching. In the same manner, ontologies in the area of knowledge representation are used to *model* and *assess* the environment, which are activities employed in the *analysing* knowledge task. The *using* knowledge task, includes the *apply*, *share*, and *reuse* activities, which are supported by ontologies with such application areas as knowledge sharing and reuse, and KBSs. The last task is *preserving* knowledge. It is accomplished by activities such as *organising*, *maintaining*, and *capitalising* knowledge which are partially aided by ontologies. The resulting application area is that of libraries of reusable knowledge components and experience repositories. The knowledge preservation task and its accompanying activities along with the relevant ontologies are the area of overlap with experience factories as denoted by the dashed box surrounding the task in figure 4.

The way in which the two boxes of figure 4 are related summarises the linkage we are suggesting in this paper. In the introductory section, we argued that OMs and KM are intertwined areas. In this figure we illustrate how the technologies used to implement them can also be intertwined. As can be seen from the curly arrow connecting the OMs technologies with the KM tasks/activities box, technologies such as experience factories, can be employed to *organise* ontologies used to support main KM tasks/activities. The latter, in turn, can *support* OMs implementation by *acquiring*, *analysing*, *using*, and *preserving* knowledge to be processed by an OM.

There are mutual benefits for integrating OMs technology in KM ontologies. On one hand, an OM framework could help to improve ontology development and deployment, facilitate understanding, and ease reuse. A better organised ontology could, in turn, overcome some of the problems identified in [34]: “perfect ontology hype, library ontologies, scale-up, interface, formality”, and analysed from a cost-benefit point of view in [25]. We envisage deployment of experience factories, especially experience bases, in the whole spectrum of ontology life-cycle: experience bases on testing(an example of which was presented in section 4, taken from [27]), on development, on maintenance, etc. Ideally, all these experience bases could be placed alongside with their ontology counterparts in large repositories of reusable knowledge components. A metaphor of this idea

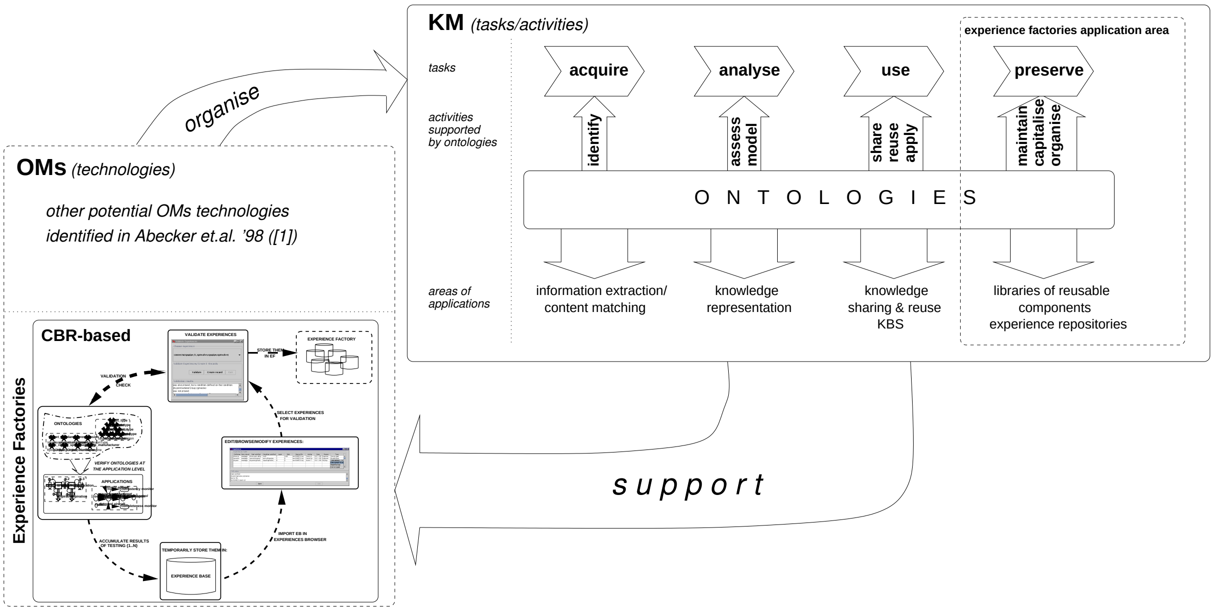

Figure 4. OMs in KM: OMs technology used to organise KM tasks/activities which in turn support the implementation of OMs.

are the implemented on-line libraries of ontologies, like for example, *Ontolingua*, *Ontosaurus*, and *WebOnto*.

On the other hand, better ontologies could help to meet practical requirements for the implementation of OMs. These were identified by Abecker and colleagues in [1]: "(i)collection and systematic organisation of information from various sources, (ii)ability to minimise up-front knowledge engineering, by taking advantage of readily available information, (iii)exploiting user feedback for maintenance and evolution, (iv)integration into existing work environment, (v)active presentation of relevant information". Requirements (i),(iv) and (v) could benefit more from ontologies as we briefly described in section 2 and diagrammatically present in figure 4. Requirements (ii) and (iii) could benefit more from the presence of an experience factory. In [2], Abecker and colleagues show how they satisfied these requirements in the *KnowMore* system which realises proactive knowledge delivery in order to enable employees solve cooperatively knowledge-intensive tasks.

6 Conclusions

In this paper we explored the possibility of employing OMs technology to improve understanding and enhance the usage of ontologies. The latter are a core technology for KM as we discussed in section 2. Despite the fact that OMs and KM are intertwined areas and usually are treated as a whole, we argue that the technologies used to implement and support them are not. We discussed the potential convergence of technologies for OMs, in particular experience factories, with ontologies in section 4 where we presented an example case in the area of ontology verification and testing. We generalised the approach in section 5 and speculated on the benefits of such convergence for both OMs and KM.

The intersection of these two technologies also highlights the role of knowledge engineering. As Milton and colleagues argue in [33] there is a change of focus in KM, from IT-based solutions to Knowledge Technology(KT)-based ones. To quote the authors: "[…] the main purpose of KT is to provide solutions to certain key problems associated with KM, such as scoping what knowledge is to be captured and disseminated, dealing with tacit knowledge, and facilitating better understanding". The field of knowledge engineering has been studying and practising solutions to these problems for years.

Applying these solutions to KM could help us to realise the 'ultimate goal' of OMs as envisaged by Kuhn and Abecker: "[…] to provide the necessary knowledge whenever it is needed. To assure this, [OMs] realize an active knowledge dissemination approach which does not rely on users' queries but automatically provides knowledge useful for solving the task at hand. To prevent information overload, this approach has to coupled with a highly selective assessment of relevance"([29]). The CBR-based experience factories coupled with ontologically-based reasoning are a way of achieving this 'selective assessment of relevance'.

In accordance with the 'ontology harvesting' notion we mentioned in the introduction we conclude the paper with a motto: *if ontologies are to be a cornerstone for successful KM, we need to manage them by bringing OMs technology into their development, deployment, and maintenance phases, which in turn, will result in a more effective OM*.

ACKNOWLEDGEMENTS

The research described in this paper is supported by a European Commission's Marie Curie Fellowship(programme: Training and Mobility of Researchers). Thanks to David Robertson for his comments on an early draft of this paper and to anonymous reviewers for their comments.

REFERENCES

[1] A. Abecker, A. Bernardi, K. Hinkelmann, O. Kuhn, and M. Sintek, 'Toward a Technology for Organizational Memories', *IEEE Intelligent Sys-*

tems, **13**(3), 40–48, (June 1998).

[2] A. Abecker, A. Bernardi, and M. Sintek, 'Proactive knowledge delivery for enterprise knowledge management', in *Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering(SEKE'99), Kaiserslauten, Germany*, pp. 120–127, (June 1999).

[3] K-D. Althoff, P. Bergmann, and L.K. Branting, eds. *Case-Based Reasoning Research and Development(ICCBR'99)*, number 1650 in Lecture Notes in Artificial Intelligence. Springer Verlag, July 1999.

[4] K-D. Althoff, A. Birk, S. Hartkopf, W. Muller, M. Nick, D. Surmann, and C. Tautz, 'Managing Software Engineering Experience for Comprehensive Reuse', in *Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering, SEKE'99, Kaiserslauten, Germany*, pp. 10–19, (June 1999).

[5] K-D. Althoff, F. Bomarius, and C. Tautz, 'Using Case-Based Technology to Build Learning Software Organizations', in *Proceedings of the ECAI'98 Workshop on Building, Maintaining, and Using Organisational Memories(OM-98), Brighton, England*, (August 1998).

[6] K.-D. Althoff, M. Nick, and C. Tautz, 'Improving organizational memories through user feedback', in *Proceedings of the Learning Software Organizations(LSO'99) workshop, Kaiserslauten, Germany*, ed., F. Bomarius, pp. 27–44, (June 1999).

[7] J. Aspirez, A. Gomez-Perez, A. Lozano, and S. Pinto, '(onto)2agent: An ontology-based www broker to select ontologies', in *Proceedings of the Workshop on Applications of Ontologies and Problem-Solving Methods, ECAI'98, Brighton, England*, pp. 16–24, (August 1998).

[8] V. Basili, G. Caldiera, and D. Rombach, 'Experience Factory', in *Encyclopedia of Software Engineering*, ed., J. Marciniak, volume 1, 469–476, John Wiley & Sons, (1994).

[9] V.R. Basili and H.D. Rombach, 'The TAME Project: Towards Improvement Oriented Software Environments', *Transactions on Software Engineering*, **SE-14**(6), 758–773, (June 1988).

[10] R. Benjamins and D. Fensel, 'The Ontological Engineering Initiative-KA2', in *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy*, ed., N. Guarino, pp. 287–301. IOS Press, (June 1998).

[11] R. Benjamins, D. Fensel, and A. Gomez-Perez, 'Knowledge Management through Ontologies', in *Proceedings of the 2th International Conference on Practical Aspects of Knowledge Management(PAKM'98), Basel, Switzerland*, (October 1998).

[12] P. Borst, H. Akkermans, and J. Top, 'Engineering Ontologies', *International Journal of Human-Computer Studies*, **46**, 365–406, (1997).

[13] B. Chandrasekaran, R. Josephson, and R. Benjamins, 'What Are Ontologies, and Why Do We Need Them?', *IEEE Intelligent Systems*, **14**(1), 20–26, (January 1999).

[14] B.L. Clarke, 'A calculus of individuals based on "connection"', *Notre Dame Journal of Formal Logic*, **22**, 204–218, (1981).

[15] P. Cohen, R. Schrag, E. Jones, A. Pease, A. Lin, B. Starr, D. Gunning, and M. Burke, 'The DARPA High Performance Knowledge Bases project', *AI Magazine*, **19**(4), 25–49, (1998).

[16] S. Decker, M. Erdmann, D. Fensel, and R. Studer, 'Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information', in *Proceedings of DS-8, Semantic Issues in Multimedia Systems, Boston, MA, USA*, ed., R & et.al. Meersman, pp. 351–369, (1999).

[17] J. Domingue, 'Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web', in *Proceedings of the 11th Knowledge Acquisition, Modelling and Management Workshop, KAW'98, Banff, Canada*, (April 1998).

[18] A. Farquhar, R. Fikes, and J. Rice, 'The ontolingua server: a tool for collaborative ontology construction', *International Journal of Human-Computer Studies*, **46**(6), 707–728, (June 1997).

[19] D. Fensel, V.R. Benjamins, E. Motta, and B. Wielinga, 'UPML: A Framework for knowledge system reuse', in *Proceedings of the 16th International Joint Conference on Artificial Intelligence, IJCAI'99, Stockholm, Sweden*, pp. 16–21, (August 1999).

[20] M. Fernandez, A. Gomez-Perez, and N. Juristo, 'METHONTOLOGY: From Ontological Arts Towards Ontological Engineering', in *Proceedings of the AAAI-97 Spring Symposium Series on Ontological Engineering, Stanford, CA, USA*, pp. 33–40, (March 1997).

[21] T.R. Gruber, 'Towards principles for the design of ontologies used for knowledge sharing', *International Journal of Human-Computer Studies*, **43**, 907–928, (1995).

[22] N. Guarino, 'Formal Ontology and Information Systems', in *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy*, ed., N. Guarino, pp. 3–15. IOS Press, (June 1998).

[23] N. Guarino, ed. *Formal Ontology In Information Systems*, Frontiers in Artificial Intelligence and Applications. IOS Press, June 1998. ISBN: 90-5199-399-4.

[24] N. Guarino, C. Masolo, and G. Vetere, 'OntoSeek: Content-Based Access to the Web', *IEEE Intelligent Systems*, **14**(3), 70–80, (May 1999).

[25] Y. Kalfoglou, T. Menzies, K-D. Althoff, and E. Motta, 'Meta-knowledge in systems design: panacea...or undelivered promise?', *The Knowledge Engineering Review(submitted)*, (2000).

[26] Y. Kalfoglou and D. Robertson, 'Managing Ontological Constraints', in *Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods(KRR5), Stockholm, Sweden*, http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-18/, (August 1999). Also as: Research Paper No.948, Dept. of AI, University of Edinburgh.

[27] Y. Kalfoglou and D. Robertson, 'Applying Experienceware to support ontology deployment', in *Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering, SEKE2000, Chicago, IL, USA*, (July 2000).

[28] Y. Kalfoglou, D. Robertson, and A. Tate, 'Using Meta-Knowledge at the Application Level', *Journal of Artificial Intelligence Research, submitted*, (2000). Also as: Research Paper No.956, Dept. of AI, University of Edinburgh.

[29] O. Kuhn and A. Abecker, 'Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges', *Journal of Universal Computer Science*, **3**(8), 929–954, (1997).

[30] D.B. Lenat and R.V. Guha, *Building large knowledge-based systems. Representation and inference in the Cyc project*, Addison-Wesley, Reading, Massachusetts, 1990.

[31] M. Liao, A. Abecker, A. Bernardi, K. Hinkelmann, and M. Sintek, 'Ontologies for Knowledge Retrieval in Organizational Memories', in *Proceedings of the Learning Software Organizations(LSO'99) workshop, Kaiserslauten, Germany*, ed., F. Bomarius, pp. 19–26, (June 1999).

[32] L.D. McGuinness, 'Ontological Issues for Knowledge-Enhanced Search', in *Proceedings of the 1st International Conference on Formal Ontology in Information Systems(FOIS'98), Trento, Italy*, ed., N. Guarino, pp. 302–316. IOS Press, (June 1998).

[33] N. Milton, N. Shadbolt, H. Cottam, and M. Hammersley, 'Towards a knowledge technology for knowledge management', *International Journal of Human Computer Studies*, **51**(3), 615–641, (September 1999).

[34] D. O'Leary, 'Impediments in the use of explicit ontologies for KBS development', *International Journal of Human-Computer Studies*, **46**(2), 327–337, (1997).

[35] D. O'Leary, 'Knowledge Management Systems: Converting and Connecting', *IEEE Intelligent Systems*, **13**(3), 30–33, (June 1998).

[36] D. O'Leary, 'Using AI in Knowledge Management: Knowledge Bases and Ontologies', *IEEE Intelligent Systems*, **13**(3), 34–39, (June 1998).

[37] U. Reimer. Building, Maintaining, and using Organisational Memories. Invited talk in the ECAI'98 Workshop on Building, Maintaining, and Using Organisational Memories(OM-98), Brighton, England, August 1998.

[38] P. Simons, *Parts: A Study in Ontology*, 5–128, Oxford: Clarendon Press, 1987.

[39] T. Summer and S. Buckingham-Shum, 'From Documents to Discourse: Shifting Conceptions of Scholarly Publishing', in *proceedings of the CHI'98: Human Factors in Computing Systems, Los Angeles, CA, USA*, pp. 95–102. ACM Press, (1998).

[40] B. Swartout, R. Patil, K. Knight, and T. Russ, 'Toward Distributed Use of Large-Scale Ontologies', in *Proceedings of the 10th Knowledge Acquisition, Modeling and Management Workshop(KAW'96),Banff,Canada*, (November 1996).

[41] C. Tautz and C. Gresse von Wangenheim, 'A Representation Formalism for Supporting Reuse of Software Engineering Knowledge', in *Proceedings of Workshop in Reuse in Developing Knowledge-Based Systems(XPS'99), Wurzburg, Germany*, (March 1999).

[42] M. Uschold, 'Where are the Killer Apps?', in *Proceedings of Workshop on Applications of Ontologies and Problem Solving Methods, ECAI'98, Brighton, England*, ed., Gomez-Perez,A. and Benjamins,R., (August 1998).

[43] M. Uschold, P. Clark, M. Healy, K. Williamson, and S. Woods, 'An Experiment in Ontology Reuse', in *Proceedings of the 11th Knowledge Acquisition Workshop, KAW98, Banff, Canada*, (April 1998).

[44] M. Uschold and M. Gruninger, ‘Ontologies: principles, methods and applications’, *The Knowledge Engineering Review*, **11**(2), 93–136, (November 1996).

[45] M. Uschold and R. Jasper, ‘A Framework for Understanding and Classifying Ontology Applications’, in *Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods(KRR5), Stockholm, Sweden*, (August 1999). http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-18/.

[46] A. Valente, T. Russ, R. MacGrecor, and W. Swartout, ‘Building and (Re)Using an Ontology for Air Campaign Planning’, *IEEE Intelligent Systems*, **14**(1), 27–36, (January 1999).

[47] G. von Wangenheim, K-D. Althoff, and M.R. Barcia, ‘Intelligent Retrieval of Software Engineering Experienceware’, in *Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering(SEKE’99), Kaiserslauten, Germany*, (June 1999).