B.H. Far's Research
© B.H. Far
far@ucalgary.ca
http://www.enel.ucalgary.ca/People/far

Theme 6: Enterprise Knowledge Management Using Knowledge Orchestration Agency

Abstract

Distributed knowledge management, intelligent software agents and XML based knowledge representation are three research challenges which are changing the face of knowledge management solutions. Distributed knowledge management allows multiplicity and heterogeneity of perspectives within organizations and consequently map technological architecture to organizational structure. In order to take the maximum advantage of distributed knowledge paradigm, software agents are well suited to handle complex interactions in social process of knowledge management. On the representation side, XML based knowledge representation moves toward popularization of knowledge representation and consequently allows machines to exchange knowledge and process it. In this article we explore structural requirements of distributed knowledge management and how to empower that with software agents and XML based knowledge representation in order to make an orchestrated knowledge management system which is compatible with the organizational structure.

1. Introduction

Over the past decade, we have seen a growing interest in Knowledge Management (KM) solutions for intellectual assets in organizations including techniques and tools for document management, workflow management, transparent capture, web conferencing, visual thinking, digital whiteboards, as well as data warehouses, decision support systems, group-wares and intranets. While most of the enabling technologies for knowledge management have been around for many years, the ability to seamlessly integrate the components into a cohesive infrastructure evaded organizations until the advent of the Internet. Open Internet standards present unprecedented opportunities for implementing knowledge management solutions, and its evolution provides a ubiquitous binding medium for integrating distributed applications, formats, and contents.

The first attempts towards business integration were made at the data level, where client-server computing frameworks have been developed to support distributed computing in heterogeneous environments and provide an interface description language and services that allow objects to be defined, located and invoked across locations and applications [1]. The most popular of such distributed object paradigms are CORBA, the Common Object Request Broker Architecture developed by OMG, DCOM, Microsoft's Distributed Component Object Model and RMI Sun's Remote Method Invocation. Such frameworks encapsulate the heterogeneity of legacy systems and applications within standard, interoperable wrappers. These frameworks are defined and are well suitable to the 'data' level of communication and sharing. They presuppose a relatively stable environment, well defined processes and common grounds of understanding.

A relatively new paradigm is business integration at the knowledge level. In this level, the basic requirements for KM solutions are:

- Incorporate the management of knowledge assets with environments that support distributed knowledge bases which belong to both people and departments.
- Allow for dynamic classification and distribution of knowledge.
- Allow for adapting to diversified contents, representation and personalized styles.
- Incorporate efficient (i.e., fast and effortless) retrieval mechanisms.
- Facilitate interaction between distributed knowledge bases in order to support social process of knowledge management.

A knowledge-level integration framework must be able to create dynamic relationships between knowledge-intensive business processes and knowledge sources that do not compromise the autonomy of the different parts while support distinctive perspectives and transformation between them. Also KM systems must provide uniform access to a diversity of knowledge and information sources of different degree of formality and granularity. Furthermore, knowledge management environments must be able to adapt to the different needs and preferences of individuals or group of users, and integrate naturally with existing work methods, tools and processes.

These requirements imply that the integration of knowledge and business processes requires a different architecture than the one provided by the centralized frameworks. The peer-to-peer (P2P) technology is considered as a solution. P2P technology is viewed as a way of enabling distributed control, differentiation, customization and redundancy. This paradigm represents organizational cognition as a distributed process that balances the autonomous knowledge management of individual and groups, and the coordination needed in order to exchange knowledge across different autonomous entities [11]. Furthermore, in order to have sophisticated knowledge sharing between individuals and/or groups there must be a coordination mechanism in place which is not on basis of creating common languages or schemas, like centralized approaches. Consequently, organizations and their supporting systems must be able to allow for a degree of negotiation and adaptability in order to accommodate individual participation [1].

P2P infrastructures are a very straightforward way of mapping social architectures onto technological architecture and because of focus on individuals it encourages the discovery of tacit knowledge in addition to traditional discovery from encoded knowledge. Furthermore because of distribution of knowledge, it has a highly robust and scalable structure and because of autonomy of peers it distributes maintenance and administration among peers. These may lead to higher business integration efficiency and lower running costs.

Although P2P is a particularly powerful paradigm when it comes to satisfying the basic requirements of knowledge management solutions by allowing distribution of knowledge and diversity of contexts, it says nothing about implementation the KM helper tools and KM processes. From the conceptual point of view, peers are simple computational entities and their work is limited to the protocol level, therefore they don't support complex category of skills which are necessary in KM solutions such as collaboration, coordination, knowledge acquisition and reasoning. With regard to these capabilities, intelligent software agents and socialability of multiagent systems (MAS) seem well suited for knowledge management. Software agents not only can get involved in complex interactions but offer implementing the KM helper tools (search, scheduling, negotiating, decision support, etc.) and KM processes (knowledge discovery, knowledge capture etc.) Therefore the choice of MAS as the implementation technology for KM is motivated by the following observations:

- The solution for KM problems cannot be entirely prescribed from start to finish and therefore problem solvers are required that can respond to changes in the environment, to react to unpredictability of business process, to proactively take opportunities when they arise and make decisions on different situations from knowledge acquisition phase to maintenance (reasoning and learning capability). [22]
- Interactions are fairly sophisticated, including negotiation, collaboration, and coordination and require certain level of environment awareness.

In this article we investigate challenges in three different KM related research areas which we believe are changing the face of knowledge management solutions:

- ***Distributed Knowledge Management***: The impact of technology on organizational structure shaping and the impact of organizational structure on the concrete formation of technology is a bilateral relation which, in this article, we explore it in the context of KM systems. Many researchers believe that on the one hand knowledge is developed within communities, namely groups of people that share a common perspective, and on the other hand, knowledge developed as a consequence of the interaction between different communities [14]. Based on these beliefs, technological architectures should support distribution of knowledge bases in fine grained organizational subpart level (departments, group of people and individuals) and provide the following features: autonomy of individuals and groups, diversity of content, representation and personal style and sociality of knowledge management environment with enabling interaction between participants.
- ***Intelligent Software Agents***: Distribution of knowledge must be supported by using intelligent techniques to help organizational entities to get involved in the KM process. Knowledge distribution transfers some responsibilities to people or group of people and if KM solution cannot provide facilitating mechanisms it will be useless. On the other hand, social architecture of KM environment need to be enabled by allowing complex interactions (e.g., negotiation) between participants. These requirements are unique characteristics of software agents.

- ***XML based Knowledge Representation***: The emerging power of XML to increase the degree of control over document representation and to explicate the standards for exchanging information that is structured for further processing is going to dramatically change the knowledge representation. It offers a simplified machine processable syntax which helps in popularization of knowledge representation mechanism. Semantic Web technology uses XML based ontology definition language in order to make KM solutions more efficient.

In the following sections we provide a chronological overview of various technologies for the knowledge management; describe the current progresses in knowledge representation and architectures; and discuss the role of software agent technology in the next generation knowledge management systems. We conclude with comparing the different architectures for knowledge management from the KM processes point of view.

2. Knowledge Integration on the Web

The Web as the most important innovation in the Internet is a technology which affects KM solutions in an extreme manner. This technology offers client-server model for information access and provides ubiquitous knowledge sharing environment. On the other hand, in conjunction with collaboration software, it provides a communication infrastructure which with facilitating interactions among individuals and encourages sociability among them. The installation of corporate wide intranets which is commonly constructed using the Web technology helps organizations to exploit Web in a smaller scale.

2.1 Structural issues

World Wide Web offers a simple and straightforward infrastructure to implement centralized KM solutions. It provides a shared and unique point of access supporting the concept of enterprise knowledge portals. Usually in the web based solutions for KM, corporate knowledge is externalized and installed on a server or a cluster of servers. In order to provide efficient and transparent access to heterogeneous knowledge sources, some additional systems should be added to KM systems in order to act as intelligent intermediaries between users and various knowledge sources (Figure 1). These systems include [22]:

- ***Facilitators*** that search for potential sources of information and provide mechanisms for accessing them. They rely on automating mediation tasks to the extent feasible. Facilitators typically require human intervention when change to resources, users requirements, or process specific knowledge occur. Query processor is an example of facilitator that reformulates user requests to improve the prospect of accessing relevant information. Another example for facilitator systems, which we believe are vital for KM solutions for heterogeneous knowledge sources, is retrieval mediators. Mediator architecture provides intermediary services that link data sources and application programs so that information follow across heterogeneous sources are facilitated without the need to integrate the base data sources.

- ***Intelligent agent and data mining components*** include tools such as decision support systems, push and pull based intelligent agents which are installed on client side, automatic indexing tools and pattern discovery tools. Filtering, editing searching, and packaging are essential to knowledge management design. Without powerful search and intelligent retrieval and mining tools to support meta-information creation, a knowledge management system is as limited as a traditional information system.

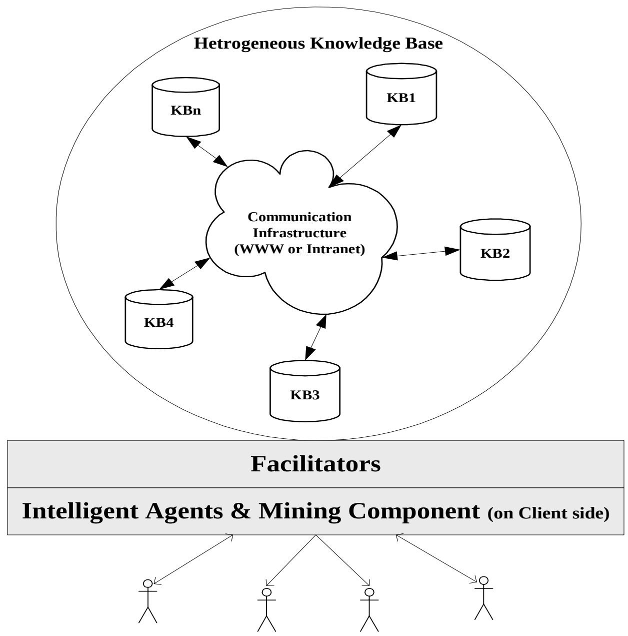

Figure 1. Typical structure of web-based centralized knowledge management systems

Some features of Web based knowledge management solutions are:

- Provision of ubiquitous access to heterogeneous knowledge repositories.
- Supporting centralized infrastructure for semantically homogeneous knowledge sharing.
- Creation of a new role, knowledge manager whose goal is to support and facilitate knowledge externalization across organizational subparts.
- Creation of codification process that is supposed to be used as a common process in externalization, codification and refinement of corporate knowledge.
- Supporting the creation of virtual communities around KM system in which community members can communicate in order to share expertise and capture process knowledge.

With regard to mapping between organizational structure and technological architecture, there are two very important questions which should be answered.

1. Is the Web suitable to map organizational structure to technological architecture?
2. If the Web technology is not suitable for mapping organizational structure why it is structurally the most dominant KM solution?

To answer the first question we should accept that the Web technology does not completely map organizational structure into technological architecture. As we mentioned before, knowledge in organizational structure is intrinsically distributed among people and departments and technological architecture must provide an environment that support distribution of knowledge among individuals and groups and facilitate and encourage interaction between them. The Web technology distributes access to knowledge rather than distributing knowledge among knowledge owners. One consequence of centralized KM solutions is elimination of all subjectivity and context of knowledge. Web based centralized solutions for KM usually implement a process of knowledge extraction and refinement whose aim is to eliminate different perspectives to knowledge and create a general representation that can be reused by other people in a variety of situations. Many researchers believe that this generalization is one of the reasons why so often KM systems are abandoned [14].

To answer the second question, firstly we should accept that the add-ons to the Web provide some levels of informality and sociality in working environments. For example, collaboration software enables and facilitates organizational interactions and mining agents allow information retrieval. It should be noted that these add-ons are not a basic building block of the Web technology. Second reason for acceptance of Web based KM solutions is strongly related to traditional paradigm of managerial control [15]. In particular, management allocates and distributes resources to employees and monitors the use of resources. Centralized nature of Web based solutions is compatible with this traditional view on the managerial function of command and control. Opposed to centralized systems, in distributed solutions knowledge remains embedded within different participants resulting that it falls outside the boundaries of managerial control.

Despite of many advantages of Web based systems it seems that these solutions make people and organizations adapt to technologies rather than focusing to design KM solutions for the people and organization.

2.2 Knowledge Representation issues

Web pages, as an image of parts of human knowledge, are building blocks of World Wide Web which usually includes users (organization) Web pages, community Web pages and users' profile. By projecting human knowledge to Web pages we keep information on pages, represent them by human natural language and lose reasoning aspect (subjectivity and contextual aspects) of our knowledge (Figure 2). It is up to the users to use their human intelligence to figure out relations in structure of pages and understand them by discretion. Current knowledge management vision to the Web has significant weaknesses: [10]

- Existing keyword-based searches can retrieve irrelevant information that includes certain terms in different meanings. They also miss information when different terms with the same meaning about the desired content are used. Currently, human browsing and reading is required to extract relevant information from information sources. This is because automatic agents can not possess contextual aspects of knowledge and also common sense knowledge.
- Weakly structure text sources are difficult to maintain. Keeping such collections consistent, correct and up-to-date requires explicit representation of semantics.

There are some signs of another vision of the Web*: huge distributed knowledge base* [5]. Based on this vision, Web is constructed from pages as 'an image of (part of) human knowledge'. Although current efforts can help in construction of Web as huge distributed knowledge base, they are not enough to sway our mind from Web of information repository to Web of huge distributed knowledge base. Promoting information on the Web to knowledge and managing this knowledge require the use of intelligent techniques both in representing the knowledge and reasoning about it.

Figure 3 shows the next generation of the Web. In this generation we have a kind of knowledge representation, which in conjunction with agents gives reasoning capability to machines. Currently these agents are located on the client side and with accessing remote web sites can provide precise and classified knowledge for their owners.

The Web currently contains a lot of data, more and more structured data (online databases, structured documents) and simple metadata but very little knowledge. Data indexed by formal terms (categories) or organized by inclusion links is called "structured data". Beliefs, definitions, facts, or rules represented using a formal language and ontology is most often called "knowledge" (or knowledge representation) [6]. Currently, two efforts are going on to make information on the Web to be represented explicitly as (machine process-able) knowledge.

- ***Intelligent Data Preprocessing*** [7]: These techniques are trying to excerpt knowledge from Web pages (from Human language representation). It uses Web-mining techniques including Web link structure mining, Web content mining and Web log mining
- ***Semantic Web*** [8, 9, 10]: This method seems more efficient than the first one and tries to inject machine understandable knowledge within Web documents. This method actually enriches the Web documents semantically and has been developed base on XML. XML is the basis for a rapidly growing number of software activities. It is designed for mark-up in documents of arbitrary structure, as opposed to HTML, which was designed for hyper text documents with fixed structure. A document type definition (DTD) defines a grammar to specify allowable combinations and nesting of tag names, attribute names, and so on. There is an alternative for DTD: XML Schema. Although XML Schema offers several advantages over DTDs, their role is

essentially the same. XML is used to serve a range of purposes: serialization syntax for other mark-up languages, separating form from content and uniform data-exchange format.

Although XML has given software industry a revolutionary power it only specifies syntactic conventions and any intended semantics are outside the domain of the XML specification. The resource description framework (RDF) is a recent W3C recommendation designed to standardize the definition and use of meta-data descriptions of web-based resources. However, RDF is equally well suited to representing data. The basic building block in RDF is an object–attribute–value triple, commonly written as A(O,V). That is, an object O has an attribute A with value V. RDF also allows a form of reification in which any RDF statement can be the object or value of a triple on the web this allows us, for example, to express doubt or support of statements created by other people. Finally, it is possible to indicate that a given object is of a certain type. It is important to note that RDF is designed to provide a basic object–attribute– value data model for meta-data. Other than this intended semantics, described only informally in the standard, RDF makes no data-modeling commitments. In particular, no reserved terms are defined for further data modeling. As with XML, the RDF data model provides no mechanisms for declaring property names that are to be used.

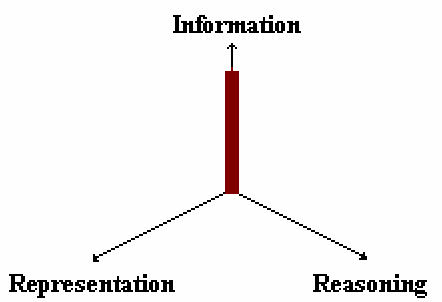

Figure 2. from machine understanding point of view we only have information in current Web.

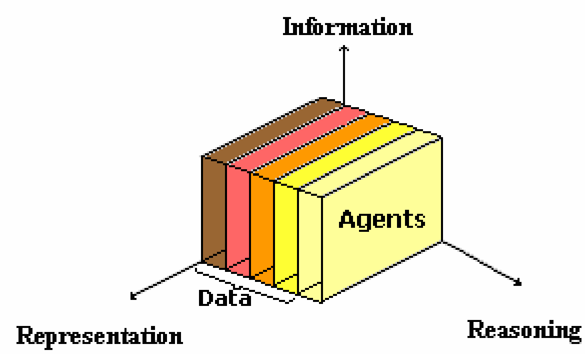

Figure 3. Next generation of the Web gives reasoning capability to machine. Also agents will explore maximum efficiency of Web.

RDF Schema takes a step further into richer representation formalism and introduces basic ontological modeling primitives into the web. With RDFS, we can talk about classes, subclasses, subproperties, domain and range restrictions of properties, and so forth in a web-based context. Despite the similarity in their names, RDFS fulfils a different role than XML Schema. XML Schema, and also DTDs, prescribe the order and combination of tags in an XML document. In contrast, RDFS only provides information about the interpretation of the statements given in an RDF data model, but it does not constrain the syntactical appearance of an RDF description. RDFS lets developers define a particular vocabulary for RDF data (such as hasName) and specify the kinds of object to which these attributes can be applied. In other words, the RDFS mechanism provides a basic type system for RDF models. With regard to ontologies, RDFS provides two important contributions: a standardized syntax for writing ontologies and a standard set of modeling primitives such as instance-of and subclass-of relationships.

The RDFS can be regarded as an ontology language However; many types of knowledge cannot be expressed in this simple language. It is clear that a richer language than RDFS is required if we want to be able to express anything but the most trivial domain models on the Semantic Web. In response the DARPA has been released DARPA Agent Markup Language (DAML) [24], a simple language for expressing more sophisticated RDF class definitions than permitted by RDFS. The DAML group soon joined efforts with the Ontology Inference Layer (OIL) another effort providing more sophisticated classification, using constructs from frame-based artificial intelligence. The result of these efforts is DAML+OIL, a language for expressing far more sophisticated classifications and properties of resources than RDFS.

3. Peer-to-Peer Knowledge Management

The term peer-to-peer (P2P) refers to a class of systems and applications that employ distributed resources to perform an important function in a decentralized manner. With the pervasive deployment of computers, P2P is increasingly receiving attention in research, product development, and investment circles. Some of the benefits of P2P approach include: improving scalability by avoiding dependency on centralized points; eliminating the need for costly infrastructure by enabling direct communication among clients; and enabling resource aggregation. P2P is about sharing: giving and obtaining from the peer community. Assuming that peer is defined as “like each other” a P2P system then is one in which autonomous peers depend on other autonomous peers. Peers are autonomous when they are not wholly controlled by each other or by the same authority. Peers depend on each other for getting information, computing resources and forwarding requests which are essential for the functioning of system as a whole [14].

Conceptually, P2P computing is an alternative to the centralized and client-server models of computing, where there is typically a single or small cluster of servers and many clients. In the purest form, the P2P model has no concept of server; rather all participants are peers.

Tree main classes of P2P applications have emerged: parallelizable, content and file management, and collaborative. Parallelizable P2P application split a large task into smaller subtasks that can execute in parallel over a number of independent peer nodes. Content and file management P2P applications focus on storing information on and retrieving information from various peers in the network. Collaborative P2P applications allow users to collaborate, in real time, without relying on a central server to collect and relay information.

3.1 Structural issues

Figure 4 shows an architectural overview to P2P knowledge management. P2P system architecture box shows an informal architecture for P2P solutions. We discuss the function of each layer which mostly based on [19].

The P2P Model covers a wide spectrum of communication paradigms. At one end of the spectrum are desktop machines that mostly connected via stable, high speed links over the Internet and at the other end there are small wireless devices such as PDAs which are connected in an ad-hoc manner via a wireless medium. The basic challenge of communication layer is overcoming the problems associated with dynamic nature of peers. Peers group are frequently changed and maintaining application level connectivity is one of the biggest challenges facing P2P developer.

Group management layer and robustness layer are essential layers which implement basic requirement for a P2P network. Peer group management includes discovery of other peers in community and locating them and routing between peers. Discovery of peers can be highly centralized or highly distributed or somewhere in between. In robustness layer there are three main functionalities which should be considered: security, resource aggregation and reliability. Main benefit of P2P is that it allows nodes to function both as a client and as a server. Transforming an standard client to a server raises number of hazards to a peer. Security requirements usually require intervention from the user, or interaction with a trusted third party which centralize the task of security resulting in voiding the benefit of distributed infrastructure. Although P2P model should provide the basis for interacting peers to aggregate different resources available on their systems, we believe in KM solutions this can be done in class specific layer. Generally, resources in a P2P network are not limited to files and CPU processing power, bandwidth and disk space can be considered as resources. The distributed essence of P2P networks makes it difficult to guarantee reliable behaviour. The most common solution to reliability across P2P systems is to take advantage of redundancy. For example in case of file sharing applications data can be replicated across some peers.

Class specific layer is a concrete abstraction to define knowledge management requirements. This layer should satisfy knowledge management objectives [14, 21]. These objectives can be achieved by implementing processes that we consider them as KM design processes and later, we use these processes as criteria when we compare different perspectives to KM.

Although the theory behind P2P has limited it into a protocol level concept, applications which are running on top of these protocols can solve many problems concerning KM. *That is because of potential power of P2P model in mapping social architecture of organization onto technological architecture.* Sociologically; creation, codification, and sharing of knowledge within complex organization is a process that can be described along two qualitatively different dimensions [15]:

- Knowledge is developed within communities, namely people or groups of people that share a common perspective rather than development by separated group of experts.
- Knowledge is developed as a consequence of the interaction between different communities.

Bonifacio at el. [14] has argued *that the main problem of most current KM systems is that they do not support the two social knowledge processes described above but rather tend to impose a process of a very different nature resulting in a technological architecture which is semantically centralized.*

P2P knowledge management is an emerging concept which is aiming the need for technological architecture that is more coherent with the social model for organizational cognition. Rather than a central knowledge repository, P2P KM considers (group of) individuals in organization or processes in organization as knowledge repositories and should provide tools to facilitate creation, codification and propagation through enabling interaction among knowledge owners. In addition to its highly scalable structure it distributes administration and maintenance of knowledge repositories among individuals.

The P2P KM technology should mainly support the following requirements:

- Automatic or semi-automatic capturing of peers' owner knowledge. P2P KM distributes responsibility of knowledge acquisition among peers which can be a tedious work for them. In order to decrease the knowledge acquisition efforts automatic or semi automatic techniques should be exploited.
- Coordination process among autonomous peers, in order to capture processes knowledge.
- Giving each community the possibility to represent and organize knowledge according to its goals and interpretative perspective.
- Providing tools to support the exchange of knowledge across different knowledge bases without assuming shared meanings, but rather enabling the dynamic translation of different meanings. (semantic interoperability)
- Setting mechanisms and protocols to enable the emergent and bottom-up formation of informal communities and communication practices (such as finding or addressing people to trusted individuals/groups)

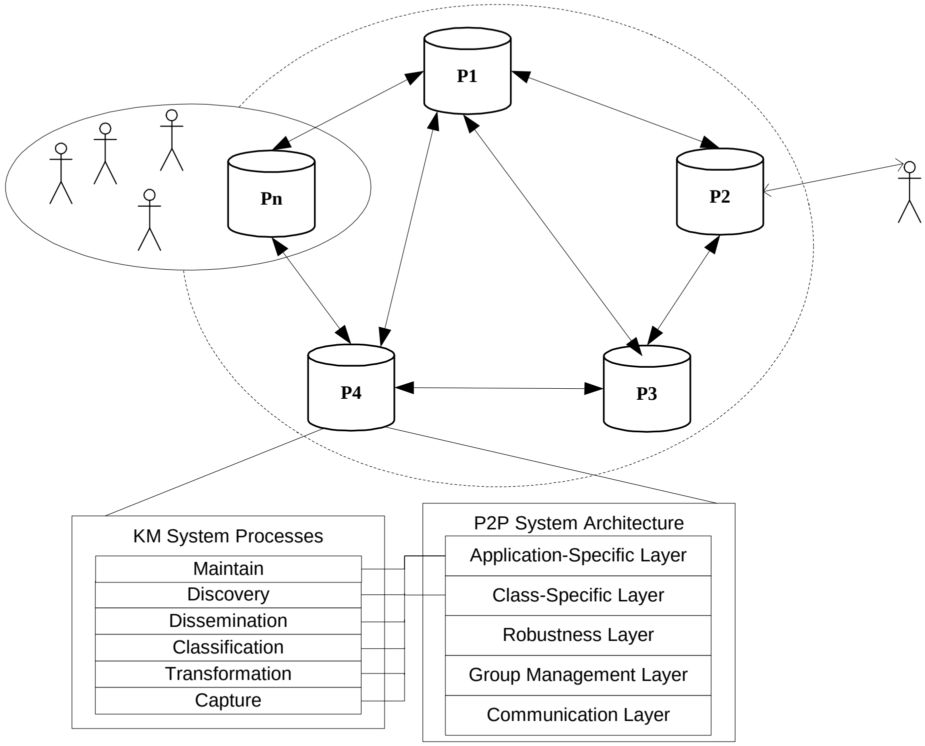

Figure 4. Typical structure of distributed peer to peer knowledge management systems.

3.2 Knowledge Representation Issues

Peer-to-Peer paradigm in conjunction with Semantic Web technology will be an interesting path to switch from the more centralized KM solutions that are currently implied by ontology-based solutions to a decentralized approach. P2P solutions open up the way to extract common conceptualizations among employees within an enterprise in a bottom-up manner which means, they directly map technological architecture onto organizational structure rather adapting organizational structure with technological architecture. Another feature of this solution is that it makes knowledge management solutions be nearly free of administration in order that they may be used by everyone including private persons and small cooperating departments. The current P2P solutions have some important limitations that should be solved when we combine it with Semantic Web technology [3]:

- Leading industry efforts such as JXTA by Sun Microsystems are limiting P2P services to string matching. No support for shared ontologies is provided. Queries are specified in arbitrary XML formats. No use is being made of the opportunities to use RDF/RDF Schema for expressing shared vocabularies. Finally, JXTA limits query answering to using resources in a single location, while in fact many queries will require the combination of information retrieved from different sources at different locations.

- Peer selection is currently not really based on content. This needs to be improved to route a query to the next knowledgeable peer rather than some arbitrary one.
- In many cases, P2P is discussed as a solution at the protocol level (no client-server dichotomy) and as a means for distributing disk space. However, this is of minor importance for improved service in knowledge management. Here it is the actual sharing of information and knowledge, which needs to be supported, and not the organization of disk storage or network traffic.

It is widely accepted that the use of emergent semantic could be the key to the success of peer-to-peer paradigm in KM solutions [3, 25]. Semantics builds on lightweight and/or heavyweight ontologies that different individuals, departments, or organizations have created. It considers the overlap between ontology definitions and the use of concepts and relations with actual data in order to extract shared ontologies for sets of individuals or groups of people. Intelligent tools will use such definitions to ensure that knowledge will be appropriately structured, so that it can be easily found. Knowledge Management can occur in a distributed fashion without overhead through central administration.

Some of the Challenges when combining P2P systems with Semantic Web technology are:

- There is a strong similarity between looking for a right document in a centralized KM system and looking for a right peer in a distributed KM system. Semantic Web can help peer selection mechanisms to exploit similarity of ontologies for this purpose.
- Having distributed knowledge sources allows peers to have different ontologies. Efficient techniques should be developed in order to let ontology-based interoperability and to align and map different ontologies.
- In a P2P environment, one cannot expect any maintenance to happen on the ontologies (in fact, users will often not know what is in the ontologies on their machine). As a result, mechanisms must be designed to allow the ontologies to update themselves, in order to cope with ontological drift. Based on the queries and answers elsewhere in the P2P network, ontologies will have to adjust their own definitions accordingly.

4. Agent Mediated Knowledge Management

Agent Mediated Knowledge Management (AMKM) has been an active research area in software agent research community in recent years [22, 26, 27]. Conceptually it uses agent concepts for analyze and model organizations and their knowledge needs and to provide a reusable architecture to build KM systems.

From distributed systems point of view, multiagent systems (MAS) fall in P2P category. Instead of peers, agents are the fundamental actors on MAS and they can discover other agents, locate them and communicate with them. We believe that AM KM and P2P KM are structurally compatible and knowledge representation mechanisms can be the same for both. Also some of the observations which motivate researchers to use agents in KM solutions are common with P2P systems. For instance, inherent distribution of knowledge, autonomy of organizations' subparts and functional separation between knowledge use and knowledge sources as a way to incorporate dynamic behaviour into information system design are some observations which programs running on class specific layer and application layer of P2P systems are able to succeed in doing them.

We strongly believe that what make AM KM different from P2P KM are that agents can learn and adapt themselves to changes in environment and also they can get involved in complex interactions.

If we consider agents as autonomous and social entities then we have accepted that they can learn and can get involve in complex interactions. Autonomy of an agent cannot be achieved without complete definition of its role and giving it learning capability and learning is nothing but having the ability to adapt to changes in the environment.

As we stated before, P2P solutions for KM tasks introduce an infrastructure which is straightforward in mapping social architecture onto technological architecture. Although this mapping distributes control over KM task participants, it introduces some other problems. These problems are essentially complex in nature and need capabilities which ordinary programs usually cannot undertake them.

4.1 Agents and peer knowledge capture

P2P paradigm distributes knowledge creation and capture process among peers. Distribution of knowledge acquisition and maintenance among individuals and groups of people, if not treated in an appropriate way, can make users to escape from KM system resulting in deserting of KM system. It goes without saying that capturing peer knowledge must be facilitated by intelligent techniques which should be built based on machine learning techniques. An agent can be initially trained by a subject matter expert and a knowledge engineer, in a way that is similar to how an expert would teach an apprentice, through problem solving examples and explanations. Once trained to a significant level of competence, copies of the agent can be installed to typical computer users. These agents then assist their users through his interaction with KM environment while he is recording his knowledge resulting in increase of speed and accuracy, without impeding their creativity and flexibility [17]. In the same time, the assistants continue to learn from this joint problem solving experience, adapting to their users to become better collaborators that are aware of users' preferences, biases and assumptions.

4.2 Agents and social aspect of knowledge management

Multiagent system is seen as a society of entities which define a structured pattern of behaviour that enhances the coordination of agent activities [1]. We consider KM as a social process and based on that we should make use of the complex social skills with which agents are endowed. Interactions in KM environments are fairly sophisticated, including negotiation, information sharing, and coordination.

As an example of interaction in a KM environment let's consider process knowledge acquisition which is intrinsically distributed, and needs coordination between process manager peer (e.g. agent) and other participating peers. The knowledge of process is distributed among peers and knowledge management environment must provide a sophisticated coordination mechanism to gather process knowledge for future reuse. For instance, agents can be coordinated using contraction protocols which is an often approach to coordination in agent societies. The process manager agent, which is responsible to force process workflow and to capture process knowledge, can bid a contract with each participating agent to capture the knowledge of their assigned job and send resulting knowledge back to process manager agent.

Another example of using agent capability in KM system is knowledge dissemination. Knowledge dissemination is usually done by query-answer conversation but is interaction between people limited to query-answer model? How do we use software programs to help in formalization of social mechanisms through which peers find (based on trust, recommendation, familiarity and organizational hierarchy) other peers to get involved in a conversation to proper discovery of resources. Some researches show that familiarity between participating people in a specific process can improve product quality.[20, 12]. We believe that agents can model some difficult aspect of human behaviour like familiarity and hierarchical distance [2] and also can get involved in complex dissemination process such as coalition making or referral networks [13, 18].

4.3 Agents and resolving different perspective

Communication and social interaction are always embedded in a social context. Distribution of knowledge among peers allows peers to use their own context while capturing their knowledge. In traditional systems knowledge acquisition has been done by forcing shared context. As we mentioned in previous sections, having different context by different peers can be considered as one of the main benefits of P2P knowledge management. In a very general way, context can be seen as a collection of things (parameters, assumptions, presuppositions, etc.) a representation depends upon. The fact that a representation depends upon these things is called context dependence. The basic intuition is that locally produced knowledge (personal knowledge or the knowledge of a group or department) cannot be represented in a universal structure because we cannot be sure that this structure is understood in the same way by different agents (people, groups or software agents). To integrate knowledge from different sources, a process of meaning negotiation is needed [14]. Integration of knowledge is therefore a mechanism of social agreement.

From a technological point of view, the representation of interpretive context and the analysis of meaning negotiation processes can be a valid basis for research in the intelligent agents' field. In particular, a distributed intelligence system within a technology enabled communication environment can take advantage of agents to be able to socially negotiate information under the light of locally represented knowledge.

5. KM Solutions: Discussion from knowledge management requirements point of view

In this section we describe six KM related processes and compare their requirements in the context of three different technological infrastructures. The design and implementation of a KM solution can be accomplished by using six processes [16]. These processes, that need not occur sequentially, are: *capture*, *transform*, *classify*, *maintain*, *discover* and *disseminate*.

- ***Capture***: The capture (acquisition) process brings the data about events of interest into the system by capturing information from projects and by collecting and interpreting information from sources inside and outside of organization. This can be pulled into the system before, during or after the event has occurred. Knowledge acquisition is the first stage of the knowledge engineering process. It is where a manual tool such as interviewing is used to actually extract the knowledge from a person with expertise in the problem domain. The way in which the knowledge is extracted from the expert, will affect the choice of the knowledge representation scheme used to organize the knowledge in the knowledge base (a form of database). It is also where the coding of the knowledge into the knowledge base actually takes place. Five stages in knowledge capture process are: identification which identifies the problem characteristics, conceptualization which finds concepts to represent the knowledge, formalization which designs the structure to organize the knowledge, implementation which formulates rules and frames to represent the knowledge and finally, testing which validates the rules that organize the knowledge.

- ***Transformation:*** During the transformation process, the information is attributed to its source, given context and validated, thereby making it easier to access, interpret and use. As a result of this process a package of material is produced that has been certified as important and that includes the perspective of firm's top expert. For each community we have an explicit representation of community's interpretation schema (or perspective or context). A context can be extracted in an automatic or semi-automatic way and to share knowledge, KM system should be able to transform knowledge from one context to another.

- ***Classification***: This process includes activities such as chunking, indexing, filtering and linking. As a result of this process, a classification scheme is developed and the new information is integrated and link with the existing content. A major challenge during classification is in deciding how to divide the inputs in to meaningful categories of knowledge objects. Each object should cover the context that will make it reusable in other settings while, at the same time, maintaining compactness for easy understandability. We believe that classification can be divided into two categories: static classification, which can be accomplished after knowledge acquisition or maintenance, and dynamic classification which is conceptually close to knowledge discovery and data mining. Based on this assumption we exclude classification from our comparison in next subsection.

- ***Maintenance:*** This process is a critical process which identifies the methods and policy of updating of knowledge bases. This process is to add and delete materials and maintain freshness and currency. A knowledge base without maintenance is worth nothing.

- ***Discovery:*** This process identifies information from the knowledge-base to make recommendation to different stakeholders in the organization. In addition to helping knowledge-base owners discover new relationships between the knowledge objects, discovery is particularly useful when functional units within organization have independently developed their own knowledge-bases. This process usually applies data analysis and discovery algorithms that within acceptable computational constraints produce useful patterns and knowledge from data.

- ***Dissemination:*** This process determines how people gain access to the content. The objective is to make it easy for people to find what they are looking for. The early uses of the knowledge-base might be described as a pull use, that is, information is pulled from knowledge base. This assumes that users know what they want and will get it using detailed searches. Another technique is to push the information. This technique is a proactive process and usually could be accomplished by personal agents or centralized process oriented knowledge management systems.

Tables 1-5 show a detailed comparison of three technological architecture regarding knowledge management requirements. Here we consider P2P as a general paradigm and opposed to the centralized structures which the Web based systems offer. Software agents can be implement either using independent agents infrastructure (e.g., JADE or FIPA_OS) or on top of P2P infrastructure (e.g., JXTA).

Table 1. Capture Process

	1. Capture (acquisition)
Web Based KM	Force the creation of knowledge manager role. KM solution is centralized and knowledge capture should be aimed toward objective knowledge. Shared interpretative schema must be extracted. This schema then will be forced to capture organizational objective knowledge resulting in loosing different perspective of different stakeholders.
Peer-to-Peer KM	Capturing peer knowledge should be completely distributed among peers (e.g. people or group of people). They can extract their own context and codify knowledge based on it. Knowledge acquisition is an extra activity which can cause deserting of KM system.
Agent Mediated KM	Agents can be used to facilitate context extraction and knowledge acquisition. Learning agent can learn context from an expert and then deployed in KM system to act on behalf of their owner to automatically extract knowledge. On the other method agents and capture knowledge from marked up documents. This can be considered as a semi-automatic process.

Table 2. Transformation Process

	2. Transformation
Web Based KM	Usually knowledge transformation is not a big issue. Structurally centralized KM systems consider objective knowledge and users often query knowledge based on pre specified interpretative schema.
Peer-to-Peer KM	In a distributed knowledge environment, KM system must provide a way for each community to make explicit its own interpretation schema (or interpretation or context). Having different context, each community (individuals, groups, departments) must be enabled to create relations with explicit context of other communities. If KM system does not want to fall back to an objectivistic view of knowledge P2P KM is required to transform knowledge from one context to another.
Agent Mediated KM	For human beings, knowledge sharing is often the result of a social process, in which many different cooperative strategies are used. This goes beyond the idea of simple context matching. AMKM should aim at reproducing of social process of negotiation for context transformation.

Table 3. Maintenance Process

	3. Maintenance
Web Based KM	Administration and maintenance of knowledge bases usually be done by knowledge manager role. Communication between knowledge administrators and knowledge producers has been major problem for maintenance. Maintaining freshness and currency of huge knowledge bases is a critical task which usually is a problem for centralized solution.
Peer-to-Peer KM	Knowledge is not a centralized asset. Distribution of administration and maintenance can help in reducing maintenance difficulties but it introduces knowledge acquisition problems in P2P paradigm. Updating knowledge bases can be a difficult task if knowledge representation mechanism is complex. Therefore maintaining text based knowledge bases seem easier comparing with other representation mechanisms (e.g. annotated knowledge assets or other specific knowledge model).
Agent Mediated KM	Maintenance of knowledge bases would be a difficult task if knowledge representation mechanism was complex. According to representation mechanisms, the knowledge maintenance difficulty can vary. Learning agents can facilitate maintenance process.

Table 4. Knowledge Discovery Process

	4. Knowledge Discovery
Web Based KM	Traditional knowledge discovery techniques have been developed for centralized knowledge repositories. Recently research communities are moving toward upgrading discovery techniques for distributed knowledge repositories. Discovery of knowledge and data mining in the centralized Web based solutions are easier to implement than distributed solutions.
Peer-to-Peer KM	P2P paradigm intrinsically helps in distributed knowledge discovery. Distributed knowledge discovery is expected to perform partial analysis of knowledge at peers and then to send the outcome as a partial result to other peer where it sometimes required to be aggregated to the global result. Knowledge discovery and mining algorithms can be implemented in application layer of P2P systems.
Agent Mediated KM	Learning agents can increase quality of discovered knowledge by allowing flexible knowledge discovery strategies, allowing collaboration between different knowledge manager agents through negotiation to come up with a jointly planned solution and increasing security and trustworthiness. [4]

Table 5. Dissemination Process

	5. Dissemination
Web Based KM	Because of ubiquitous accessibility, the Web is a unique infrastructure in knowledge dissemination. Using its technology in smaller scale (e.g. Intranets) has same advantages. Web usage is mostly restricted to query-answer conversation which support pull method. Personalized and focused web spiders have been introduced to increase the speed and efficiency of information retrieval. These spiders are client side programs and collaboratively understand users' requirements and adapt themselves with them.
Peer-to-Peer KM	Because knowledge is distributed its dissemination needs more effort in structural level. *P2P KM solutions should bring an environment in which users virtually view the chunk of distributed knowledge assets as a centralized knowledge repository*. Conceptually and technologically providing this environment could be the most challenging research issue. Conceptually it must provide transformation and technologically it must provide virtually centralized environment.
Agent Mediated KM	In AM KM agents are not client side agents. Here agents are deployed in server side and enable interactions between peers. For example, during a search, an agent can ask other agents to provide more information about a concept in their context (context matching), can remember past interactions and reuse them for similar queries, can redirect queries to other agents (referral networks), can get suggestions by other agents who get same query or can make a coalition to provide a set of relevant knowledge to another agent.

6. Conclusion

In this article we have discussed the distributed knowledge management, intelligent software agents and XML-based knowledge representation. Knowledge is inherently individualized, evolving and distributed and needs to be managed by a compatible technological architecture that supports individualized styles, distribution of resources and dynamic changes. There is no doubt that XML well suits the individualized representation style. Currently there are two paradigms cooperating and aiming at changing the face of knowledge management solutions from a rigid centralized architecture into a flexible and distributed one: the peer-to-peer paradigm and multi agent technology.

P2P paradigm provides basic communication protocols which is necessary to interchange knowledge between peers. It also provides group management capability in which discovery of other peers in community and locating them and routing between peers is accomplished. P2P implementations usually consider a robustness layer which allows individuals to implement security and reliability concerns. Class specific layer and application layer can be completely implemented by individual communities. From distributed computer system point of view, multiagent systems fall into the P2P category. A multiagent system can be built independently by agent specific structures (e.g., FIPA_OS) or on top of a pure P2P architecture (e.g., JXTA). In later case peers should be supplied with agent specifications.

Although peer to peer paradigm is a straightforward infrastructure to map technological architecture to organizational structure, switching from centralized paradigm to distributed one introduces major difficulties. Peer knowledge acquisition, social aspect of knowledge creation, maintenance and dissemination and resolving different perspectives are some of these difficulties which needs software agents which can be involved in complex interactions and learn. The orchestrated knowledge management based on the marriage of XML, P2P and agent technologies will be the prominent solution for KM in enterprises in the coming years.

References

[1] V. Dignum, "A model for organizational interaction: based on agents, founded in logic," SIKS Dissertation, series No. 2004-1.

[2] B.H. Far, W. Wei, and M. Afsharchi, "A Unified View of Software Agents Interactions," Transactions of Institute of Electronics, Information and Communication IEICE, Vol. E87-D, No. 4, pp. 896-907, April 2004.

[3] D. Fensel, S. Staab, R. Studer, F. Van Harmlen, and J. Davies, "A Future Perspective: Exploiting Peer-to-Peer and the Semantic Web for Knowledge Management," Towards the Semantic Web, Wiley, 2003.

[4] M. Klusch, S. Lodi, and G. Moro, "Agent-based distributed data mining: The kdec scheme," AgentLink, LNCS volume 2586, Springer, 2003.

[5] J. Hendler , E. Feigenbaum, "Knowledge is power: The Semantic Web", Web Intelligence 2001: 18-29

[6] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "Social networks: from the Web to knowledge management," in Web Intelligence, Springer Monograph 2003.

[7] J. Han, and K. Chang, "Data mining for Web Intelligence," in Web Intelligence, Springer Monograph 2003.

[8] P. Martin, "Knowledge representation, sharing, and retrieval on the Web.," in Web Intelligence, Springer Monograph 2003.

[9] Y. Sure, "On-To-Knowledge: Ontology based Knowledge Management Tools and their Application," in: German Journal (KI), Special Issue on Knowledge Management, vol .1, pages 35-37, 2002.

[10] D. Fensel, "Ontology-Based knowledge management," IEEE Computer, vol. 35, no. 11, 2002.

[11] A. Susarla, D. Liu, and A. Whinston, "Peer-to-Peer Knowledge Management," Handbook of knowledge management, vol. 2, chapter 39.
[12] D.B. Walz, J.J. Elam, and B. Curtis, "Inside a Software Design Team: Knowledge Acquisition, Sharing and Integration," Communication of ACM, vol. 36, no. 10, 1993.
[13] J. Denzinger, Knowledge-based Distributed Search Using Teamwork, Proc. ICMAS-95, San Francisco, 1995, pp. 81-88.
[14] M. Bonifacio, P. Bouquet, G. Mameli, and M. Nori, "Peer-Mediated Distributed Knowledge Management" Proceedings of AAAI Spring Symposium on Agent Mediated Knowledge Management (AMKM'03), Stanford University, 2003.
[15] M. Bonifacio, P. Bouquet, and P. Traverso, "Enabling Distributed Knowledge Management" Novatica and Informatik/Informatique, vol. III, no. 1, Feb 2001.
[16] P. Balasubramanian, K. Nochur, J.C. Henderson, and M.M. Kwan; "Managing process knowledge for decision support" Decision Support Systems, Vol. 27, November, 1999, p145-162.
[17] B. Stanescu, C. Boicu, G. Balan, M. Barbulescu, M. Boicu, and G. Tecuci, "Ontologies for Learning Agents: Problems, Solutions and Directions", Proceedings of the IJCAI-03 Workshop on Workshop on Ontologies and Distributed Systems, Menlo Park, CA, pp. 75-82, 2003.
[18] B. Yu, and M.P. Singh, "Searching Social Networks " Proceedings of Second International Joint Conference on Autonomous Agents and Multi-Agent Systems, pages 65-72, 2003.
[19] D. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu, "Peer-to-Peer Computing," HP Labs Technical Report HPL-2002-57, March, 2002.
[20] C.B. Seaman and V.R. Basili, "Communication and Organization in Software Development: An Empirical Study," IBM Systems Journal, vol. 36, no. 4, 1997.
[21] W. Nejdl, B. Wolf, C. Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmer, and T. Risch, "EDUTELLA: A P2P Networking Infrastructure Based on RDF". Proceedings of the World Wide Web Conference 2002 (WWW2002), Hawaii, 2002.
[22] V. Dignum, "An Overview of Agents in Knowledge Management", Institute of Information and Computing Sciences, Utrecht University, technical report UU-CS-2004-017, 2004.
[23] A. Tiwana, and B. Ramesh, "Integrating Knowledge on the Web", IEEE Internet Computing, v.5 n.3, p.32-39, May 2001.
[24] The DARPA Agent Markup Language, http://www.DAML.org
[25] J. Broekstra, M. Ehrig, F. van Harmelen, M. Sabou, R. Siebes, S. Staab, H. Stuckenschmidt, and C. Tempich "SWAP: Ontology-based Knowledge Management with Peer-to-Peer" On Workshop ontologiebasiertes Wissensmanagement WM 2003, Lucern 2003.
[26] CoMMA: http://www.si.fr.atosorigin.com/sophia/comma/Htm/HomePage.htm
[27] KRAFT: http://www.csc.liv.ac.uk/~kraft