Enabling Distributed Knowledge Management: Managerial and Technological Implications

Matteo Bonifacio, Paolo Bouquet and Paolo Traverso

In this paper we show that the typical architecture of current KM systems reflects an objectivistic epistemology and a traditional managerial control paradigm. We argue that such an objectivistic epistemology is inconsistent with many theories on the nature of knowledge, in which subjectivity and sociality are taken as essential features of knowledge creation and sharing. We show that adopting such a new epistemological view has dramatic consequences at an architectural, managerial and technological level.

Keywords: Semantic Interoperability, Distributed Architectures, Context, Agents, Epistemology, Managerial Models, Community Management.

1 Introduction

Knowledge Management (KM) refers to the process of creating, codifying and disseminating knowledge (e.g., in the form of documents, repositories, databases, procedures) within complex organisations, such as large companies, universities, and organisations for social and civil services. These organisations are typically structured in different components (e.g., departments, local companies, communities) that need to operate with a high degree of *autonomy*, and at the same time need some degree of *coordination* among each other to achieve common goals. Managing the processes of creating local knowledge within autonomous groups and exchanging knowledge across them is what we call *distributed knowledge management* (DKM).

Even though managers and consultants would certainly agree with the need to balance autonomy and coordination especially, in the domain of KM, we argue that current KM systems and architectures are designed in a way that is not consistent with this goal. This gap between ideal goals and concrete implementations will be described as rooted in a collection of epistemological assumptions (on the nature of knowledge) and of managerial assumptions (on the control of corporate processes) that are circularly interrelated. We show that these assumptions are incompatible with the goal of a distributed management of knowledge, as they rule out crucial factors such as the social nature of knowledge and its intrinsic subjectivity. Starting from the outline of a different epistemological framework, we propose a new high level architecture for enabling DKM, underlining the main implications in both the managerial and the technological domain. On the one hand, this means reconsidering traditional managerial assumptions on company processes; on the other hand, rather then new tools, it requires reconsidering the role of existing technologies such as text mining and intelligent agents within a new architectural frame.

2 Assumptions in Current KM Systems

In the last ten years, companies have been trying to design and develop systems to capitalise organisational knowledge. Even though the resulting systems use different technologies, tools and methodologies, they seem to share some typical features that we briefly describe (for a typical description of these features, see [Davenport et al. 98]):

- the installation of corporate-wide intranets in order to ensure physical and syntactical accessibility to information (i.e. connectivity and shared formats);
- the creation of knowledge-based systems, namely of semantically homogeneous and ubiquitously accessible repositories that contain corporate knowledge in its general and abstract form. This idea of a shared and unique point of access yielded the concept of *enterprise knowledge portal*, the most evident metaphor of such an approach;
- the creation/support of informal communities that represent the place where "raw" knowledge is produced through the spontaneous and emerging social interaction of company peers. In practice, these communities are materialised as "virtual communities" through the adoption of computer supported cooperative tools, such as groupware applications;
- the creation of a new role, the *Knowledge Manager*, whose goal is to support and facilitate the interaction across communities;

- the design of *corporate languages*, which are supposed to represent a company's knowledge in a standard and common way;
- the design of contribution processes which enable community members to explain their tacit knowledge through the codification in the corporate language.

These features are based on some general assumptions. In particular, we stress two of them:

- from a *structural* standpoint: though spread throughout the organisation, sometimes even at a implicit level [Nonaka/Takeuchi 95], organisational knowledge is something that can and should be centralised within explicit, shared repositories;
- from a *process* standpoint: even though knowledge is produced at a peripheral level as a specific and concrete matter (the so-called *tacit dimension of knowledge*, see [Polany 66]), its use requires that it is transformed into an abstract (i.e. independent from the production context) and general (i.e. applicable in any context) object.

Even though most studies show that, as far as knowledge is concerned, the social form of an organisation is similar to that in the left hand side of Figure 1, namely is a constellation of communities, each with its own languages, processes, tools, and so on (in a word, is a distributed system of local "knowledges"), most KM systems exhibit a completely different architecture (see right hand side of Figure 2), where the distribution is eliminated in favour of a centralised knowledge base (KB), which is used to feed the enterprise knowledge portal. The KB is normally built by creating a small task force within the organisation, whose aim is to specify the corporate language and knowledge map, and to design the way information is to be presented to users. This process is typically supported by tools like text miners, content management tools, and similar technologies, which allow the knowledge managers to couple the corporate knowledge map with existing and new incoming documents. Of course, there are variations of this general architecture, for example allowing some form of personalisation; a typical example is the use of personal assistants, whose goal is to support the single user (or group) to access in a personalised way to the corporate knowledge (a well-known example is the introduction of personal agents in the engine of Autonomy). However, these variations do not change the architecture in a significant way of creating the corporate KB is the process is basically the same.

Despite the claim of business operators that this architecture is the right answer to the needs of managing corporate knowledge, KM systems are often deserted by users, who instead continue to produce and share knowledge as they did before, namely through structures of relations and processes that are quite different from those embedded within KM systems (in [Bonifacio et al. 00], a paradigmatic experience which embeds the assumptions of this approach was concretely illustrated using the case of a worldwide consulting company). In the next two sections we will propose an explanation of why this is the case, and argue that the features described above, far from being necessary elements of a KM system, represent a very clear choice in terms of what knowledge is and, moreover, on the nature of managerial control.

2.1 Epistemological Assumptions

Despite the intention of supporting a subjective and social approach, the way most KM systems are designed reflects a marginal notion of sociality and embodies an objectivistic view of knowledge.

On the one hand, social aspects are still considered through a Hawthorn-like approach [Mayo 45], which is well rooted in western sociology. Essentially, this view considers sociality as a constraint to be dealt with in order to have operational and cognitive work done by people under the assumption that people work better if some level of informality and sociality is preserved and guaranteed in working environments. Under this light, communities represent the place where socialization occurs, a way of satisfying the social factors needed by people in order to operate, learn and share knowledge.

On the other hand, the subjective nature of knowledge, although commonly underlined, is viewed as a sort of primordial state rather then an intrinsic feature of knowledge. When knowledge is created, it is dependent on the context of production, such as the particular viewpoint of the individual or the daily practice of work, and thus not replicable. Therefore, this raw form of knowledge, called implicit by [Nonaka/Takeuchi 95] and tacit by [Polany 66], must and can be "cleaned up" (objectivised) from any contextual element, and be transformed

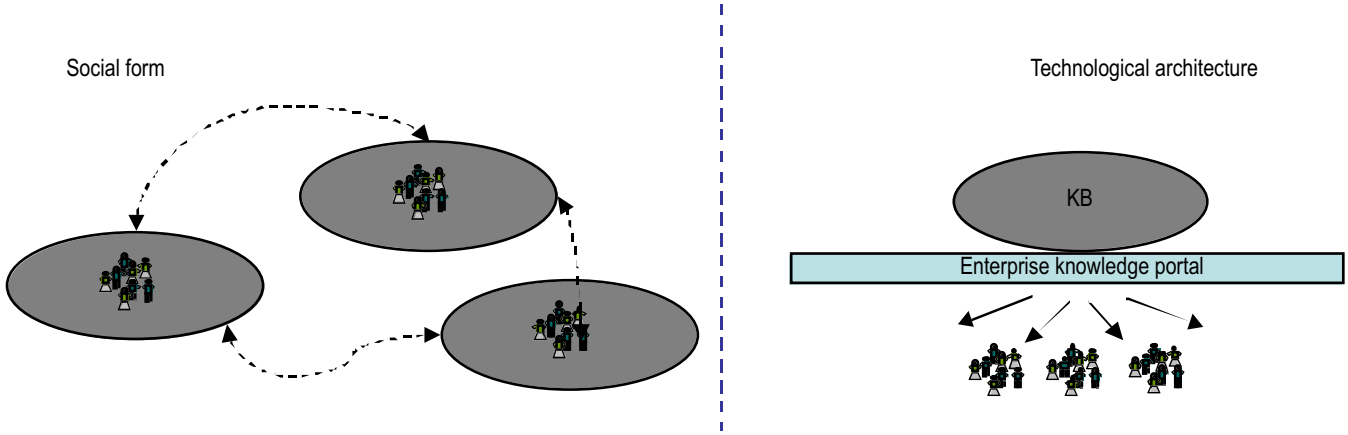

Figure 1: contrast between social form and technological architecture in the traditional approach

into an abstract (avulse from the original context) and general (applicable in any similar situation) form.

2.2 Managerial Assumptions

This epistemological view, which determines the organisational features of KM systems, is strongly related to a traditional paradigm of managerial control. This paradigm views management as a function grounded in the capacity of centralising the control on the company processes, such as resource and task allocation and monitoring. In particular, management allocates/distributes tasks and resources to employees and monitors the proper execution of tasks and use of resources. This traditional view on the managerial function can cope with KM only if an approach is taken which is compatible with the traditional paradigm; the processes of knowledge (*resource*) production and dissemination (*tasks*) need to be centrally driven (*allocation*) and controllable (*monitoring*). This condition is satisfied only if knowledge is thought of as an object, which can therefore be kept separate from the people that produced it. Otherwise, as far as knowledge remains embedded within subjective dimensions, it becomes a resource that falls outside the boundaries of managerial control.

It's hard to say whether it is the epistemological view that led to applying the traditional managerial paradigm to KM, or the other way around. Quite likely, they are in a circular relation in which the view on knowledge legitimates a managerial attitude, and the managerial attitude fits within the objectivistic view of knowledge. What is relevant for our purpose is that if the epistemological view becomes unrealistic, then the relationship between KM and the traditional managerial practices becomes problematic.

2.3 The Role of Technology

According to a structuration approach (see, for example, [Orlikowski/Gash 94]), technology cannot be considered a neutral matter with respect to organisational structures. Technological architectures can shape organisational forms, and organisational forms can affect the concrete appropriation of technology. This is more valid the more a technology is flexible and thus capable to be adapted to changing environments and to facilitate/anticipate the transformation of an organisational form. The experience with *groupware* applications in business processes re-engineering (generally considered a key technology for KM) shows the active role played by a flexible technology to stimulate organisational change. Therefore technologies, and in particular technologies for KM, far from being an independent variable, are a constraining and constrained force in play.

From this perspective, the relevance of a KM technology is primarily to be judged on the basis of architectural considerations, rather then on technical or functional features. IT architectures represent and embed the assumptions on how an information environment is "factorised" (described as composed by basic elements) and "processualised" (described through the processes and interactions that are possible among the different elements). In other words, IT architectures can be seen as the informational lens through which an organization can be red, described and modelled. Therefore we can judge whether a technology is consistent or inconsistent and capable of shaping or being shaped by an organizational form only at an architectural level.

From what we said, it follows that we should be able to identify the architectural features of current KM technological architectures that reflect the epistemological view described above. And in fact, if we look at the way technology is used to build the systems, we see that:

- content management tools (text miners, search engines, and so on) are used to produce a single, shared view of the entire collection of corporate documents. This is meant to create a common and explicit (e.g. taxonomies, ontologies, category systems) or implicit (e.g. clusters, patterns) interpretative schema. Corporate portals represent this common organisational view on the world;
- new standard formats (like HTML, XML, PDF, and so on) are introduced in order to reduce syntactic heterogeneity of documents from different knowledge sources. This is meant to provide physical access to documents, though this completely disregards the possibility that documents from different knowledge sources may also be semantically heterogeneous;
- chat forums and discussion groups are used to satisfy the need of social interaction, but do not provide a real support to the consolidation and exchange of socially produced knowledge.

Through this perspective, we underline that a different epistemological view on knowledge requires to rethink not only the managerial function, but also, as we will see later, the technological architectures that are implemented in order to support a KM system.

3 A Different Epistemology for KM

Most authors who stressed the subjective nature of knowledge argued also that meanings are not externally given; rather, individuals give meaning to situations through subjective interpretation. Interpretation is subjective, since it occurs according to some "internal" interpretation schema, not directly accessible to other individuals. These schemas have been called, for example, mental spaces [Fauconnier 85], contexts [Giunchiglia/Ghidini 00], or mental models [Johnson-Laird 92]. Internal schemas can be made partially accessible to other individuals only through language, since language is not just a means to communicate information, but also a way of manifesting an interpretation schema. As a consequence, when interpretation schemas are deeply different, people will tend to give a very different meaning to the same facts. Conversely, in order to produce similar interpretations, people need to some extent to share interpretation schemas, or at least to be able to make some conjectures on what the other people's schema is. Shared schemas have been called in many different ways, for example paradigms [Kuhn 70], frames [Goffman 74] and thought worlds [Dougherty 92], and the process of "viewing" and reasoning about other people's schemas has been called perspective taking [Boland/Tenkasi 95] or double loop learning [Argyris/Schon 78]. Since we are talking about organisations,

and thus about a collective level, what is relevant for our purpose is that without this inter-subjective agreement (or at least believed agreement) communication cannot take place, coordinated action is impossible, and meaning remains confined just at an individual level [Weick 93]. From an epistemological point of view, this approach leads to some significant consequences:

- knowledge is intrinsically subjective, as the meaning of any statement is always dependant on the context [Giunchiglia/Ghidini 00] or schema of the interpreter;
- at a collective level, groups of people can assume they share (or have a reciprocal view on) some part of their different schemas, and therefore they share (or understand) the other's meanings. Nonetheless meaning "lives" in the inter-subjective agreement of different individuals. Thus collective knowledge is social, since sociality is the precondition for something to be a meaning.

As a result, the notion of Knowledge as an absolute concept that refers to an ideal objective picture of the world leaves the place to a notion of "local knowledge", which refers to the different, partial interpretations of portions of the world or domains that are generated by individuals and within groups of individuals (e.g. communities) through a process of negotiating interpretations. At an organisational level, each local knowledge appears as the synthesis of both a collection of statements, and the schemas that are used to give them a meaning. A local knowledge is then a matter that was (and is continuously) socially negotiated by people that have an interest in building a common perspective (*perspective making* for [Boland/Tenkasi 95] or *single loop learning* for [Argyris/Schon 78]), but also in understanding how the world looks like from a different perspective (*perspective taking* for [Boland/Tenkasi 95] or *double loop learning* for [Argyris/Schon 78]). Therefore, rather then being a monolithic picture of the world as it is, organisational knowledge appears as a heterogeneous and dynamic system of "local knowledges" that live in the interplay between the need of sharing a perspective within a community (to incrementally improve performance) and of meeting different perspectives (to sustain innovation).

4 A Distributed Architecture for KM Systems

The aim of this section is to draw some consequences on how an architecture for a KM system should be designed to be consistent with the distributed social form in which knowledge is created within organisations. The architecture we discuss in this section, depicted in Figure 3, is under development as part of a joint project of the Institute for Scientific and Technological Research (IRST, Trento) and of the University of Trento, called EDAMOK (*Enabling Distributed and Autonomous Management of Knowledge*) in which the authors of this paper are involved.

As we said, the main limitation of a centralised architecture (Figure 1, right hand side) is not technological, but organisational. It creates a mismatch between social form and technological architecture, and this often produces a non acceptance by users. Our alternative proposal is based on three main ideas.

The first idea is that knowledge should be *autonomously* managed where it is created and used, namely within each community. Autonomy means, for example, that each community should be allowed to build its own local knowledge map, and choose whatever tools and processes are more appropriate to manage the life cycle of documents and other data. From an architectural point of view, this means that each community is allowed to build its own "local application" (see Figure 2) for managing local knowledge. This autonomy is a necessary condition for local intelligence to be exploited at its best, as it produces strong ownership on each community's knowledge.

The second idea has to do with coordinating many autonomous knowledge sources. Indeed, *autonomy* without *coordination* is almost useless for knowledge sharing in complex organisations. In traditional KM systems, coordination is achieved by creating common languages and schemas (see section 2 above) which are then used to categorise all corporate knowledge; however, we already discussed the theoretical and practical limitations of such an approach. In a distributed system, coordination should be reached through interoperation rather than centralisation. Our way to semantic interoperation is based on two steps:

- on the one hand, the system must provide a way for each community to make (at least partially) explicit its own interpretation schemas. This explicit representation of a community's interpretation schema (or perspective) is what we call a *context*. A context can be "extracted" in different ways. Of course, a community can create its context from scratch (e.g., through a context editor): However, much more interesting is the possibility of extracting contexts in an automatic or semiautomatic way; for example, one can provide tools for translating category structures already in use in a community (e.g. local database schemas or taxonomies of content management tools) into the format of the *context description language*; or for extracting the context from the documents created or collected in a community's repository;
- on the other hand, each community must be enabled to create relations with explicit contexts of other communities. If we don't want to fall back to an objectivistic view of knowledge (e.g., by requiring that each local context is translated/mapped into a centrally built, shared knowledge map), we need to allow the creation of (partial) mappings from context to context. In our work, contexts are represented as directed acyclic graphs in which both nodes and arcs are labelled. Nodes represent concepts, and are labelled with terms from natural language; arcs represent relations between concepts and are labelled with a type (e.g., is-a or part-of). Intuitively, a mapping between two contexts is a "link" between a concept in the first context and concepts of the second, for example, a link between the concept of "Venice" in a context and "the city of San Marco" in another. Again, these mappings can be defined by hand, but a KM system should support the creation of automatic (or semiautomatic) mappings. This requires the ability to compare, in a semantically relevant way, autonomously generated contexts. From a technological point of view, this can be done by implementing algorithms of context matching, for

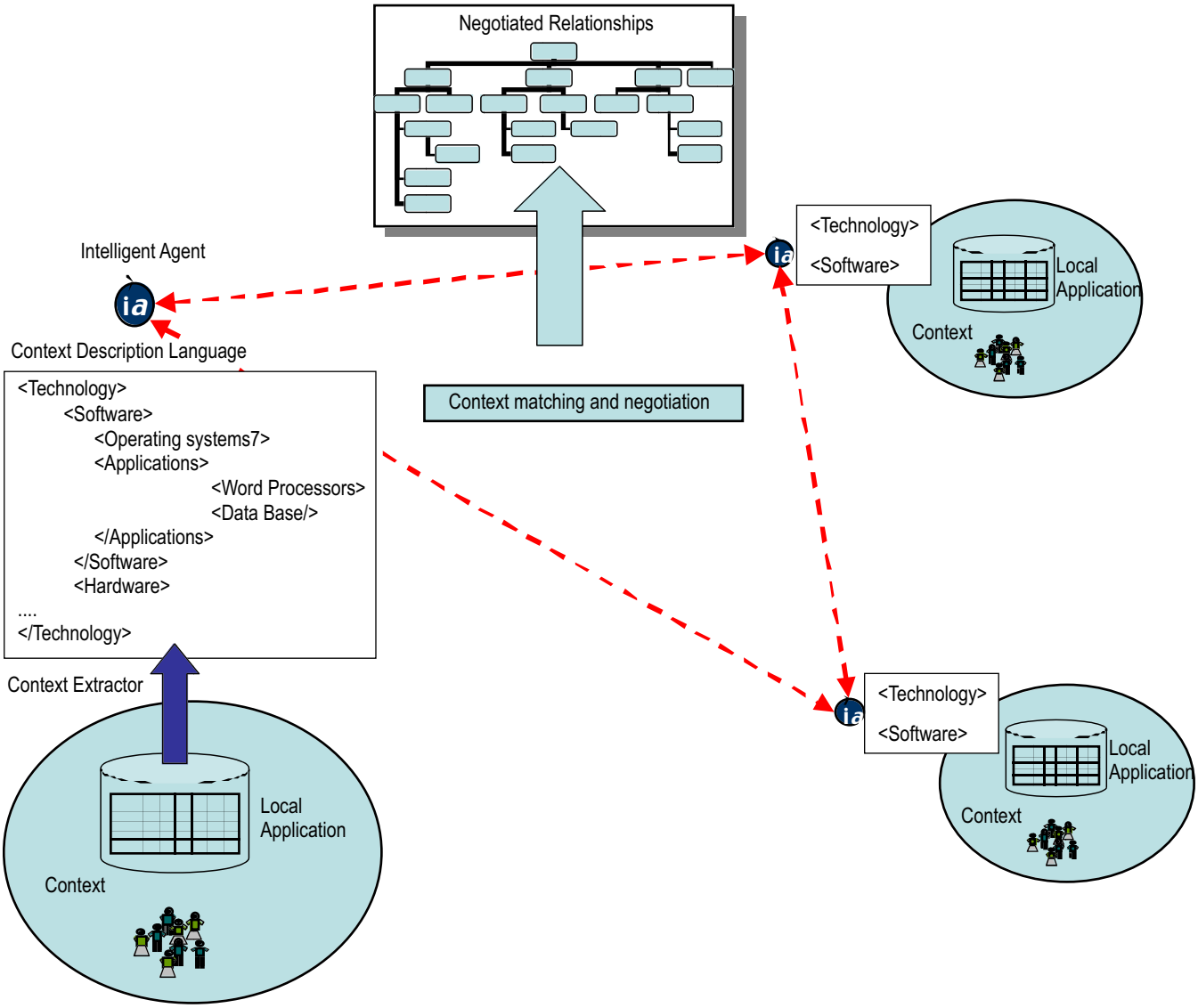

Figure 2: the distributed approach

example, using graph matching combined with natural language processing techniques to suggest possible mappings).

However, context matching is not enough. For human beings, knowledge sharing is often the result of a social process, in which many different cooperative strategies are used. A typical example is a query-answer conversation, in which someone tries to understand what the interlocutor really means with a request (e.g., a question like "Do you have any information about Venice" can be interpreted in many different ways, depending, among other things, on the speaker's goals). This goes beyond the idea of simple context matching. The third idea underlying the proposed architecture is that the system should aim at to reproducing this social process. This is what we call *meaning negotiation* in Figure 2. In our architecture, meaning negotiation is implemented via communication protocols between autonomous agents. In the architecture a *software agent* is associated to each community. The role of agents is twofold. On the one hand, each agent "knows" the explicit context of its community; on the other hand, through a process of query-answer, agents can improve the simple idea of context matching. For example, during a search, an agent can ask other agents to provide more information about a concept in their contexts (to improve the result of context matching), can remember past interactions and reuse them for similar queries, can be redirected to other agents, can get suggestions by overhearers (i.e. other agents that overheard the request); and so on.

An important advantage of this architecture is that it reproduces the dynamic of social interaction at the system level. This provides a high degree of flexibility and scalability. An intuitive extension is the following. It is well-known that, in complex organisations, a crucial role is played by those who act as a bridge between organisational units, groups and communities. This role (sometimes called *brokering*) is well known in social

network analysis and community management, as it enhances the ability of a social system to exchange knowledge across semantic boundaries [Wenger 98]. In the proposed architecture, we can easily introduce brokers as a new type of agent whose contexts are "road maps" to the other contexts available in the system. Thus, brokers may facilitate the exploitation of synergies between communities that, even though working in different contexts, could fruitfully work together. Notice that brokers can become also repositories of "negotiated relations" (see top of Figure 2), namely collect and remember relations that other agents have successfully negotiated in previous interactions and combine them in innovative ways.

5 Conclusions

The proposed architecture has many important implications, both from a managerial and technological perspective.

From a managerial standpoint, a distributed approach to KM poses fundamental questions about the nature of the managerial function. First of all, the supportive role played by knowledge managers and claimed by organisational experiences, needs to be brought from the aesthetics of a "nice to be" to a substantial acceptance of a loss of control. In the domain of knowledge, control is to be reinterpreted as a bottom-up process where emerging communities legitimate the "cognitive leadership" of knowledge managers rather then a top-down process of organising "cognitive activities" of people. Moreover, knowledge managers should be the internal expression of a community rather then being assigned from outside.

Second, managers should abandon the dream of having a unique and homogeneous materialisation of "knowledge" represented as an asset in some knowledge base, such as a balance sheet is for financial assets. Knowledge is intrinsically distributed, embedded and localised within the context of informal communities, where context is the fundament of meaning and sense making and thus, from an organisational standpoint, value.

Third, communities, when viewed as the ground where knowledge is created and resides, become centre of power and interest that negotiate organisational paradigms and resources. A community, which is a local knowledge, fights for its survival through the acquisition of resources in order to sustain the "evangelisation" of other communities according to its perspective. In this sense, a community represents not just an opportunity for its valuable knowledge, but also a threat for its natural tendency to create the conditions for its existence. Existence that, being based on knowledge, means forcing that knowledge to be "true".

Fourth, the emphasis on homogenisation as the main process to ensure control (the reduction of complexity through the elimination of differences) should give way to concepts such as "interoperation" and "coordination" among different and autonomous actors. In fact autonomy and heterogeneity are not just intrinsic to the notion of knowledge, but are also the basis of innovation that has been described as the moment in which different perspectives meet, this way generating a discontinuity in traditional and incremental organisational learning paths [Boland/Tenkasi 95], [Argyris/Schon 78]. The assumption of homogenisation as a KM fundamental process leads to the contradiction that knowledge is maximised when innovation (one fundamental organisational learning process) is minimised. On the one hand, managers should support the growth of different perspectives as a premise for innovation; on the other hand, they should seek for coordination providing processes and tools to support systematic interoperability.

From a technological standpoint, distributed architectures presuppose the explicit recognition of the distributed nature of knowledge. Distributed architectures sustain autonomy at different levels: technological (different groups may use different technologies), syntactical (different groups may use different information formats) and, most of all, semantic (different groups may generate different systems of meaning). From a group's or a community's perspective, a distributed system supports the exploitation and representation of a community's context that stands for the layer upon which a community's members produce and negotiate a common semantics. Contexts can be represented as local ontologies, taxonomies and, in general, as theories through which community members interpret their environment and make sense of organisational events. From this perspective, text mining tools, for example, instead of being used as a means to produce a corporate wide homogeneous classification of documents, can be seen as domain specific applications that are able to serve the specific needs of each community. Text miners become a community technology that, through use, circularly learns a community context from that community's documents, and learns how to classify documents on the basis of a community's context.

From an inter community standpoint, a distributed architecture facilitates knowledge sharing without assuming company wide shared meanings, and, as a consequence, naturally supports the genesis of innovation. Moreover, since it is more respectful of each community's interpretation schemas, it is more likely to be adopted and used by people. Indeed, it can cope with the need expressed by each community to protect and preserve its specific way to organise and use information.

If semantic interoperability ensures information sharing without the need of a shared semantics, the reduced need of sharing tools and application is dramatically facilitates technological interoperability. Implementing a shared way to semantically organise information (homogenisation) is a strong constraint on what technologies can be installed within an organisation. IT managers know how difficult is to balance group specific technological needs with the need to maintain a common information environment. As a consequence technological choices have been addressed towards general purpose content management platforms that were able to offer some set of differentiated functionalities in order to match some group's need, and at the same time to offer a shared environment in order to ensure integration. The problems arise when group specificity grows enough to overcome the platform's ability to manage differences and, moreover, when companies are faced with situations that require the integration of different platforms (see merge and acquisitions). Indeed, if one considers the huge investments required, centralised solutions manifest their irreversibility and a lack both in maintainability (for example a

change in classification schemas issued by a local group requires a change in the overall structure) and scalability (platforms are scalable to the extent that the same platform is assumed to be used). We claim that distributed architectures, rather then being an expression of IT anarchy, are the best way to balance group specificity (each group can use the appropriate local technology) and shared environment (through interoperability), ensuring reversible investments (disinvesting a technology doesn't mean disinvesting the overall system), system maintainability (a change can be managed locally), and scalability (the addition of a component or technology is compatible by default).

References

[Argyris/Schon 78]
C. Argyris and D. A. Schon: Organizational Learning: a Theory of Action Perspective. Reading Mass, Addison-Wesley 1978

[Boland/Tenkasi 95]
R. J. Boland and R. V. Tenkasi: "Perspective Making and Perspective Taking in Communities of Knowing", Organization Science, 6(4), 350–372, 1995.

[Bonifacio et al. 00]
M. Bonifacio, P. Bouquet and A. Manzardo: "A Distributed Intelligence Paradigm for Knowledge Management". AAAI'2000 workshop on "Bringing knowledge to business processes", Stanford University, March 2000.

[Davenport et al. 98]
T. Davenport, D. W. DeLong and M. C. Beers: "Successful Knowledge Management Projects". Sloan Management Review, Winter 1998.

[Dougherty 92]
D. Dougherty: "Interpretative Barriers to Successful Product Innovation in Large Firms". Organization Science, 3, 1992.

[Fensel et al. 00]
D. Fensel, I. Horrocks, F. Van Harmelen, S. Decker, M. Erdmann, M. Klein: OIL in a nutshell. Proceedings of the European Knowledge Acquisition conference (EKAW-2000), Dieng et al. (eds.), Lecture Notes in Artificial Intelligence (LNAI), Springer-Verlag.

[Fauconnier 85]
G. Fauconnier: Mental Spaces: Aspects of Meaning Construction in Natural Language 1985. Bradford Books, MIT Press.

[Giunchiglia/Ghidini 00]
F. Giunchiglia and C. Ghidini: "Local Models Semantics, or Contextual Reasoing = Locality + Compatibility", Artificial Intelligence, 127(2), p. 221–259, 2000.

[Goffman 74]
I. Goffman: Frame Analysis. Harper & Row 1974, New York.

[Johnson-Laird 92]
P. Johnson-Laird: Mental Models. Cambridge University Press 1992, Cambridge.

[Kuhn 70]
T. S. Kuhn: The Structure of Scientific Evolutions. Chicago 1970, University of Chicago Press.

[Mayo 45]
E. Mayo: The Social Problems of an Industrial Civilization. Boston, Graduate School of Business Administration 1945, Harvard University.

[Nonaka/Takeuchi 95]
I. Nonaka and H. Takeuchi: The knowledge creating company 1995. Oxford University Press.

[Orlikowski/Gash 94]
W. Orlikowski and G. Gash: "Technological Frames: Making Sense of Information Technology in Organizations", ACM Transactions on Information Systems 1994, 12,2.

[Polany 66]
Polany: The tacit dimension, Routledge and Keel Paul, 1966

[Starr/Bowker 99]
S. L. Starr and G. Bowker: Sorting Things Out: Classification and its Consequences 1999. MIT Press.

[Weick 93]
E. K. Weick: "Collective Mind in Organizations: Headful Interrelating on Flight Decks", Administrative Science Quarterly 1993, 38

[Wenger 98]
E. Wenger: Communities of Practice. Learning, Meaning and Identity. Cambridge University Press 1998.