ELSEVIER

Interacting with Computers 15 (2003) 57–89

Interacting with Computers

www.elsevier.com/locate/intcom

Using conversational agents to support the adoption of knowledge sharing practices

Claudia Roda[1], Albert Angehrn[1], Thierry Nabeth*[,1], Liana Razmerita[1]

INSEAD—Centre for Advanced Learning Technologies, Bd. De Constance, F-77300 Fontainebleau, France

Received 30 April 2001; revised 30 November 2001; accepted 29 June 2002

Abstract

In this paper, we present an agent-based system designed to support the adoption of knowledge sharing practices within communities. The system is based on a conceptual framework that, by modelling the adoption of knowledge management practices as a change process, identifies the pedagogical strategies best suited to support users through the various stages of the adoption process. Learning knowledge management practices is seen as a continuous process, taking place at individual and social level that includes the acquisition of information, as well as the contextual use of the information acquired.

The resulting community-based system provides each member of the community with an artificial personal change-management agent capable of guiding users in the acquisition and adoption of new knowledge sharing practices by activating personalised and contextualised intervention.

Keywords: Software agents; Knowledge sharing; Virtual communities; Knowledge management; Change management; User modelling

1. Introduction

The issue of how to help communities of people work together, facilitating the exchange of their knowledge, enabling learning, and overall increasing their ability to achieve individual and collective goals, has been the target of innumerable theoretical

* Corresponding author. Tel.: +33-16-072-4312; fax: +33-16-074-5550.

E-mail addresses: thierry.nabeth@insead.edu (T. Nabeth), claudia.roda@insead.edu (C. Roda), albert.angehrn@insead.edu (A. Angehrn), liana.razmerita@insead.edu (L. Razmerita).

[1] http://www.insead.edu/calt/.

0953-5438/03/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0953-5438(02)00029-2

research and practical projects (amongst the most recent ones: Angehrn et al. (2001), DeSanctis et al. (2001), Dyer and Nobeoka (2000), Gongla and Rizzuto (2001) and Lesser and Storck (2001)).

It is generally recognised that communities perform better when all members adopt certain behaviours, such as sharing their knowledge or making effective use of the knowledge produced by others. Those behaviours that facilitate the effective functioning of communities are often referred to as 'knowledge management' behaviours/culture/-attitude and many organisations have recently invested considerable resources to ensure that all their members will adopt them.

Unfortunately, many organisations have found that ensuring the adoption of knowledge management behaviours is not an easy task (Beer and Nobria, 2000). This may be due to many different factors. For instance, some community members may not want to adopt certain knowledge management behaviours because they feel that it would be against their interests to do so, e.g. one may not want to share his/her knowledge. Other members may not be capable of using the tools required to implement such behaviours; e.g. one may not know how to retrieve the information stored in a database. Some members may simply not be aware that they are expected to behave in certain way, e.g. one may not know that he/she is expected to use other people's results when working in a given team. Some may simply judge the overhead associated to knowledge management behaviours disproportionate to the advantages that they would gain.

Our work addresses the problems above by showing that it is possible to build tools that help people to become acquainted with, understand, and eventually adopt those behaviours that improve the overall performance of a community. We provide an operational model of the cognitive, social and cultural factors that influence the acquisition of knowledge management behaviours and we describe how this model is implemented in the multi-agent system knowledge-intelligent conversational agents (K-InCA). In particular, we discuss the adoption of knowledge sharing behaviours because they are vital for good knowledge management and often encounter the strongest resistance amongst community members.

In order to help people become acquainted with, understand, and eventually adopt knowledge sharing behaviours, K-InCA bridges two types of systems: those supporting learning and those supporting community-based knowledge exchanges. Most of the former systems are designed to 'teach' some concepts but fall short of supporting the user in applying the concepts in a 'real world' environment; they normally reflect the school model. The latter systems are generally designed to facilitate the 'practice' of some knowledge management behaviours, however, they rely on the assumption that the user has already adopted the behaviours and wants to use the tools supporting those behaviours. K-InCA, instead, aims at providing personalised guidance throughout the whole adoption process: from the introduction of the behaviours to the user (explaining what the desired behaviours are and why they should be adopted) to their practice within the community.

K-InCA relies on the idea of offering personalised guidance to each user. The system observes the user's actions and, whenever appropriate, it makes suggestions, introduces concepts, proposes activities and, in general, encourages the user in the adoption of the desired knowledge sharing behaviours. These behaviours, and the tools that may facilitate their practice, are introduced accordingly to the users' characteristics. K-InCA is in fact capable of selecting amongst several pedagogical approaches on the basis of the evaluation of factors, such as the users' acquaintance with the behaviours, their learning preferences, their social position within the community and their current activities.

The agent-oriented design of the K-InCA system has allowed us to incrementally add new, autonomous, interactive, proactive, reasoning components responding to an increasingly wide range of users' needs and users' characteristics. The system is composed of several agents each fulfilling a different role, such as interacting with the user, diagnosing the user state, implementing pedagogical strategies, presenting tools. Autonomy, modularity and proactivity make the agent-oriented design a particularly suitable approach to the problem of adopting knowledge management behaviours. The agent-oriented paradigm also allows the system to be more easily distributed over computer networks, such as those normally available within organisations. The system presented can be used, for instance, to provide virtual community platforms with a mechanism that helps improve people's participation, or by organisations to help foster people's collaboration.

The main contribution of this paper is to show that it is possible to create tools supporting all aspects of the learning process: from the acquisition of information to the use of the information acquired, from individual learning to learning as a social process, from passive learning to active selection of one's learning objectives and priorities.

The paper is structured in three main parts. First we introduce the conceptual framework that is used to model people's adoption of knowledge management practice as a change process and we discuss how this process can be facilitated by appropriate pedagogical interventions. Then, we describe the agent-oriented K-InCA system that makes the framework operational by implementing the pedagogical intervention necessary to guide users in the adoption of knowledge sharing behaviours. We conclude by describing how our system relates to existing systems and by summarising the different findings of this approach.

2. Supporting the adoption of knowledge management practices: a conceptual framework

The objective of this section is to present the conceptual framework of K-InCA. From this framework, we derive the operational model of the interventions that should be activated to accelerate the adoption of knowledge management practices. In particular, we explain how knowledge management studies may help identifying what makes a community perform better and we define the type of communities addressed by our framework. We then represent the process undertaken by individuals in adopting knowledge management behaviours as a *change process* and we explain how this change process may be facilitated through a set of intervention strategies acting at cognitive, social and cultural levels.

2.1. Knowledge management in organisations

Recently, several fields of research, notably organisational management and information technology, have converged towards the study of what has become known as knowledge management. These studies often identify the behaviours that members of a community should adopt in order to improve the overall performance of the community. These are called knowledge management behaviours (or knowledge management culture, or knowledge management attitude). Knowledge management studies also indicate how these behaviours can be encouraged, facilitated, and supported. For example, efficiently sharing one's knowledge with other community members may be defined as a knowledge management behaviour that can be supported by providing good communication tools. Knowledge management advocates have demonstrated that the implementation of knowledge management behaviours increases the productivity of organisations, and improves their ability to transform themselves and adapt to their environment (Davenport and Prusak, 1998; Dore, 2001; Grant, 1996; Koulopoulos et al., 1997; Leonard-Barton, 1995; Nonaka and Hirotaka, 1995; Teece et al., 1997).

Notwithstanding (or perhaps, because of) the enormous amount of academic publications, consulting firms, implementation projects and 'experts' on knowledge management, no commonly agreed-upon definition of knowledge management exists. Some authors emphasise the processes needed to create an organisational memory. For example, Stuart (1996) defines knowledge management efforts within organisations as "intended to retain, analyse and organise employee expertise, making it easily available anywhere, any time". Other authors concentrate on the human processes that allow efficient exploitation of the available information. For example, Malhotra (2000) sees knowledge management as embodying "organisational processes that seek synergistic combination of data and information processing of information technologies, and the creative and innovative capacity of human beings". Along the same line, Davenport and Prusak (1998) stress how people are the main repositories of knowledge.

Concerning tools supporting knowledge management processes, Manville (cited in Malhotra (2000)) gives us an important hint about the requirements for systems assisting the work of communities: they should "support competencies for communication building, people networks, and on-the-job learning". Note that Manville stresses not only the availability of tools for communication, networking or learning, but also the need to support *competencies* enabling communication, socialisation and learning. Our framework was designed with the objective of supporting the *competencies for* knowledge management (rather than just supplying the tools), and because of this emphasis on competencies it differs from many of the existing community-based software.

Some authors have indicated that knowledge management processes are most effective when tailored to the community's organisational structure and goals (Want, 1995). For this reason, it is important to understand those elements determining the community structure, such as shared values, responsibilities, roles and duties. Our research concentrates on *organisations*, i.e. communities where the above elements are clearly identifiable. Examples of organisations are large, medium and small size business, some charities, political parties and some social intervention groups and civic networks.

Organisations are also characterised by the fact that they have commonly understood goals. In this sense, they can be seen as *communities of purpose* in that they "are composed of people who typically share a common desire to forward the interests of the organisation as a whole" (Carotenuto et al., 1999). If however, we observe organisations at a finer granularity level, they are characterised by a set of foci on multiple subjects (such as:

products, customer relationship strategies, projects, social intervention activities) spanning both interests and activities. At this level, a complex social structure including many sub-communities becomes apparent (Sumner et al. (1999) describe organisations as typically "composed of multiple interacting communities, each with highly specialised knowledge, skills, and technologies"). These sub-communities may be distinguished as communities of practice (Wenger, 1998), communities of interests, communities of purpose, etc. (for a description of such communities, see Carotenuto et al. (1999)). Sub-communities may emerge spontaneously within the organisation (e.g. a sub-community of practice may emerge from the interaction of a group of salesmen within a corporation) or they may be purposely created (e.g. the board of governors may form a sub-community of purpose).

As discussed later, understanding the social dynamics related to sub-communities allows us to support better the adoption of knowledge management practices.

2.2. The change process

A recent survey on the implementation of knowledge management practices in banks and insurance companies (Dore, 2001) reports that the main barriers to knowledge sharing are the "lack of understanding of the benefits derived from knowledge sharing" and the 'technology inadequacies' due to the fact that 'knowledge is held in too many formats and repositories'. Some IT literature has also identified organisational and behavioural change management as a critical success factor in the implementation of knowledge management (Alavi and Leidner, 1999; Vandenbosch and Ginzberg, 1996–1997).

We believe that the lack of understanding of benefits discussed by Dore can be addressed precisely by supporting behavioural change processes aimed at motivating community members towards the adoption of knowledge management practices. The analysis proposed by Dore tells us that users can be supported through the change process by addressing two needs. (1) Community members must be informed about the benefits of knowledge sharing and be offered occasions to experience the advantages themselves. (2) Users must be supported by a system capable of matching their needs with the information available within the community (or accessible outside the community). The former need is our main concern and it can be described in terms of innovation diffusion within the organisation (O'Reilly and Tushman, 1997; Rogers, 1995).

We describe users as undergoing a change process that brings them from their old practices to the conscious adoption of knowledge management practices (e.g. transition from low or non-existing levels of knowledge sharing practices to the widespread adoption of best behaviours in knowledge sharing). This process of change must be managed to ensure that all members of the community smoothly adopt the new behaviours.

In an extensive review of the literature on change management models (Angehrn and Atherton, 1999), we have summarised the critical ingredients needed to implement a successful change management program as a four stage cycle of *visioning*, *planning*, *implementing*, and *reviewing and learning*. The *visioning stage* defines the new behaviours and practices envisioned for the organisation, here we call these behaviours and practices the *change domain*. The user's *change state* is a description of the level of adoption of the behaviours and practices included in the change domain, i.e. the change state defines how far or close a community member is to the adoption of the desired behaviours. During the *implementation stage*, the community members should move from their initial change state (e.g. complete ignorance of all the desired behaviours) to a change state where all behaviours in the change domain have been adopted. The *intervention model*, developed in the planning stage, describes the strategies employed to bring about this state transition.

K-InCA has been designed to support the *implementation* stage of the change process; i.e. the system is designed to support people's progressive adoption of the behaviours defined in the change domain. In the rest of the paper, every reference to the change process is meant as a reference to the implementation stage of the change process.

We define a change process as a sequence of change operations upon user states, leading to an end (the acquisition of the desired behaviours). In particular, we define the change process following the model proposed by Rogers (1995) in which "an individual [...] passes (1) from first knowledge of an innovation, (2) to forming an attitude towards the innovation, (3) to a decision to adopt or reject, (4) to implementation of the new idea, and (5) to confirmation of this decision" (Rogers, 1995, p. 161). This model of the innovation–decision process rests on research evidence that the identified stages exist (Rogers, 1995, chapter 5).

Borrowing Near's (1993) terminology and mapping it into Rogers' theory (see Angehrn and Nabeth (1997) and Manzoni and Angehrn (1998)) the following user states can be identified: "The first stage is *awareness*, in which the individual is alerted to the existence of something new. Next is the *interest* stage, in which the individual gathers information and an aroused level of curiosity. This is followed by the *appraisal/trial* stage, in which the new idea is tried out in a trial operation. The final stage is *adoption*, in which the individual incorporates the innovation as a part of the resources he or she uses on-the-job" (Near, 1993). See Fig. 1.

Initially, individuals become *aware* of a new behaviour (for instance, the behaviour 'acknowledge sources of information'); this corresponds to the achievement of phase (1) of Rogers' model. Becoming aware of a behaviour implies knowing what the behaviour is, how it can be practised, what are the advantages or disadvantages of the behaviour, who can help in the adoption, and what are the main repositories of information for the given behaviour. The awareness state is a passive state that may resemble some classic learning phases in which a teacher, or a book, supply a set of basic information on a given subject to a learner.

Based on the information acquired while passing from the ignorance state to the awareness state, community members can form a positive or negative attitude towards the innovation being introduced and they may move to an interest state. This corresponds to the positive achievement of phase (2) of Rogers' model. When in this state, people actively seek information that is often much more specific and situated than the information obtained in the awareness phase. This state is also characterised by an active and positive response to stimulus received regarding the new behaviour, e.g. willingness to participate in conversations about various aspects of the behaviours.

Once individuals have collected enough information, they are in the position to take the decision whether they should try to practise the new behaviour. This corresponds to phases (3) and (4) in Rogers' model. If the decision is favourable, then they enter the *trial* phase.

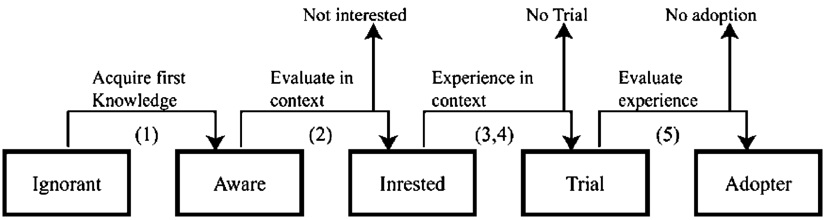

Fig. 1. A model of the change process. The numbers indicate the mapping to Rogers' model.

In this phase, learners actually experiment with sharing their knowledge. They evaluate, on the basis of practical experience, the pros and cons of the new behaviour in their specific environments (e.g. with a given group of colleagues, with a group of friends, with an interest group), using particular tools (e.g. updating a database, participating in workshops, taking part in online discussions), and with respect to specific situations.

Finally, the trial may result in confirming the adoption of the behaviour, in which case the individuals enter into the adoption state, otherwise they may opt for the rejection of the behaviour (phase (5) of Rogers' model).

2.3. Facilitating the change process: intervention strategies

Given the model of Fig. 1, the change process is defined as a set of change operations upon user change states, aiming at achieving the adoption state for a given behaviour. The change operations are: *acquire first knowledge*, *evaluate in context*, *experience in context*, *and evaluate experience*. The change states are: Ignorant, Aware, Interested, Trial, and Adoption.

Each one of the change operations can be described in terms of the learning processes (possibly involving physical actions) enabling that operation. As a consequence, the whole change process can be seen as (a more complex) learning process.

In order to facilitate the change process, it is possible to define and activate a set of *intervention strategies* that facilitate such learning. Intervention strategies are actions, external to the individual undergoing the change process, aimed at facilitating a change operation; they correspond to teachers' interventions or pedagogical processes.

For instance, for individuals in the *ignorant* state with respect to a behaviour, it is more appropriate to supply information about the new behaviour rather than presenting them with proposals for the evaluation of the behaviour. Also, intervention strategies can be tuned to individuals' characteristics, such as one's attitude towards change, social network, and learning preferences.

In Sections 2.3.1 and 2.3.2, we analyse how intervention strategies can be adapted to the individual's change state (Section 2.3.1) and to other personal characteristics (Section 2.3.2). This will result in the operational definition of an intervention model (Section 3.4).

2.3.1. Intervention strategies and change state

Various learning theories have emphasised some aspects of learning that can be related to the change operations described earlier. Below we illustrate the change process as a learning process with a set of examples that emphasise how the process can be facilitated.

The operation acquire first knowledge corresponds to the user acquiring information about the new behaviour. This operation can be facilitated using classic teaching methodologies that aim at 'transferring' some facts from some source to a learner (e.g. a teacher or a book).

The operation evaluate in context allows the discovery of the relationships between the new behaviour and the user's current (individual or social) goals and activities. Situated learning theories (Lave, 1991; Lave and Wenger, 1990) emphasise precisely this context-dependent aspect of learning, stressing the importance of social interaction. Context-based evaluation is often based on the observation of other members of the community. For example, how do members implement the new behaviour? How well does the behaviour work for them? This aspect is also emphasised in social learning theories (Bandura (1971), for a cognitive analysis, see Conte and Paolucci (2001)).

The operation experience in context reflects aspects of learning analysed in theories, such as Rogers' experiential learning (Rogers, 1969). The focus is on addressing the needs and desires of the learner to achieve personal involvement, and on making learning self-initiated and evaluated by the learner. Bruner's constructivist theory (Bruner, 1996) also stresses experience but in the sense that learning is an active process in which learners construct new ideas or concepts based upon their current knowledge and the teacher helps students to discover principles by themselves.

The learning theories briefly introduced above offer instructors some guiding principles for the facilitation of the learning process emphasising various factors (cognitive, social, cultural) of the process. We have used these principles to define a set of intervention strategies facilitating the transition through the various stages of the change.

Table 1 classifies a set of intervention strategies along two dimensions: the change operation (listed in the first column) and the factors influencing learning (listed in the first row).

2.3.2. Intervention strategies and learner's characteristics

Table 1 allows matching intervention strategies to change operations and therefore to user change states. However, the choice of the intervention strategy should also take into account the learner's characteristics. In particular, the learner's attitude towards innovation and the learner's social position are factors just as important as the learner's change state in selecting and tuning the strategies to apply.

Not every person has the same attitude towards change and innovation, nor the same role in the diffusion process. Rogers (1995, p. 262) identifies five categories of adopters, in increasing order of resistance to change. (1) *Innovators* (about 2.5%) are the more venturesome. Although they are frequently at the origin of the introduction of the innovation in the system, they are not normally social leaders. (2) *Early adopters* (about 13.5%) are well integrated into the social group, often have a leader role, and generally play an important role in initiating the diffusion of the innovation in the group. (3) *Early majority* (about 34%) have a relatively positive attitude towards innovation, but need more time to adopt the innovation. (4) *Late majority* (about 34%) are more cautious about

Table 1
Examples of intervention strategies

	Cognitive	Social	Cultural
Acquire first knowledge	Introduce the behaviour and how it is implemented	Describe how the behaviour has been useful to others	Present the relevance of the behaviour in general
Evaluate in context	Describe how the behaviour could be useful for the user's current activity	Describe how community members, in the user's social network, have implemented the behaviour	Describe the relevance of the behaviour to the community
Experience in context	Propose actions to implement the behaviour within the user's current activity	Propose actions, implementing the behaviour, that would benefit community members in the user's social network	Propose actions that would benefit the community and explain how they respond to internal policies
Evaluate experience	Summarise the results obtained by implementing the behaviour	Evaluate the advantages in terms of social network	Evaluate advantages for the community

innovation and typically adopt once the majority of the people have adopted. (5) *Laggards* (about 16%) have the most cautious attitude towards change.

Following Rogers' classification, we define the *user's attitude towards innovation* as a measure of how quickly a user goes through the change process with respect to other users in the community. Note that this definition means that the same person may appear as an innovator within one community whilst for instance, he may be ranked as late majority in a different community. The same is true for the user's attitude towards innovation within different change domains: an individual may be an innovator within certain domains and a laggard within others. This definition seems to correspond to the common sense understanding of people's *attitude towards innovation*.

The user's attitude towards innovation can be exploited to define the best intervention strategy to apply for each individual. Late majority people, for example, attach a lot of importance to others' opinions, therefore strategies, such as the *acquire first knowledge/social* and *evaluate in context/social* described in Table 1, may be very effective with this category of users. On the other hand, innovators are less influenced by the social group and value hands-on experience to forge an opinion, therefore strategies, such as the *experience in context/cognitive* may be more appropriate for them.

The user's *social network* represents the relationships of one community member with others and with individuals and communities external to the one considered. Included in the social network are: (1) the personal network composed of friends and acquaintances; (2) the affiliation to sub-communities, and (3) the organisational network, such as the boss, the colleagues, the work acquaintances. The importance of social networks in innovation diffusion, business processes and economics is very well recognised, some of the latest studies include (Cohen and Stathis, 2001; Deroïan, 2002; Janssen and Jager, 2001).

The change process is inherently a social process and people's contribution to the social diffusion of innovation depends on their social position. The social network supplies information that can be exploited to understand the role of people in the process of diffusion and adoption. For example, individuals with a leading role in the organisation and a wide personal network are usually more likely to influence the adoption process. Individuals are regarded as *members of a group* rather than as *stand-alone* learners. Intervention strategies can be made more effective exploiting knowledge of the social network. For example, the evaluate in context/social strategy in Table 1 is made more effective by supplying examples of the implementation of the behaviour by friends of the learner or by leaders. As another example, the *experience in context/social* strategy may generate proposals for group work, the selection of participants in the group may be based on the individuals' attitude towards innovation, their areas of expertise and their social network.

More details on how the learner's characteristics are represented, assessed and updated within the K-InCA system are given in Section 3.2.

3. K-InCA: a multi-agent system supporting the adoption of knowledge sharing practices

The K-InCA system implements and makes operational the concepts introduced in Section 2. Fig. 2 illustrates the fundamental elements of the system. They are: (1) a change

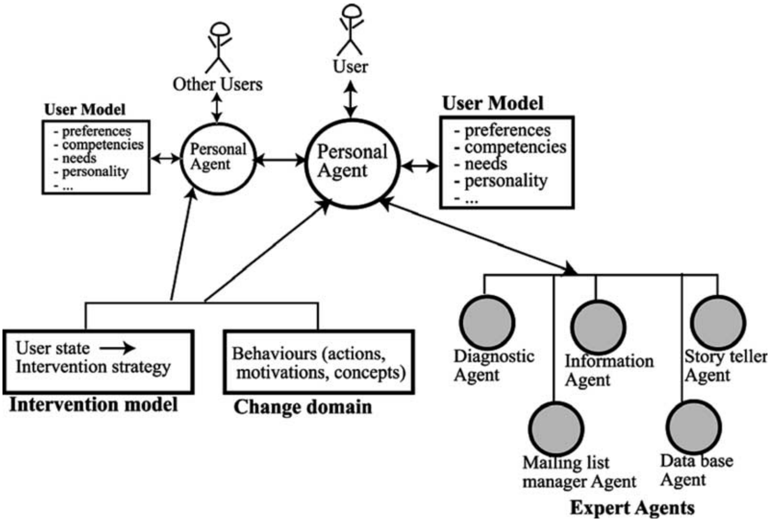

Fig. 2. Overall K-InCA architecture.

domain, representing the content to be delivered, the behaviours to be adopted; (2) a set of user models, describing the users' characteristics, needs, interests, learning state; (3) a set of agents, that observe the users actions and provide personalised guidance by applying the most appropriate teaching strategies; (4) an intervention model, defining the best teaching strategies.

This section first describes the four components of the system, then presents K-InCA's user interface giving some examples of how the user may interact with the system, finally it briefly presents the system implementation.

3.1. Change domain: representing knowledge sharing behaviours

The change domain represents the collection of behaviours envisioned for the organisation. Our definition of 'behaviour' follows the classic psychological definition of 'a goal directed activity' (Plotkin, 1994). A behaviour is something the user does for some reason/motivation/goal. By performing certain actions users *implement*—i.e. temporarily adopt—certain behaviours.

Within the system, a behaviour is defined and represented by:

- a set of *actions*: describing what one would do when implementing that behaviour
- a set of *motivations*: describing how one would be motivated when implementing the behaviour
- a set *concepts*: which illustrate the behaviour to the user.

These three elements are described in Sections 3.1.1–3.1.3.

Behaviours are structured in a hierarchy to reflect the fact that one complex behaviour may be decomposed in simpler behaviours. Fig. 3 sketches part of the *knowledge sharing*

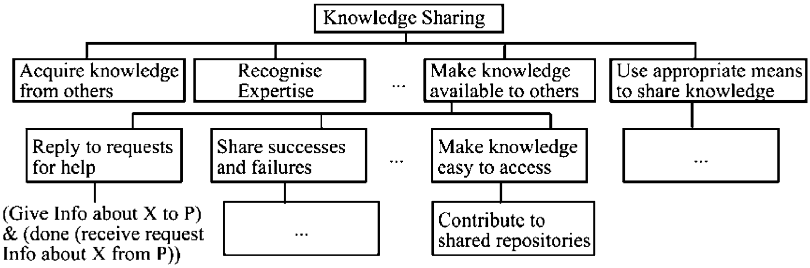

Fig. 3. Example of hierarchy of behaviours in the change domain.

hierarchy. For example, one may adopt a knowledge sharing behaviour by adopting the *make knowledge available to others* behaviour or by adopting the *acquire knowledge from others* behaviour.

Internal nodes of the hierarchy are called *complex behaviours*. Users implement complex behaviours by either performing their associated actions or by implementing one or more of their sub-behaviours. For example, the *reply to request for help* behaviour implements the make knowledge available to others behaviour. Leaf nodes, i.e. behaviours that are not further decomposed in sub-behaviours, are called *simple behaviours*. Users implement simple behaviours by performing their associated actions.

The whole change process is rooted in one behaviour (e.g. knowledge sharing behaviour, *knowledge management* behaviour, *entrepreneurial* behaviour, *social responsibility* behaviour). In the K-InCA system, the root behaviour is the knowledge sharing behaviour.

3.1.1. Action representation

User's actions are described by a structure indicating: the action's context, the action's type, the action's content, the action's subject and the action's third party. Actions may be only partly specified (partially instantiated).

The *action context* describes the situation in which the action takes place, e.g. a meeting, an informal talk, a seminar, a conference, an online situation.
The *action's type* describes the user's act. For example, the user may *ask for something*, *tell something*, *receive something*, *write something*. Currently only communication actions are implemented (ask, tell, receive).
The *action's content* specifies *what* has been asked for, told, written, etc. It supplies the value for the *something* of the action's type. Examples are: information, suggestion, and acknowledgement.
The *action's subject* indicates the specific subject of the action. It can be any sequence of wording, such as 'project Alpha' as in *asking information about project Alpha*.
The *action's third party* specifies who else is involved in the action. For example, one may ask information about the project Alpha to *the Alpha mailing-list*.

An example of a completely instantiated action is *In an informal talk I have asked* *advice about project Alpha to Mr Smith*. An example of a partially instantiated action may be *I have asked advice to Mr Smith*.[2]

We assume that actions associated to behaviours are observable (by the system). Such actions are generally partially instantiated. For instance, the bottom left box of Fig. 3 states that if the user performs the action to *give information about some subject X to person P* after *receiving a request for information about X from person P* then the user has implemented the *reply to request for help behaviour*.

The actions and sub-behaviours associated to a behaviour (its children in the tree of Fig. 3) represent non-exclusive, alternative ways of implementing the behaviour.

3.1.2. Motivations for behaviours

The motivations associated to behaviours define why a user should adopt the given behaviour. A motivation is composed of four elements indicating (1) who will benefit from the user's adoption of the behaviour, (2) the type of advantage (e.g. economic, professional, image, meet-organisation-requirement, meet-organisation-aspiration), (3) the applicability conditions for the motivation and, (4) some text used to present the motivation to the user. Super-behaviours are always considered as possible motivations for sub-behaviours (with the *organisation* as a beneficiary, *meet-organisation-aspiration* as the type of advantage and the applicability condition always satisfied). For instance, one may *reply to a request for help* in order to make knowledge available to others (super-behaviour), or in order to receive a monetary compensation attached to that type of request.

3.1.3. Concepts associated to behaviours

Five concepts are associated to each behaviour. They respectively describe: (1) what the behaviour is, (2) why and for whom the behaviour is relevant, (3) how to practise the behaviour, (4) who can provide further information on the behaviour and (5) where further information on the behaviour may be found. Concepts are basically collections of text descriptions and pointers to documents about the behaviour.

3.2. User model

As discussed in Section 2.3, several types of information about the user are relevant to the management of the change process and in particular to the selection and implementation of the intervention strategies. This section explains how this information is organised, collected and updated within the user model.

The diagnostic agent is responsible for updating the user model. Whenever a new user's action is reported, the diagnostic agent updates the model accordingly. This choice has the advantage to clearly separate diagnostic functions, aimed at inferring user's characteristics, from other types of functions. As the set of actions represented by the system will extend, several diagnostic agents may be introduced, each specialising on a different category of action; for instance we may have diagnosis of communication actions, diagnosis of document manipulation actions, diagnosis of search actions.

[2] All the people's and project's names mentioned in the examples of this paper are fictitious.

3.2.1. User's change state

The user's change state is a description of the level of adoption of each one of the behaviours included in the change domain. With respect to each behaviour, the user can be in one of the following change states: *ignorant*, *aware*, *interested*, *trial*, or *adoption*.

We assume that community members are initially ignorant with respect to all behaviours and their state change is marked by specific events. The transition from the ignorant state to the aware state follows an event in which the user has been informed about the behaviour. The transition from aware to interested is marked by the user's requesting information about the behaviour or spending time finding out about the behaviour in general. The transition from interested to trial is achieved when the user actually performs some actions implementing the behaviour. Finally, the transition from trial to adoption corresponds to the repeated implementation of the behaviour. The diagnostic agent is capable of recognising these events and updating the change state accordingly.

3.2.2. User's attitude toward innovation

We have seen (Section 2.3.2) that the user's attitude towards innovation is a measure of how quickly a user goes through the change process with respect to other users in the community. The diagnostic agent maintains this measure. A user's attitude indicator is associated with each behaviour. Also, a user's attitude indicator is associated with the whole domain. When the state of the user changes with respect to a given behaviour (e.g. the user achieves the trial state for the *reply to request for help*), his/her inclination towards the adoption of that behaviour is measured as the inverse of the number of members of the community who have already achieved that change state for that behaviour. The user's attitude towards innovation is the average of his/her inclination towards the adoption of all the behaviours in the change domain. These numbers are then associated to a category (e.g. laggard, innovator) following Roger's statistics presented in Section 2.3.2.

3.2.3. User's social network

The social network is represented by listing the people in the network and, for each entry, indicating the type/s of relationship/s (e.g. friend, colleague, belonging to the same sub-community) and the 'level' of the relationship. By level of relationship we mean how close the relationship is (e.g. a friend versus a good friend, a colleague versus a colleague with whom one works every day).

Users can declare (and update) entries describing their social network and the diagnostic agent infers relationships between users by both observing the user's action and by inferring properties using simple inference rules or stereotype hierarchies (Rich, 1979, 1989). For example, when the user declares that he has *asked information to Mr Smith about project Alpha*, the diagnostic agent adds Mr Smith to the user's acquaintances or it raises the level of acquaintance of the user with Mr Smith.

3.2.4. User's interests and expertise

The user's *interests and expertise* are represented in the user's model through two lists of keywords each with an associated value indicating the interest/expertise level. These components can be declared and updated by both the user and the diagnostic agent on the basis of the users' actions. For instance, when the user declares that he has asked information to Mr Smith about project Alpha, the diagnostic agent adds the project Alpha to the user's interests. If this project is already listed amongst the user's interests, the diagnostic agent rises the level indicator.

The weights/levels assigned to user's interests, expertise and social network allow representation of facts, such as the user being 'very interested' in a given subject and only 'marginally interested' in another one; or the user 'just knowing' Mr Smith whilst being 'well acquainted' with Mr Black. Using the same weight assignment procedure for all users allows, for instance, comparison of the expertise level of several users on the same subject (for a more detailed discussion of this aspect of the system, see Roda et al. (2001)).

3.2.5. User's activity

The user's *activity* is a log of the actions performed by the user. The order in which the actions appear in the log file is significant and represents their temporal relation. This field of the user model allows the system to reason about user's behaviours that are related to *sequence* of actions rather than individual actions.

3.2.6. User's knowledge management agenda

The user's *knowledge management agenda* contains two ordered lists indicating those behaviours that are currently most relevant to the user. We consider that a behaviour may be relevant either because it has been recently implemented by the user, or because there has been an occasion in which it could have been implemented but it was not. The two lists correspond to these two types of behaviours. The procedure used by the diagnostic agent to update the knowledge management agenda is described in Section 3.3.1.

3.3. The agents

The agents' goal is to bring the user from some initial state to a state of adoption for the behaviours included in the change domain. This goal is achieved with the execution cycle of Table 2.

Each user interacts with the system through a *personal agent*. This personal agent co-ordinates the activity of a set of agents each having specific roles and expertise.

After observing a user's action (Step 1), the personal agent activates the *diagnostic agent* who updates the user model (Step 2). Based on the updated model, the personal agent selects a new current *learning objective*, i.e. a set of behaviours in the change domain whose level of adoption should be increased. Then, it solicits proposals from the expert agents to achieve the learning objective. The *expert agents* may propose to implement one or more intervention strategies. The personal agent selects the best proposal by applying the preference rules given in the intervention model (Step 3). Finally, in Step 4, the personal agent requests the selected expert agent to implement its intervention.

The personal agent guides the process and has control over the interaction with the user. In this section, we analyse the agents' behaviour in terms of their activity during the various steps of the execution cycle.

Table 2
Flow of execution within the agent system, given an action of the user

Step 1	Observe user's action
Step 2	Update user's state
Step 3	Select a learning objective and an intervention mode
Step 4	Execute the intervention

3.3.1. The diagnostic agent: updating the user model

Upon observing a user's action (Step 1 of the execution cycle) the personal agent activates the diagnostic agent in order to update the user model (Step 2 of the execution cycle).

Many of the diagnostic agent's activities have already been described in Section 3.2. This section explains how the diagnostic agent updates the knowledge management agenda.

The diagnostic agent's objective is to identify the behaviours that are most relevant to the user's current activity. Given (1) the user's current action, (2) the user's previous activity, and (3) the change domain, the diagnostic agent generates a set of hypotheses about the behaviours that the user may have implemented by performing the current action. These hypotheses are generated by matching the user's actions with those associated with behaviours in the change domain—see, for example, the action associated to the *reply to requests for help* behaviour of Fig. 3.

Assume, for instance, that the user performs the action: *in an informal talk I have given information about project Alpha to Mr Smith*. Also assume that the user had previously declared the action *in an email I have received a request for information about project alpha from Mr Smith*. The diagnostic agent would recognise that the user's actions match the actions associated to the *reply to request for help* behaviour. Consequently, the diagnostic agent would generate the hypothesis that the user has implemented the *reply to request for help* behaviour.

Some actions associated to the behaviour may only partially match the user's action. These partial matches allow the diagnostic agent to identify those behaviours that have not been implemented but could have been implemented (would have been desirable in the given context).

In the example above, following the user's action *in an informal talk I have given information about project Alpha to Mr Smith*, the behaviour *distribute information to all interested parties* may be identified as a desirable behaviour that has not been implemented.

Consequently, the diagnostic agent updates the knowledge management agenda by raising the relevance of the identified behaviours in the respective lists: the implemented behaviours and the desired behaviours lists. During the process of behaviour identification, the diagnostic agent may interrogate the user in order to gather the information necessary to confirm its hypotheses.

The diagnostic agent also updates the change state, attitude toward innovation, social network, interests and expertise, and user activity slots of the user model, as explained in Section 3.2.

3.3.2. The expert agents: select a learning objective and an intervention mode

Once the user's state has been updated, Step 3 of the execution cycle (select a learning objective and an intervention mode) is executed through the following sub-steps:

Step 3.1 the personal agent selects the current learning objective
Step 3.2 the personal agent solicits proposals from the expert agents by broadcasting a request for proposals for the selected learning objective
Step 3.3 the expert agents respond to the personal agent's request by proposing to implement one or more intervention strategies to achieve some part of the learning objective
Step 3.4 the personal agent selects the most appropriate proposal.

In Step 3.1, the personal agent selects a learning objective, i.e. the set of behaviours whose adoption level should be raised. Currently, the personal agent composes the learning objective by selecting the four most relevant behaviours (those at the top of the knowledge management agenda's lists of *implemented* and *desired* behaviours) that have not yet been adopted by the user. In the future, the selection of the learning objective may be guided by more complex rules. Such rules may not only take into consideration the current relevance of a behaviour for the user, but also other factors, such as the importance of the behaviour with respect to the user's role within the organisation, or the balance amongst the behaviours that are being adopted and those that are lagging behind.

In Step 3.2, the personal agent starts an *intervention selection protocol* that may be completed during Step 4 if an intervention is implemented by one of the expert agents. The complete intervention selection protocol is depicted in Fig. 4.

The personal agent broadcasts a request for proposals by sending a message:

```
generateProposal(PA, LearningObjective)
```

where `PA` is the Personal Agent itself and the `LearningObjective` is the selected learning objective, i.e. the list of behaviours whose adoption level should be increased by the experts' interventions.

Upon receiving the personal agent's request, the expert agents verify if they can generate proposals capable of satisfying the learning objective (Step 3.3). If they can, they put forward their proposals by sending a message:

```
Propose (AgentName, ProposalID, Behaviour,
  InterventionStrategy, Personalisation,
  TextDescription)
```

where:

- `AgentName` is the expert agent's name (this corresponds to a unique identifier in the system)
- `ProposalID` is the unique (for the expert agent) identifier of the proposal. This is necessary in case the expert agent submits more than one proposal.

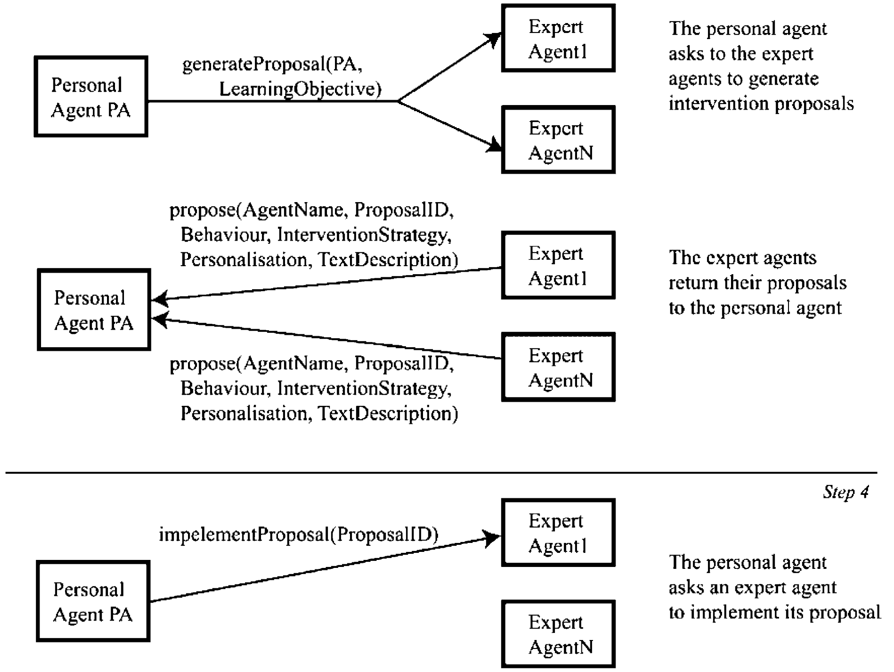

Fig. 4. The intervention selection Protocol: N expert agents propose intervention strategies to the personal agent who selects the most appropriate one for presentation to the user.

- `Behaviour` is he identifier of the behaviour (which must belong to the learning objective set) that would benefit from the proposed intervention
- `InterventionStrategy` is the name of the intervention strategy that will be implemented
- `Personalisation` is a (possibly empty) set of pairs (user_model_slot = user_model_slot_value) specifying how the proposal is personalised for the user. For example, this field may tell the personal agent that the proposal acts upon one specific user's interest.
- `TextDescription` is a text description of the proposal allowing the user, if he wishes so, to select the best proposal in place of the personal agent.

Fig. 5 shows an instantiated example of the intervention selection protocol where a personal agent requests proposals to improve the adoption level of the behaviours *reply to requests for help* and *make knowledge easy to access*. The *Story-Teller* agent proposes to implement the *acquire_first_knowledge_cognitive* strategy in order to improve the adoption of the make knowledge easy to access behaviour by giving some information on the behaviour to the user. The *Emailer* agent proposes to implement the *experience_in_context_social* strategy in order to improve the adoption of the *reply to requests for help* behaviour, this strategy will be personalised by involving the user's friend *Paul* and by being related to the user's interest in *InCA*. The personal agent selects the Emailer agent's proposal and asks to this expert to implement its proposal.

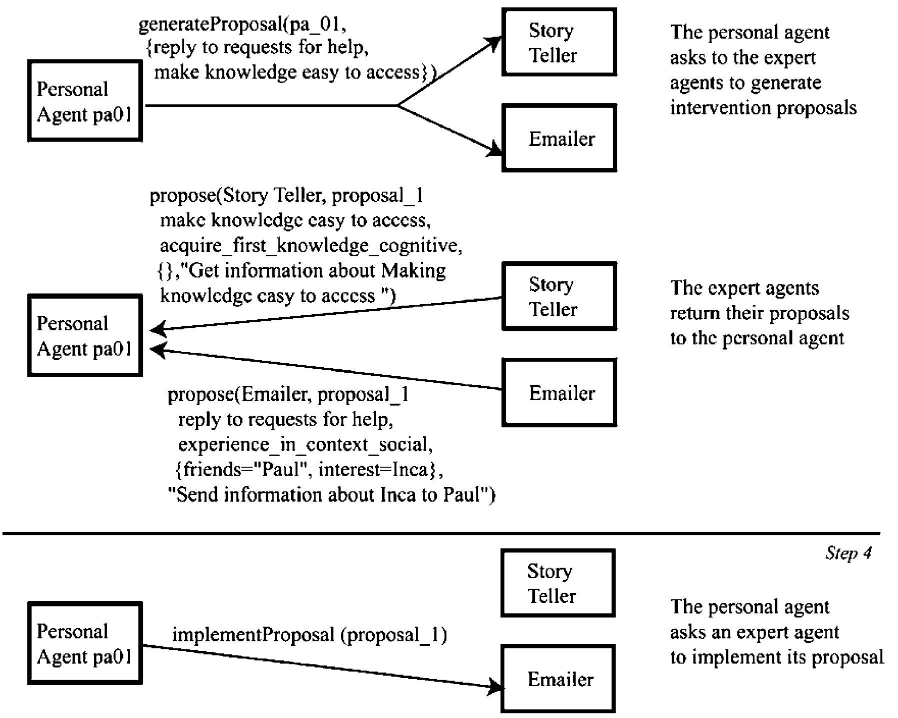

Fig. 5. An example of instantiation of the intervention selection protocol.

Note that several expert agents may implement the same intervention strategy. For example, the *experience in context/cognitive* strategy for the *contribute to shared repositories* behaviour may be implemented by one expert agent who proposes the use of a database and by another expert agent who proposes the use of a shared annotation system.

Finally (Step 3.4), the personal agent selects the best proposal by evaluating each one of them using the rules specified in the intervention model (these rules are detailed in Section 3.4). It then activates the corresponding expert for presentation to the user.

K-InCA agents communicate, with the user and amongst each other, through a set of protocols similar to the intervention selection protocol. For example, the personal agent may co-ordinate with other personal agents whenever this is necessary for the implementation of certain intervention strategy (e.g. to find ‘producers’ of information needed by the user, see Roda et al. (2001)). The discussion of the set of protocols controlling agents’ communication is behind the scope of this paper and it is not further discussed here.

The broadcasting mechanism of the intervention selection protocol allows one to dynamically introduce and remove expert agents from the system. When a new expert agent is added to the system, it is registered with the personal agent who, from then on, will include it in the broadcasting mechanism.

Note that whilst some expert agents are purposely built for the K-InCA system (e.g. the story-teller agent), others act as *wrappers* for existing systems. For example, an expert agent may wrap a mailing list manager, say the *project-Alpha-mailing-list-manager* expert. When it receives a request for proposals, it may propose that the information collected by the user be distributed through its list. In this case, it would submit the following proposal for contributing toward the adoption of the *distribute information to all interested parties* behaviour:

```
propose(project-Alpha-mailing-list-manager, ID001,
distribute-information-to-all-interested-parties,
experience_in_context_cultural,
{interest = "project Alpha"},
"You could send a message about project Alpha to all
members of the project Alpha mailing list")
```

3.4. Intervention model

Conceptually, the intervention model formalises and makes operational the analysis of the intervention strategies presented in Section 3.3.

The intervention model can be seen as a set of rules guiding the personal agent in the assignment of a *preference level* to each of the expert agents' proposals. These rules have the general format:

```
If {Conditions_on_user_state AND
Conditions_on_expert_proposal}
then prefer (Strategy1 OR…OR StrategyN)
```

Currently, the function evaluating the preference level is very simple:

- All experts' proposals start with a preference level = 0
- For each proposal, the preference level is equal to the number of rules that *apply* to the proposal

A rule *applies* to a proposal if:

- the `Conditions_on_user_state` hold;
- the `Conditions_on_expert_proposal` hold;

the strategy proposed belongs to the preferred set `prefer (Strategy1 OR…OR StrategyN)`.

3.4.1. General preference rules

In their simplest format, the intervention model's rules allow the personal agents to verify if the `InterventionStrategy` that the expert proposes in order to improve the adoption level of the `Behaviour`, is adapted to the user's change state for that behaviour. These rules have the form:

```
If user_change_state(Behaviour) = Change_state
then prefer(Strategy1 OR…OR StrategyN)
```

These simple rules, that we call *general preference rules*, reflect the discussion of Section 2.3.1 and identify the intervention strategies that are best suited to support the change operation required by the user's change state.

Given a *general preference rule*, it applies to a proposal:

```
propose (AgentName, ProposalID, ImprovedBehaviour,
  InterventionStrategy, Personalisation,
  TextDescription)
```

if:

- `user_change_state(ImprovedBehaviour) = Change_state`
- and `InterventionStrategy` belongs to `{Strategy1,…,StrategyN}`

The following is an instances of general preference rules and it states that "If the user is ignorant with respect to behaviour `Behaviour` then one of the acquire first knowledge strategies should be applied":

```
if (user_change_state(Behaviour) = "ignorant")
then prefer(acquire_first_knowledge_cognitive OR
  acquire_first_knowledge_social OR
  acquire_first_knowledge_cultural)
```
(GP1)

When the personal agent receives a proposal, say:

```
propose(Story Teller, proposal_1, make knowledge easy to
access,
  acquire_first_knowledge_cognitive, { },
  "Get information about making knowledge easy to
access")
```

It can verify that the above rule applies to the proposal by verifying that:

- `user_change_state(make knowledge easy to access) = "ignorant"`

- and `acquire_first_knowledge_cognitive` belongs to {`acquire_first_knowledge_cognitive`, `acquire_first_knowledge_social`, `acquire_first_knowledge_cultural`}

3.4.2. Other preference rules

In Section 2.3.2, we have seen that certain users' characteristics can help identifying the learning factors (cognitive, social, cultural) that are more likely to be effective in the adoption process. Those considerations are used to define more specific rules, called *state preference rules*, that in general identify a smaller set of preferred strategies than the general preference rules.

The general format of state preference rules is:

```
if user_change_state(Behaviour) = Change_state AND
  Other_conditions_on_user_state
then prefer(Strategy1 OR…OR StrategyN)
```

The following is an instance of a *state preference rule* that makes (GP1) more specific and reaches only one conclusion:

```
if (user_change_state(Behaviour) = "ignorant") AND
  (user_innovation_attitude = "late majority")
then prefer(acquire_first_knowledge_social)
```

In order to verify if this type of rules apply to a proposal, the personal agent verifies the user change state with respect to the behaviour, as well as other users characteristics (in the example, it verifies the user's attitude toward innovation).

In general, the intervention model may include other rules allowing the personal agent to make the best possible selection amongst the expert proposals. For example, *personalisation preference rules* state that certain intervention strategies should be preferred when adapted to some specific user's characteristic. These rules reflect the discussion in Section 2.3.2, where we have seen that intervention strategies may be more effective when personalised for certain characteristics of the user (e.g. his/her relations with certain people, his/her specific interests).

Depending on how many, and how specific, are the preference rules in the intervention model, the personal agent may come to select a single proposal or it may reach the conclusion that several proposals are equivalently well suited for the achievement of the learning objective. In the former case, the expert agent that has submitted the best proposal is activated with an `implementProposal` message. In any case, the proposals with the highest preference are shown to the user (by displaying their `TextDescription`) who may make his own selection.

In the system set-up phase the 'super user' initialises the change domain and the intervention model using two editors designed for this purpose (these editors may also be used at run time). Before initialisation, the change domain is empty and the intervention model contains a set of default rules: the four general preference rules.

3.5. K-InCA user interface

The current version of the system concentrates on the management of the change process without addressing the recognition of the user's actions. We assume, for the time being, that users will declare their actions and motivations. Users initiate K-InCA *coaching sessions* after a *practice period*. During the practice period, users perform their normal activity. During the coaching session, users report their activities and interact with the K-InCA agents who supply advice and stimulus towards the adoption of knowledge sharing behaviours.

In this section, we give an example of the use of the system that may clarify the concepts presented. First we show the interface provided by K-InCA to give visibility and access to the user model and the change domain, then we concentrate on a sample execution cycle showing how agents respond to users' actions.

3.5.1. User model visibility

The user interacts with the system through a main *input window* displayed in Fig. 6. This window can be customised but it generally has three areas whose dimensions may vary: a view of the user state (on the left), an area for direct interaction with the personal agent (top right), and an area for normal activity (bottom right, blank in the image).

The information contained in the user model is displayed in the *my profile* section. The *Adoption level* pane displays the user's change state.[3] The *innovation attitude* pane displays the user's attitude towards innovation with respect to each behaviour in the change domain. The *social* pane displays the user's social network. The *Interests* and *Experience* panes display the user's interests and expertise keywords along with their levels. The knowledge management agenda pane displays the behaviours that are currently most relevant for the user. The action history is not displayed in the user model summary of the input window, it is however available in the *user model* window that the user can reach by clicking on the *my profile* link. The user model window also allows the user to modify all the fields.

3.5.2. Change domain visibility

The change domain represents the behaviours that the user should adopt. In the case of K-InCA, the change domain contains knowledge related to knowledge sharing behaviours. Users can access *behaviour description windows* by clicking the link on any behaviour's name in the input window. Users may also reach the *behaviour description window* by following expert agents' suggestions. From the behaviour description window, users may explore the behaviours hierarchy by following links in the *sub-behaviours* pane. Fig. 7 is a screenshot of the behaviour description window presenting the knowledge sharing behaviours.

The actions and sub-behaviours associated to a behaviour (its children in the tree of Fig.

[3] Note that the ignorant state is represented by a u (for unaware).

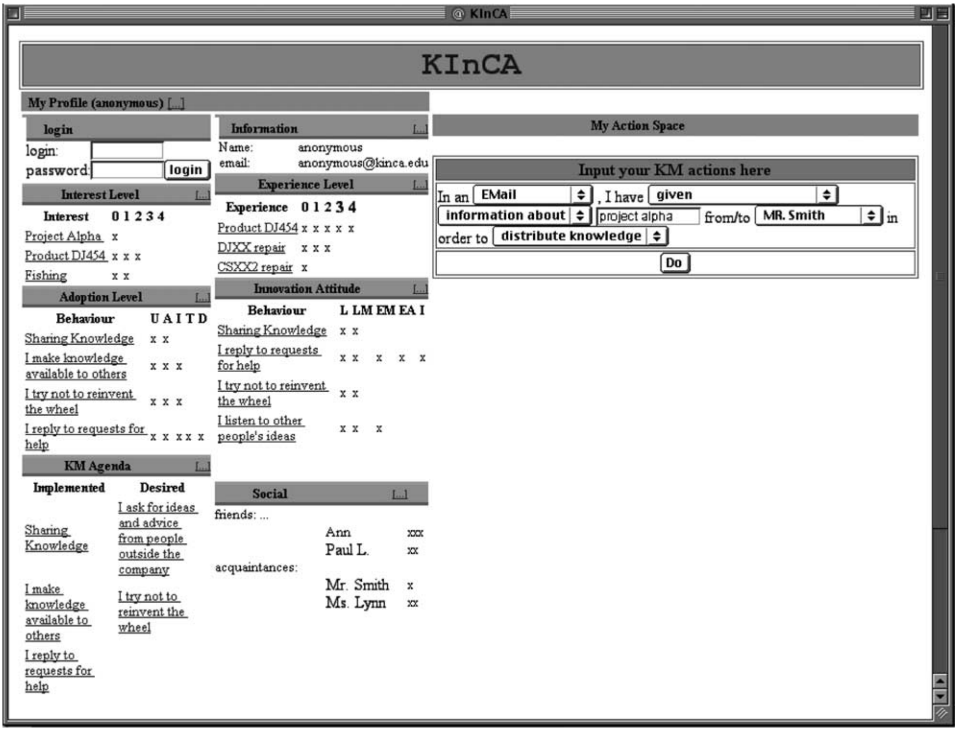

Fig. 6. K-InCA input window.

3) are presented to the user in the *actions' filter* and sub-behaviours panes of the behaviour window (if the user performs one of these actions the diagnostic agent will recognise that the associated behaviour is currently relevant for the user). The user has an overview of the concepts associated to a behaviour in the 'about' pane of the behaviour description window, in particular, the text presenting the motivation for the behaviour appears in the 'why and for whom' concept. Following the links from this window, the user can visit any of the concepts. Fig. 8 shows the 'what is' concept of the knowledge sharing behaviour. Concepts are linked to related documents through *concept illustrators*. These are shown in the *associated information* pane.

3.5.3. Sample execution cycle

This section gives an example of how the system may respond to user's actions. We will follow the 4-step execution cycle of Table 2.

Step 1. *Observe user's action*. User declares his action (see *input your KM action here* pane, Fig. 6).

User: *In an email I have given information about project Alpha to Mr Smith in order to distribute knowledge*

Note that the user can specify the super-behaviour that motivates his/her action in the

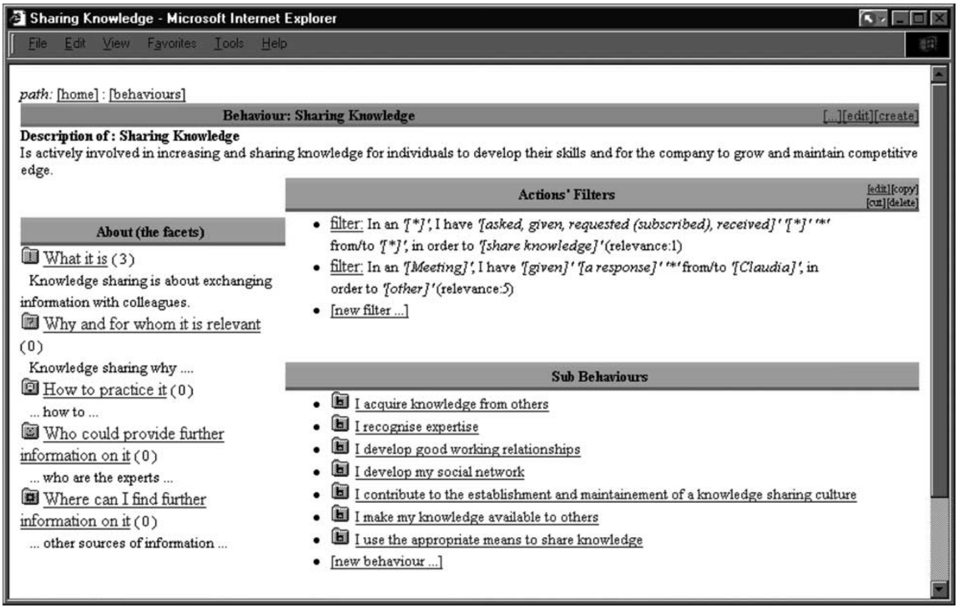

Fig. 7. Behaviour description window: knowledge sharing behaviour.

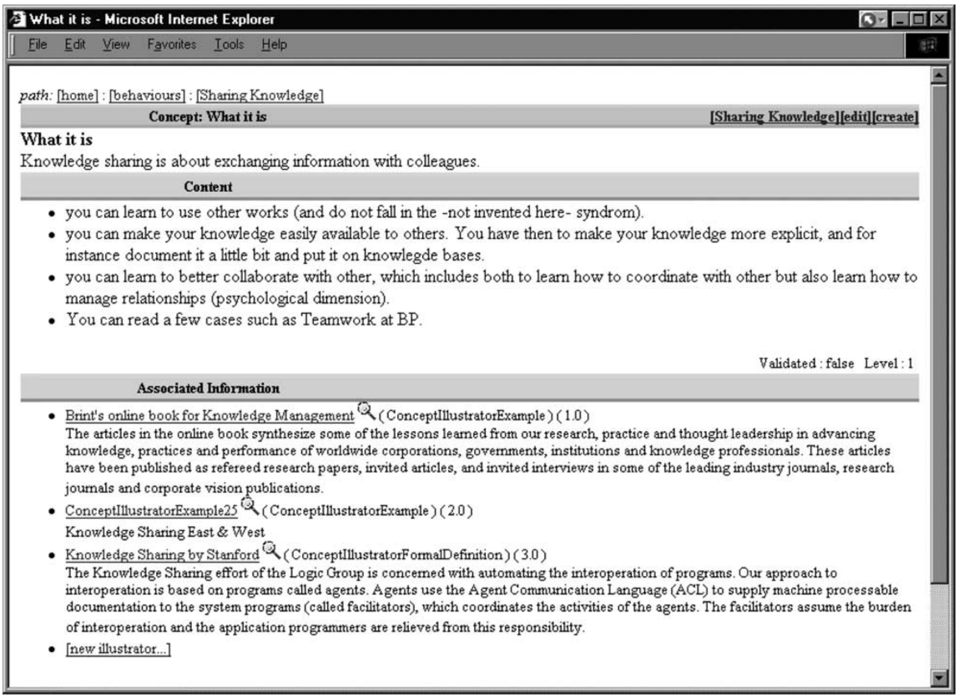

Fig. 8. Screen presenting the concept *What it is* for the knowledge sharing behaviour.

'in order to' field, however, motivations are currently only partially supported by the system.

Step 2. *Update user's state*. The diagnostic agent recognises that:

- The user has implemented the *I make knowledge available to others* behaviour and the *I reply to requests for help* behaviour.
- The behaviours *I ask for ideas and advice from people outside the company* and *I try not to reinvent the wheel* would have been desirable behaviours.
- *Mr Smith* is an acquaintance of the user.
- The user is interested in *project alpha.*

The relevance of the four identified behaviours—the implemented behaviours and the desired behaviours—is raised (see *KM agenda* pane Fig. 6). The change state for the two-implemented behaviours may be increased (see *adoption level* pane Fig. 6). Mr Smith is added to the user's acquaintances (see social pane Fig. 6). And project Alpha is added to the user's interests (see interests pane Fig. 6).

Step 3. *Select a learning objective and an intervention mode*. The personal agent selects the learning objective as composed of the behaviours {*Sharing knowledge*, *I make knowledge available to others*, *I ask for ideas and advise from people outside the company*, *I try not to reinvent the wheel*}. It then broadcasts a request for intervention to the expert agents. The expert agents propose several possible interventions:

- A database expert proposes to address the *I make knowledge available to others* behaviour, with an *experience in context/social* strategy
- A story-teller agent proposes to address the *I ask for ideas and advice from people outside the company* behaviour with an *acquire first knowledge/cognitive* strategy

Based on the information contained in the intervention model, the personal agent selects the first proposal because it matches well the user's interested state for the *I make knowledge available to others* behaviour (i.e. the experience in context operation needs to be supported). The proposed strategy also matches the user's late majority attitude to knowledge sharing behaviours.

Step 4. *Execute the intervention*. The personal agent requests the database expert agent to implement its proposal. The database expert suggests to the user that he/she could store the information sent to Mr Smith in a database and it offers to help him/her to do so.

The user may accept the proposal or continue with other activities, e.g. browsing the behaviours, inputting more actions.

3.6. Implementation

The ideas presented in this paper have been selectively implemented in a prototype. The change domain structure, including its editor, has been implemented and it has been initialised with the knowledge sharing change project envisioned by our sponsor, Xerox. This has allowed us to verify that the model we have used for the representation of behaviours, actions, motivations and concepts is powerful enough to represent a real change project. The intervention model, the personal agent and the diagnostic agent have been implemented, with the exception of the editor for the intervention model that has not been implemented yet, and the actions' filters that have been implemented on a simplified syntax. The user interface, allowing the users to specify their actions, as well as browsing the user model and the change domain, has been implemented along with a set of simple expert agents (e.g. information agent, simple story-teller agent).

The prototype is implemented in Java, using Servlet technology. The representation and management of structured knowledge, such as the change domain and the user model, is based on object-oriented principles including inheritance, encapsulation, and polymorphism. Object serialisation relies on a declarative formalism based on XML.

We are currently experimenting with the use of ontologies for a more structured representation of several fields of the user model (such as the user interests and experience). The first ontology based prototype makes use of KAON (see KAON (2002)), a framework for representing and manipulating ontologies based on the OIL standard (Fensel et al., 2001).

4. Related work

Community-based systems have concentrated on supporting community members in the collection, organisation, and distribution of information. For example, e-mail systems, chat rooms, and instant messaging systems have been proposed in order to encourage communication. Shared annotation systems and shared databases (Brush et al., 2001; Davis and Huttenlocher, 1995; Röscheisen et al., 1995a) have targeted the incremental development of a 'community memory'. Information forwarding and recommender systems (Glance et al., 2001; Huberman and Kaminsky, 1996; Kamiya et al., 1996; Linton et al., 2000; Maltz, 1995; Röscheisen et al., 1995b) have aimed at facilitating knowledge sharing within communities and in particular the sharing of ratings and recommendations of resources. Virtual community spaces (Carotenuto et al., 1999; Divitini et al., 1993) have aimed at creating integrated software environments promoting members' knowledge of each other whilst, like connected communities (Mamdani et al., 1999), facilitating the exchange of structured information, and creating collective memory.

The success of many of the above tools has been limited by the fact that many community members have found it difficult to integrate the use of these tools in their normal working routine. Some studies (Dore, 2001; Malhotra, 2000; Vandenbosch and Ginzberg, 1996–1997) have demonstrated that this is due to two factors. First, community members did not feel that they would have gained significant advantage from the adoption of the knowledge management behaviour that those tools would have supported, e.g. they did not see the advantages of sharing their knowledge. Second, the tools were technically inadequate (e.g. difficult to use, not integrated in the existing IT environment), or their features did not correspond to the expectation of the users.

The first failure factor, and some aspects of the second are addressed in the K-InCA system by supplying community members with a (software) teacher guiding them in the adoption of knowledge sharing behaviours, and helping them in the selection and use of the available tools.

In this sense, K-InCA can be compared to some of the technology-enhanced learning systems that have been proposed in recent years. Agent technologies and principles represent a very powerful approach for designing intelligent tutoring systems and more generally systems providing intelligent assistance that empower learning (Boy, 1997; Lester et al., 1997; Ogata and Yano, 1999). Agent-based architectures are able to address aspects that previous intelligent tutoring works (Shute and Psotka, 1994) have largely failed to approach in satisfactory manner, such as: tracking the user's activity, profiling them or dialoguing with the user. Agent technologies for Intelligent Learning Environments (Capuano et al., 2000; Cheikes, 1995; Chen and Mizoguchi, 1999; Jafari, 2001; Paiva, 1996) by personalising the interaction (Karagiannidis et al., 2001), open the possibility to design systems that maximise the quality of the knowledge acquisition process of the users. Such systems, which rely on building and establishing a relationship with the users, are based on a deep understanding of their inner learning needs and of their learning style. Such agents exploit this information to adapt (Brusilovsky, 1998) to the user's needs in order to provide the most efficient interaction.

More concretely, agents in this context can be used to monitor student progress and provide guidance and assistance when needed in a computer based learning environment.

5. Conclusions and future work

In the first part of this paper, we have analysed how members of organisations may be supported in the adoption of knowledge management behaviours. We have defined a framework in which the adoption process is modelled as a change process (that can be parameterised with respect to the behaviours that should be adopted—the change domain). In this framework, users can be guided through the change processes by a set of pedagogical strategies tuned to the individual's current activity, change state and personal characteristics.

In the second part of the paper, we have presented an agent-based system implementing the above framework: K-InCA. A set of agents guides the users by selecting and implementing the appropriate pedagogical strategies. This system can be used to help communities to improve the knowledge sharing practices of their members.

A well-experimented method for introducing knowledge management practices in organisations is to charge a change management team with the job of guiding the community members through the adoption process. Using K-InCA can be compared to assigning a human change agent to each user. Users should perceive the system as their personal change agent with whom they can consult to assess their (and the community's) state of adoption of knowledge management practices and to receive information and advice on the activities to perform.

Advantages of the K-InCA system include the following:

- The change domain and the intervention model are adapted to the community because they are those defined in the first two stages of the change process (visioning and planning).
- The behaviours in the change domain are introduced and explained to each user in a

personalised manner and at the most appropriate time by expert agents implementing acquire first knowledge strategies.
- The implementation of each behaviour is proposed when best related to the user's current activity (i.e. remaining within the user's current focus of attention).
- Knowledge about the user's social network can be used by expert agents to support forms of social learning, such as imitation and facilitation.
- Users are constantly informed about their change state (how well they are doing in the change process) and the change state of the community as a whole.
- All users can access the information related to the change project (the change domain and intervention strategies) so making the project visible to all members of the community.
- The change domain and the intervention model may be dynamically changed to reflect the findings of the reviewing and learning stage.

The work developed in K-InCA demonstrates that technology can provide deep support to communities by intervening directly at the level of (social) learning and interaction. More specifically, this work proves that agent-based technologies can support the transformation of (shared) behaviours within communities. K-InCA agents intervene with information and proposals for action within a context where users can recognise the relevance of the desired behaviours.

Our work will proceed along several different directions. We will analyse the possible advantages of supporting social and cognitive representations at a deeper level, as well as at sub-community level. Our current model of behaviours is in fact shallow in the sense that, although it can be extended to include representations of individual and social cognitive processes, at present it simply names these processes. That is to say that we do not model how individual and social motivations/goals/intentions are related to actions (for such models see, for example, Cohen and Levesque (1990a,b), Conte and Castelfranchi (1995), Jennings (1992), Pollack (1990), Roda (1994)), however, we represent such cognitive relationships within the behavioural structure. In other words, elements, such as motivations and actions, which in our model are atomic, could be further decomposed; and relationship between motivations and actions, which in our model are statically described in terms of behaviours, could be dynamically derived from the first principle of individual and social cognitive processes.

We will also work towards making the monitoring of the user's actions more transparent. We will start by monitoring the browsing of the change domain, and by observing the sender, receiver, and subjects of email messages, and we will continue by observing the use of other tools, such as database systems. A larger set of possible users' actions will be defined as needed.

More tools will be developed to allow users to monitor their own state, these will include tools supporting the graphic display of users participation to the community's activity and 'community visualisation' tools.

We will further explore knowledge engineering techniques and approaches, such as ontologies, that could allow us to better manage the knowledge of the platform whilst making it easily interoperable with the external environment, legacy systems, and new systems that may become available.

Acknowledgments

The work described in this paper was sponsored by Xerox Corporation. We would also like to thank the anonymous reviewers of this special issue of the journal for their thorough comments and very helpful suggestions, we are especially grateful to Kostas Stathis for having guided us through the review process with competence and patience.

References

Alavi, M., Leidner, D., 1999. Knowledge Management Systems: Emerging Views and Practices from the Field, Proceedings of the 32nd Hawaii International Conference on System Sciences,.

Angehrn, A., Nabeth, T., 1997. Leveraging emerging technologies in management education: research and experiences. European Management Journal 15 (3), 275–285.

Angehrn, A., Atherton, J.E.M., 1999. A Conceptual Framework for Assessing Development Programmes for Change Agents, Proceedings of the ECIS, Copenhagen,.

Angehrn, A., Nabeth, T., Subirana, B., 2001. Designing Virtual Communities to Support E-commerce Adoption, Proceedings of the e2001 eBusiness and eWork, Venezia,.

Bandura, A., 1971. Social Learning Theory, General Learning Press, New York.

Beer, N., Nobria, M., 2000. Cracking the code of change. Harvard Business Review May–June.

Boy, G.A., 1997. Software Agents for Cooperative Learning, Software Agents, AAAI Press, Menlo Park, CA.

Bruner, J., 1996. Toward a Theory of Instruction, Harvard University Press, Cambridge, MA.

Brush, B., Bargeron, D., Gupta, A., Grudin, J., 2001. Notification for shared annotation of digital documents. Microsoft Research Microsoft Corporation MSR-TR-2001-87; September 19.

Brusilovsky, P., 1998. Adaptive educational systems on the world-wide-web: a review of available technologies. Proceedings of the Workshop WWW-based Tutoring, Fourth International Conference on Intelligent Tutoring Systems (ITS'98), San Antonio, TX,.

Capuano, N., Marsella, M., Salerno, S., 2000. ABITS: an agent-based intelligent tutoring system for distance learning. Proceedings of the International Workshop on Adaptive and Intelligent Web-based Educational Systems, International Conference on Intelligent Tutoring Systems (ITS 2000), Montreal, Canada,.

Carotenuto, L., Etienne, W., Fontaine, M., Friedman, J., Newberg, H., Muller, M., Simpson, M., Slusher, J., Slusher, J., Stevenson, K., 1999. CommunitySpace: Towards Flexible Support for Voluntary Knowledge Communities, Proceedings of the Changing Places Workshop, London, UK,.

Cheikes, B.A., 1995. GIA: an Agent-based Architecture for Intelligent Tutoring Systems, Proceedings of CIKM'95 Workshop on Intelligent Information Agents,.

Chen, W., Mizoguchi, R., 1999. Communication Content Ontology for Learner Model Agent in Multi-agent Architecture, Proceedings of AIED99 Workshop on Ontologies for Intelligent Educational Systems,.

Cohen, P.R., Levesque, H.J., 1990a. Persistence, intention, and commitment. In: Cohen, P.R., Morgan, J., Pollack, M. (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, pp. 221–225.

Cohen, P.R., Levesque, H.J., 1990b. Rational interaction as the basis for communication. In: Cohen, P.R., Morgan, J., Pollack, M. (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, pp. 221–225.

Cohen, M., Stathis, K., 2001. Strategic change stemming from E-commerce: implications of multi-agent systems in the supply chain. Journal of Strategic Change 10, 139–149.

Conte, R., Castelfranchi, C., 1995. Cognitive and Social Action, UCL Press, London, UK.

Conte, R., Paolucci, M., 2001. Intelligent social learning. Journal of Artificial Society and Social Simulation 4, 1.

Davenport, T.H., Prusak, L., 1998. Working Knowledge: How Organizations Manage What they Know, Harvard Business School Press, Harvard, MA.

Davis, J.R., Huttenlocher, D.P., 1995. Shared Annotation for Cooperative Learning, Proceedings of the Computer Support for Collaborative Learning '95, Indiana University, Bloomington,.

Deroïan, F., 2002. Formation of social networks and diffusion of innovations. Research Policy 31 (5), 835–846.July.

DeSanctis, G., Wright, M., Jiang, L., 2001. Building a global learning community. CACM 44, 12Special issue on Global Applications of Collaborative Technology; December.

Divitini, M., Omodei, S.G., Pozzoli, A., Simone, C., 1993. Supporting the Dynamics of Knowledge Sharing Within Organizations, Proceedings of the Conference on Organizational Computing Systems, Milpitas, CA, USA, pp. 178–183.

Dore, L., 2001. Winning through knowledge: how to succeed in the knowledge economy. Special Report by the Financial World. The Chartered Institute of Bankers in Association with Xerox.

Dyer, J., Nobeoka, K., 2000. Creating and managing a high-performance knowledge-sharing network: the toyota case. Strategic Management Journal 21, 345–367.

Fensel, D., van Harmelen, F., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F., 2001. OIL: an ontology infrastructure for the semantic web. IEEE Intelligent Systems March/April.

Glance, N., Meunier, J.L., Bernard, P., Collaborative, A.D., 2001. Document Monitoring, Proceedings of Group 2001, Boulder, CO, USA,.

Gongla, P., Rizzuto, C.R., 2001. Evolving communities of practice: IBM global services experience. IBM System Journal 40, 4.

Grant, R.M., 1996. Prospering in dynamically-competitive environments: organizational capability as knowledge integration. Organization Science 7 (4), 375–387.July–August.

Huberman, B.A., Kaminsky, M., 1996. Beehive: a System for Cooperative Filtering and Sharing of Information, Dynamics of Computation Group, Dynamics of Computation Group, Xerox Palo Alto Research Center.

Jafari, A., 2001. Conceptualizing Intelligent Agents for Teaching and Learning, Proceedings of the International Conference on Intelligent Agents, Las Vegas, USA,.

Janssen, M.A., Jager, W., 2001. Fashions, habits and changing preferences: simulation of psychological factors affecting market dynamics. Journal of Economic Psychology 22 (6), 745–772.December.

Jennings, N., 1992. Towards a Cooperative Knowledge Level for Collaborative Problem Solving, Proceedings of the 10th European Conference on Artificial Intelligence, pp. 224–228.

Kamiya, K., Röscheisen, M., Winograd, T., 1996. Grassroots: a System Providing a Uniform Framework for Communicating, Structuring, Sharing Information, and Organizing People, Proceedings of the Sixth WWW Conference, Paris, France,.

KAON, 2002. Karlsruhe Ontology Infrastructure. http://kaon.semanticweb.org/.

Karagiannidis, C., Sampson, D., Cardinali, F., 2001. Integrating adaptive educational content into different courses and curricula. Educational Society and Technology Journal 4, 3July.

Koulopoulos, T.M., Spinello, R.A., Wayne, T., 1997. Corporate Instinct: Building a Knowing Enterprise for the 21st Century, Van Nostrand Reinhold, New York.

Lave, J., 1991. Situated Learning in Communities of Practice. Perspectives on Socially Shared Cognition, American Psychological Association, Washington, DC.

Lave, J., Wenger, E., 1990. Situated Learning: Legitimate Peripheral Participation, Cambridge University Press, Cambridge, UK.

Leonard-Barton, D., 1995. Wellsprings of Knowledge: Building and Sustaining the Sources of Innovation, Harvard Business School Press, Harvard, MA.

Lesser, E.L., Storck, J., 2001. Communities of practice and organizational performance. IBM Systems Journal 40, 4.

Lester, J., Converse, S., Kahler, S., Barlow, T., Stone, B., Bhogal, R., 1997. The Persona Effect: Affective Impact of Animated Pedagogical Agents, Proceedings of CHI'97, pp. 359–366.

Linton, F., Joy, D., Schaefer, H., Charron, A., 2000. OWL: a recommender system for organization-wide learning. Educational Technology and Society 3 (1), 62–76.January.

Malhotra, Y., 2000. Knowledge management for E-business performance: advancing information strategy to internet time. Information Strategy: The Executive's Journal 16 (4), 5–16.Summer.

Maltz, D., 1995. Pointing the Way: Active Collaborative Filtering, Proceedings of CHI'95,.

Mamdani, E.H., Pitt, J., Stathis, K., 1999. Connected communities from the standpoint of multi-agent systems. New Generation Computing Journal 17, 4Special Issue on New Challenges in Intelligent Systems.

Manzoni, J.F., Angehrn, A., 1998. Understanding organizational dynamics of IT-enabled change: a multimedia simulation approach. Journal of Management Information Systems 14 (3), 109–140.

Near, J., 1993. Organizational change game. In: Todd, J., (Ed.), Managing Change: Cases and Concepts, Irwin, Homewood, IL, pp. 241–242.

Nonaka, I., Hirotaka, T., 1995. The Knowledge-Creating Company, Oxford University Press, Oxford.

Ogata, H., Yano, Y., 1999. Combining Social Networks and Collaborative Learning in Distributed Organizations, Proceedings of the ED-MEDIA 99, AACE Press, Norfolk VA, pp. 119–125.

O'Reilly, C.A. III, Tushman, M.L., 1997. Using Culture for Strategic Advantage: Promoting Innovation Through Social Control, Managing Strategic Innovation and Change: a Collection of Readings, Oxford University Press, New York.

Paiva, A., 1996. Towards a Consensus on the Communication Between User Modelling Agents and Application Agents, Proceedings of the UM'96 Workshop on Standardization of User Modelling.

Plotkin, H., 1994. Darwing Machines and the Nature of Knowledge, Penquin, London.

Pollack, M., 1990. Plans as complex mental attitude. In: Cohen, P.R., Morgan, J., Pollack, M. (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, pp. 77–104.

Rich, E., 1979. User modelling via stereotypes. Cognitive Science 3, 329–354.

Rich, E., 1989. Stereotypes and user modelling. In: Kobsa, A., Wahlster, W. (Eds.), User Models in Dialog Systems, Springer, Berlin, pp. 35–51.

Roda, C., 1994. Task oriented dialogues in multi-agent systems. PhD Dissertation, Electrical and Electronic Engineering Department, Queen Mary and Westfield College, University of London, London.

Roda, C., Angehrn, A., Nabeth, T., 2001. Matching Competencies to Enhance Organisational Knowledge Sharing: an Intelligent Agents Approach, Proceedings of the Seventh International Netties Conference, Fribourg, Switzerland, pp. 931–937.

Rogers, C.R., 1969. Freedom to Learn, Merrill, Columbus, OH.

Rogers, E.M., 1995. Diffusion of Innovation, Free Press, New York.

Röscheisen, M., Mogensen, C., Winograd, T., 1995a. Beyond Browsing: Shared Comments, SOAPs, Trails, and On-line Communities, Proceedings of the Third International World-Wide Web Conference, Darmstadt, Germany,.

Röscheisen, M., Winograd, T., Paepcke, A., 1995b. Content rating and other third-party value-added applications for the world-wide web. D-Lib Journal August.

Shute, V., Psotka, J., 1994. Intelligent tutoring systems: past, present and future. In: Jonassen, D., (Ed.), Handbook of Research on Educational Communications and Technology,.

Stuart, A., 1996. Knowledge management. CIO Magazine June 1.

Sumner, T., Domingue, J., Zdrahal, Z., Millicam, A., 1999. Moving from On-the-job Training Towards Organisational Learning, Proceedings of the KAW'99 12th Workshop on Knowledge Acquisition, Modelling and Management, Alberta, Canada,.

Teece, D.J., Pisano, G., Shuen, A., 1997. Dynamic capabilities and strategic management. Strategic Management Journal 18, 509–533.

Vandenbosch, B., Ginzberg, M.J., 1996–1997. Lotus notes and collaboration: plus ça change. Journal of Management Information Systems 13 (3), 65–82.Winter.

Want, J.H., 1995. Managing Radical Change: Behind Survival in the New Business Age, Wiley, New York.

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