



Organizational learning networks that can increase the productivity of IT consulting companies. A case study for ERP consultants



Razvan Bologa*, Ana Ramona Lupu

Computer Science Department, Academy of Economic Studies, Piata Romana 6, Bucharest, Romania

ARTICLE INFO

Keywords:

Learning social networks
Knowledge sharing
Software development education
Enterprise resource planning
ABAP

ABSTRACT

This paper considers the use of social learning networks to increase the productivity of IT consulting companies. We advocate that using a carefully designed social learning network can reduce the learning time for enterprise software developers and consultants. By viewing learning as a social act, a consulting company can increase its productivity. Increased productivity is based on hastening the learning process. The focus of this paper is to identify the ways in which social networks catalyze the process of knowledge sharing in order to increase the productivity in the enterprise resource planning (ERP) consulting sector. We present a set of detailed practical results that were obtained from experiments with an original knowledge sharing method that was applied to training young software developers to enable them to work for some of the world's most demanding IT companies. The experimental data were collected from 2004 to 2011 during 12 training sessions conducted by an IBM partner in conjunction with the Computer Science Department of a large Eastern European University. The main results of this study were: (1) designed a learning community that reduced the time needed to insert junior consultants into ERP projects; and (2) statistical data were generated that measured the increase in productivity that an ERP consulting company could obtain by employing organizational learning networks. We also discuss the positive impacts of social networks that can be established between private companies and universities.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

There has been substantial research in the area of organizational learning. By definition, organizational learning is the acquisition of information knowledge and skills by individuals (Argyris & Schon, 1995; Easterby-Smith & Lyles, 2011). Organizational learning networks have been previously studied with regard to their relationships with the productivity of IT consulting companies. Skerlavaj, Dimovski, Mrvar, and Parhor (2010) found that social learning networks had a positive impact inside IT consulting companies, including those involved in enterprise resource planning (ERP) consulting.

One of the most prominent researchers in this field claimed that the only sustainable competitive advantage for a company was its ability to learn faster than its competitors (Geus, 1988; Kumar, Jones, Venkatesan, & Leone, 2011). Learning is crucial for an IT consulting company because it only sells knowledge and, thus, it has to be better than its competitors.

A consulting company's productivity is not always easily defined. Nachum (1999) observed that productivity models

developed in the manufacturing industry did not apply to professional companies. These models depend on the particular type of service that is being offered.

The main issue that an IT consulting company must solve when it comes to its productivity is the considerable time needed to insert a consultant into a project that is unknown to that consultant. In terms of ERP software packages, the time that a person must spend learning before (s)he can start working autonomously is much longer than in other industries.

According to Lazowska (2011), who cites a report from the US Bureau of Labor Statistics, we are currently facing a shortage of computer specialists that will continue to grow until at least 2018. In the US alone, from 2008 to 2018 the job market will be in demand for an extra 1.4 million new computer specialists. This is not a new situation, as it has been occurring for many years. The US President's Council of Advisors on Science & Technology stated in December 2010 that "all indicators – all historical data, and all projections – argue that (computer science) is the dominant factor in America's science and technology employment" (Lazowska, 2011).

This shortage of skills also applies to ERP consulting companies, as ERP has been a hot sector since the mid 1990s. This paper was written based on the assumption that the productivity of a consulting company in an area with a shortage of resources is directly

* Corresponding author.

E-mail addresses: razvanbologa@ase.ro (R. Bologa), ramona.lupu@ie.ase.ro (A.R. Lupu).

URL: <http://www.ase.ro> (R. Bologa).

proportional to the number of new junior consultants that it manages to successfully insert into projects over a certain period of time. Reducing the time needed to train a junior consultant in order to become stand-alone is crucial for increasing a consulting company's productivity. Additional details regarding the interpretation of productivity and statistical data that confirm this assumption are provided in the following chapters.

This paper has four main parts: (1) background, in which the problem is formulated in detail; (2) the learning model and the proposed social learning network; (3) the learning process that was conducted and the statistical data that were obtained; and (4) a discussion on productivity gains and answers to some questions and criticisms.

2. The need for faster training

The various aspects of knowledge creation are critical for sustaining the development of learning organizations (Easterby-Smith & Lyles, 2011; Selamat & Choudrie, 2007). In the IT industry, the competitiveness of a company is largely determined by the knowledge it possesses, and the knowledge of an organization is considered to be derived from its employees (Lin, 2006; Skyrme, 2012; Von Krogh, Ichijo, & Nonaka, 2000).

Due to a shortage of skills (Lazowska, 2011), providing the right training in a minimum amount of time is a critical factor for any software company that is active in today's highly competitive IT industry. This is why many companies are now looking for ways to reduce their employees' training periods and to train them in a manner that is faster than conventional on-the-job training.

To accelerate the educational process, we have developed a social framework for accelerating the sharing of knowledge between experienced software developers and trainees inside a social network. This framework consists mainly of creating a social learning network that catalyzes the knowledge sharing process and increases learning speed. This model began with the bold idea of an Italian IBM partner company that approached an Eastern Europe university in 2004. This company proposed the development of a method that would transform young IT graduates into internationally competitive ABAP programmers in six-eight months instead of the industry's standard time of two years. The endeavor that followed was successful, and the result was a framework that ensured both the transfer of explicit and tacit knowledge.

When the Italian entrepreneur launched the idea of reducing the time for educating ABAP developers from two years to six-eight months, it created mixed reactions. ABAP is the programming language for SAP, the world's largest and most sophisticated ERP solution. Since the early 1990s, most large multi-national companies have adopted SAP as an integrated information system, which has generated a huge need for skilled technical consultants (Bjorlin, 2008). ABAP has been used for over 30 years to write the code for SAP. Although it is very simple as a programming language, working in ABAP requires a thorough understanding of the way in which SAP was written over the last three decades.

Learning ABAP requires learning a vast amount of detail about SAP's database structure, the various tools used in SAP, and the way it interacts with other applications. A certain level of understanding of business procedures is also a must, as ABAP programmers often need to work in situations with fragmented specifications. In addition, because SAP is used in key processes inside large multi-national companies, the error tolerance is very low. Basically, an ABAP application that handles billions of Euros or dollars needs to work perfectly.

For these reasons, it takes considerable time to train a new university graduate in computer science to become an ABAP developer who is capable of working autonomously. By 2004, major

companies like IBM considered that the standard time for fully training an ABAP developer was 24 months. There is also a standard SAP curriculum that is used in this training process. By the time that IBM's partner had approached us, this was the standard training time accepted by the industry.

The demand for ABAP developers has always been greater than the supply (Bjorlin, 2008). This was also confirmed by the IBM managers that backed this project. This was the main reason why IBM's partner company financed this research project; they needed to reduce their training time. We found that, even after the economic crisis that began in 2008, the demand for ABAP developers remains very high.

The framework for learning developed in the project described in this paper was not limited to ABAP technology. It was actually used for other technologies that presented similar difficulties when being learned by young university graduates, such as Business Intelligence (Hyperion). Additional details regarding these technologies and the results will be described later. More and more software technologies in the IT sector are being widely adopted, which is creating a shortage in the supply of skilled consultants.

3. Knowledge management considerations

After initial discussions, we decided to become involved in the research project proposed by IBM's partner. The idea of the primary author of this paper was to use the research of Everett Rogers on the diffusion of innovations. In 1962, Rogers published a very well known sociology book entitled *Diffusion of Innovations*, which analyzed why some communities adopted innovations faster than others. This was combined with ideas from more recent research on organizational learning and how it can be used to spread knowledge inside a community (Easterby-Smith, Crossan, & Nicolini, 2000; Skyrme, 2012).

The reason for starting with the research of Rogers was that his ideas were considered to be the closest to the concept of learning innovation faster. By their nature, software technologies are highly innovative. It is quite common to see a considerable amount of innovation generated by software companies each year. We assumed that learning a sophisticated technology by a group of students was, from a sociological point of view, the dissemination of innovation at the group level.

The theory of the "diffusion of innovations" used as a background for our research proposes four main elements that influence the spread of a new idea: the innovation; communication channels; time; and a social system (Rogers, 1995).

While analyzing Rogers' theory to reduce the time needed for training software developers, we observed that, in order to reduce the time, it was necessary to improve another element among these four main elements. Essentially, if the time was to be shortened, something else had to be increased.

The first element, the innovation, could not be changed because it was fixed. The innovation that had to be disseminated in our case was the SAP and ORACLE technologies, which were already running in many companies; thus, nothing could be done to change this.

The second element, communication channels, was also something that we could not see ways to improve. The communication channels used in the IT industry are based on the most advanced technologies; thus, we decided not to improve the communication channels.

The only element that could be manipulated to reduce the time was the last: the social system. The social system was relatively unexplored in the companies with which we dealt at the beginning of this research study. In 2004, the area of organizational learning was a research field in its infancy (Easterby-Smith et al., 2000). In software companies, most of the social set-ups of teams were left

to the discretion of managers, who were not trained in the area of innovation diffusion.

Indeed, even the concept of a social system was quite difficult to define for most of the managers and entrepreneurs that participated in this research study. The definition that we used was borrowed from Rogers' theory, who states that a social system is "a set of interrelated units that are engaged in joint problem solving to accomplish a common goal. A system has structure, defined as the patterned arrangements of the units in the system, which gives stability and regularity to individual behavior in a system. The social and communication structure of a system facilitates or impedes the diffusion of innovations in the system" (Rogers, 1995, p. 23).

Rogers also discussed norms: established behavior patterns for the individuals of the system. Opinion leadership and change agents were two other important concepts introduced by Rogers; both of these concepts are also important to achieve a faster dissemination of innovation.

The theory on innovation diffusion was useful for identifying the scope of this research. It delineated where we could work to improve the efficiency of and reduce the training time. By defining a better structure for the social system, the desired result of reducing the time needed to train software developers could be achieved. However, Rogers' theory offered no indication as to what exactly could be done to actually achieve these results.

To initiate practical work, we had to go beyond the theory of Everett Rogers. A deeper understanding of the concept of a social system was required to perform a faster training process for software experts in practice. The books by Rogers and his adherents did not actually provide the necessary elements, as they focused too much on the dissemination of innovation from companies to clients rather than from trainers to trainees. Knowledge management was the area where we decided to look for the additional details that would be needed to devise a working plan.

Innovation is a form of knowledge, and programming is largely the manipulation of knowledge. Software developers work exclusively with data, information, and knowledge. Thus, it seemed natural that we should use knowledge management to explore possibilities for training ABAP programmers faster.

We started our work on the assumption that by using the techniques developed in the field of knowledge management, we could establish a social system that would work better for reducing the training time for software experts. The benefits of using knowledge management strategies to improve corporate performance were demonstrated by Lopez-Nicolas and Merono Cerdan (2011) who showed the importance of knowledge management with regard to improving corporate performance.

Our plan was to adapt some of these techniques in order to reduce the training time needed to transform an IT graduate into an internationally competitive ABAP programmer who could work for IBM's large-scale projects. We chose the research of Etienne Wenger on communities of practice to complete the theory proposed by Everett Rogers. The work by Wenger in the field of knowledge management is a powerful, highly innovative approach that we found to be very useful for the research project described in this paper.

In his book *Cultivating Communities of Practice*, Wenger, in collaboration with McDermott and Snyder, described the major principles on how to build a learning community. According to Wenger, he was trying to understand the connection between knowledge, community, learning, and identity. The basic idea is that human knowing is fundamentally a social act. This simple observation has profound implications for the way we think and attempt to support learning" (Wenger, 2012).

According to Wenger, "communities of practice are groups of people who share a concern, a set of problems, or a passion about

a topic, and who deepen their knowledge and expertise in this area by interacting on an ongoing basis. Engineers who design a certain kind of electronic circuit called phase-lock loops find it useful to compare designs regularly and to discuss the intricacies of their esoteric specialty. Soccer moms and dads take advantage of game times to share tips and insights about the subtle art of parenting. Artists congregate in cafés and studios to debate the merits of a new style or technique. Gang members learn to survive on the street and deal with an unfriendly world. Frontline managers running manufacturing operations get a chance to commiserate, to learn about upcoming technologies, and to foresee shifts in the winds of power" (Wenger, McDermott, & Snyder, 2002, p. 4).

The coinage of the phrase "communities of practice" is rather new, although the reality that it describes is very old. However, the industrial educational system has insisted quite marginally on the concept of community of practice during the last two centuries. The main concept that dominated the industrial educational system for centuries was that learning consisted mainly of transferring information and knowledge (Toffler, 1980; Tsungjuang, 2009).

Viewing knowledge as a social act is quite new, and the strength of Wenger's theory is that it adapts this centuries' old concept to modern realities. In the classical literature on organizational learning, the mind is considered as some sort of container, knowledge is similar to a substance, and the act of learning is a transfer of this substance (knowledge) to a container (mind) (Cyert & March, 1992).

Based on Wenger's approach, we considered learning to be a "social act". We started from the idea that a faster learning process could only be achieved by designing a social network that would include both trainers and trainees who studied, worked, and, in some cases, even lived in the same neighborhood.

Wenger did not analyze the possibility of using communities of practice to accelerate the learning process. This is our contribution, which we elaborate upon in this paper. We extend the concepts introduced by Wenger and show how they can be used to create social networks that reduce the time needed for transforming IT graduates into competitive software experts.

We established the architecture for a social learning network that links companies and universities to accelerate the learning process. Our original contribution consists of designing the social network used for learning, establishing a learning process to transfer knowledge, and producing a statistical data set relevant to the area of organizational learning networks.

A substantial amount of statistical data was gathered over a period of 7 years. These data have been used to perform various analyses regarding the efficiency of the process and, most importantly, to demonstrate the growth in productivity. Another important contribution is our definition of an efficiency indicator for the social learning process. Based on this indicator, decisions regarding the impact of motivational elements can be made.

Based on the research of Everett Rogers on the diffusion of innovations and that of Etienne Wenger on the communities of practice, it became clear to us that designing a social network should be the object of our research. This was also confirmed by Kim, Suh, & Jun, 2011 who stated that context, social networks, and Information Technology are the key knowledge transfer factors. In practical terms, to educate software experts in six months rather than two years, we studied the elements behind a successful community of practice and focused on how these could be adapted to the more rapid training of software developers. To obtain the desired results, we had to design and implement two main elements: a social learning network and a learning model.

The learning network had to support the learning model, and the social network was the environment in which the learning process took place. Both of these elements will be described in detail in the following chapters.

The following chapters describe the choices that we made and the results that were obtained. It should be noted that this research was successful. Apart from IBM, which was the initial beneficiary of these research efforts, there were many other large multi-national companies that have become interested in the results of this research. As of 2012, 8 years after we started of this research project, 8 multi-national companies are actively employing the results of this research. One is even using the research results to define its strategies in the IT areas that involve new and rare skills.

4. Designing the learning model

To successfully create the learning model, we had to first identify the types of knowledge that were essential. Distinguishing between explicit and tacit knowledge was a key factor. Supporting knowledge creation and sharing it in social networks is possible due to personal interactions between the members of the community. People rarely acquire critical information from impersonal sources, such as archives, electronic platforms, and other such sources (Chang, 2011; Cross, Parker, Prusak, & Borgatti, 2001).

In his work with the IBM Institute for Knowledge Management, Rob Cross conducted extensive social network analyses. He stipulates that relational qualities promote effective knowledge sharing (Cross et al., 2001).

The idea behind our model is to create good connections between the members of the social network who are involved in the learning process.

Explicit knowledge is that kind that can be verbalized (Kothari et al., 2012; Okafor & Osuagwu, 2006). This is the kind of knowledge that is available in books, reports, forums, and oral discussions. Conventional training processes are designed to transfer explicit knowledge. The most common means of transferring explicit knowledge are based on presentations and exercises.

Tacit knowledge is implicit knowledge that people retain in their minds and is difficult to access (Kothari et al., 2012; Okafor & Osuagwu, 2006). It often occurs that they are not aware of the knowledge that they possess or how it can be valuable to others. Tacit knowledge is considered more valuable because it provides a context for people, places, ideas, and experiences. The effective transfer of tacit knowledge generally requires extensive personal contacts and trust (Joia & Lemos, 2010; Nonaka & Takeuchi, 1995).

The difference between a good and a poor software expert derives from the amount of tacit knowledge that they possess. Tacit knowledge is naturally acquired over time by those individuals who work in a particular environment.

Based on our experience, the main difficulty in training software experts arises from the large amount of tacit knowledge that needs to be absorbed. Again, explicit knowledge is that knowledge that can be verbalized and can be absorbed by reading books or documents and by asking questions. Tacit knowledge is only available in the minds of people, so acquiring it is only possible by interactions. These interactions can be either real or virtual. During real interactions, people stay in the same room, while during virtual interactions they use electronic means to communicate.

To overcome the need to share both explicit and tacit knowledge, we divided the learning process into two main phases. The first was based on conventional training processes and was intended to transfer explicit knowledge. The second component of the learning model was designed to transfer tacit knowledge. Sharing tacit knowledge requires methods that are not available in conventional training processes. We decided to use scenarios, a method described by Awad and Ghaziri (2004). A scenario is a replication or a repetition of a real or a possibly real situation that allows people to share tacit knowledge.

A scenario is different from an exercise. In a scenario, the trainees are put into real life situations and asked to perform certain tasks as if they were working on real projects. The main difference between an exercise and a scenario is that a scenario cannot be solved by using only explicit knowledge. Solving the requirements of a scenario requires deep, tacit knowledge. In many cases, scenarios are actually older projects that have already been completed. Trainees are asked to solve the set of requirements of the older projects without knowing the solution.

As shown in Fig. 1, the learning model we designed contains two main parts: an explicit knowledge sharing component and a tacit knowledge sharing component. Our model was based on treating tacit and explicit knowledge differently, as with other knowledge diffusion models (Tsai, 2009). In order to share the tacit knowledge, the tacit *k*-sharing component of the system presented here uses the concept of a scenario, which derived from observing the episodic nature of the tacit knowledge to be shared. Episodic knowledge is based on experimental knowledge, or episodes (Awad & Ghaziri, 2004).

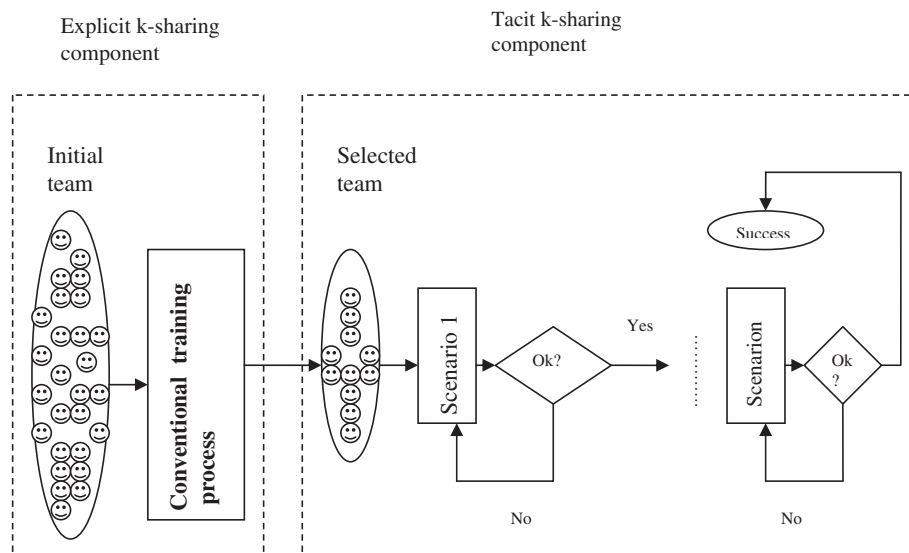


Fig. 1. The knowledge sharing system.

We developed these scenarios in conjunction with professionals from companies involved with two major technologies: ABAP and Hyperion (a business intelligence tool from ORACLE). Most of the training rounds were on ABAP. It should be noted that we had no professional knowledge of ABAP or Hyperion. To overcome our lack of tacit knowledge, older projects were chosen that were re-installed in working environments.

When using SAP and Hyperion, one can load all of a client's data into an SAP and Hyperion environment, use the client's requirements, and ask someone to work as if the project is real. Of course, a 100% replication of the real situation is not always possible because, in some cases, users are testing the outcome. However, this kind of environment allows for the easy definition of scenarios by replicating older projects.

Scenarios are important because trainees are forced to acquire tacit knowledge to solve them. At the beginning, the trainees will require the assistance of a senior programmer in order to solve the scenarios. Senior programmers have the required tacit knowledge. By working together to solve the scenarios, trainees acquire tacit knowledge from senior professionals.

For a senior professional, a scenario provides an opportunity to understand the kinds of tacit knowledge that need to be presented to the trainees while working. It is only by working on a scenario that senior professionals are able to transfer their tacit knowledge to trainees. Because tacit knowledge cannot be verbalized, senior professionals would otherwise be unable to expose the trainee to this information.

In practical terms, during the training process, the students have to undergo both an explicit knowledge sharing process and tacit knowledge sharing. At the beginning, training starts with a large team from which a specific team is selected after the explicit knowledge sharing process. The members of this selected team are chosen based on their results on a simple test. Subsequently, the students in the selected team undergo a tacit knowledge sharing process that is based on scenarios. They have to proceed through a number of scenarios and, in case they fail, the scenarios are repeated.

This knowledge sharing system was successfully tested twelve times between 2004 and 2011, and over 360 students successfully accelerated their training process by achieving in about 6 months the level of experience that would have otherwise taken 2 years of classical on-the-job training.

5. Designing the social learning network

Our main idea was to create a learning environment that could support the learning model based on scenarios as a form of transferring tacit episodic knowledge. This environment was designed as a social network in which the relationships between the members of the network were structured in such a way as to encourage social interactions that would speed up the transfer of episodic knowledge.

At the beginning of this study in 2004, the classical way of turning an IT graduate into a software expert was to place him/her into a company where they would stay "in the shadow" of a senior expert. The average time for obtaining a competitive consultant for a technology like ABAP by "shadowing" was two years.

To speed up this process, we needed a better working learning environment. The principles for designing the social network were borrowed from the previously described theory of Wenger on communities of practice.

A key issue in the success of the learning network's design was so-called "legitimate peripheral participation". This is a concept that was introduced by Wenger in his early research, which describes how newcomers are transformed into experienced members and eventually become old-timers of a community of practice or collaborative project (Lave & Wenger, 1991). In this work, they discuss the importance of encouraging weak links between the members of a community.

Encouraging weak links was the starting point of our approach to building the social learning network. According to Granovetter (1973), Xiang, Neville, and Rogati, (2010), weak ties in a social network are more important for information dissemination than are strong ties. This was also observed by Van der Leij & Sanjeev (2011) who argued that, in non-hierarchical organizations, weak ties may be important for the exchange of new information.

Based on this principle, the learning community was designed with weak links in mind. We assumed that speeding-up the training process could be obtained by encouraging the development of weak links in an educational environment. Strong links were also equally represented in the design of the social network, as each trainee was assigned to a senior person. However, most of the learning was supposed to occur from contacts with other members of the community, either trainees or senior personnel.

Creating the right structure for the social learning network was the first step for our research. The general idea was to involve four categories of people in the network: students; professors; professionals; and managers. The former two categories were from the university and the latter two were from companies.

The social network was designed based on the principle that, a social network has nodes that put tightly knit groups together and between which there are only weaker connections (Girvan & Newman, 2002). Because of this property of social networks, we decided to create a social network structure based on three main centers of connections creation: a company; a dedicated study area; and a Master's program.

Each of these centers in the social network was an area where knowledge could be transferred by social interactions between the participants during the learning process (see Fig. 2).

The university environment was created around a professional Master's program that attracted approximately 60 students each year. The trainees involved in this program were selected primarily from this Master's program. The professional Master's program was dedicated to the study of enterprise software technologies; primarily enterprise resource planning and business intelligence.

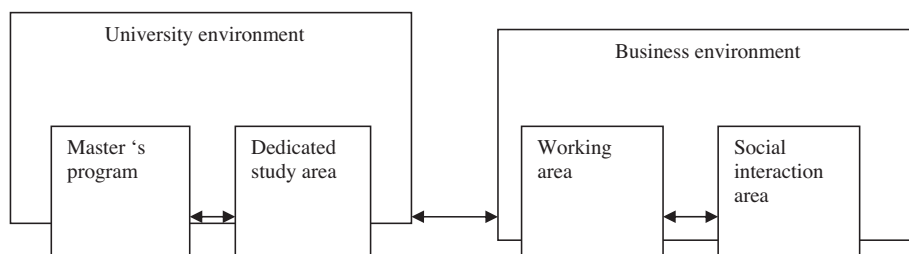


Fig. 2. Structure of the social learning network.

The curriculum of the Master's program included many subjects that used the SAP-ABAP software platform for studying. Senior consultants also participated as invited lecturers to this program.

A dedicated study area was set up at the expense of this project in order to encourage students to use this space and allow them to establish weak ties between each other. The room had a small library and 25 computers that were linked to SAP servers. Setting up this room was supported financially and organizationally by the companies that participated in the study.

The university environment was mirrored by a business environment established on the premises of IBM's partner company. Some sessions took place inside IBM's center. The business environment was divided into two main areas: a working area and a social interaction area. Both were placed in the same building and were easily accessible. Trainees were encouraged to move between both tasks in order to solve the scenarios that were given to them. We noted that a large number of issues were solved in the social interaction area. This area was both offline and online. Team-building sessions were organized in order to foster these social interactions.

All of the participants in the social network (students, professionals, managers, and professors) were encouraged to move between the four areas of the learning community to encourage the creation of links. The more links the better; thus, considerable amounts of money were spent on social learning activities.

6. Methodology

The methodology for creating the social learning network was based largely on the 6 principles needed to create a community of practice, as described by Wenger, McDermott, and Snyder (2002). Learning is considered a social act, and these principles are: design for evolution; open dialog; participation of all levels of members (active, peripheral, and outsiders); development of community spaces; focus on value; combine familiarity, excitement, and community rhythm.

In addition to these fundamental principles, a number of data procedures and course content definitions were used in order to gather experimental data over several years. Interviewing was frequently used both in the relationships with the students and in the relationships with the managers. Student interviews were based on forms and pre-defined sets of questions. These were conducted before filtering the selected team.

Interviews with managers, trainers, and professors were less formal and often conducted via e-mail. These were conducted by the authors in order to obtain the feedback needed to support

the development of the learning network. This information was very important for changing the content of the course in order to keep students interested. Similar methods were used in other studies related to knowledge sharing (Wang & Wang, 2012).

Data collection was based on attendance lists and evaluation forms. Each day, the students who were present at the training sessions had to sign a list. These lists were summarized by the staff participating in the study. Photographs were also taken in some cases in place of the attendance lists. The online evaluation forms were loaded on an e-learning platform. The online learning platform provided all of the necessary documentation and tests for evaluating students' progress.

Because a knowledge-based culture has a positive, significant influence on the performance of a company (Chang & Chuang, 2011), we considered that it was important to create a social network with roots inside the university environment. The social learning network included both academic and corporate spaces. Most of the learning took place inside the corporate space. Most of the time during the learning process, the trainees stayed within a corporate environment.

The methodology we used to organize a training session consisted of the steps shown in Table 1.

An important part of the methodology was related to course content definitions. For sharing explicit knowledge, standard documentation included with SAP and Hyperion was used. This documentation was very well organized and the trainers extracted a number of relevant topics that were presented to the students. For the tacit knowledge sharing component, the authors in conjunction with the trainers and managers identified a number of scenarios. A scenario was either an old project or an ongoing project. Each student was given the requirements for a scenario and a senior person assisted him with performing the tasks. Every two weeks, students involved in the selected team were asked to pass an evaluation that was administered by a person other than the person who had assisted them.

7. The learning process

The process represented the application of the methodology to practical situations. For a successful process, the social network had to have students. To encourage students to enter the social network, a significant number of presentations had to be conducted. The participating company promised the students jobs that would qualify for the second phase of the learning model. This was an important incentive.

The process essentially involved the steps outlined in Table 2.

Table 1
Methodology for conducting the learning process.

Step #	Description	Comments
1	Setting up the training server and acquiring the updated standard documentation (SAP-ABAP or HYPERION)	
2	Advertising the opportunity. Standard posters, presentations, and online advertorials were standard practices. Target groups were the students in their final year or students enrolled in professional Master's programs	On some occasions prizes were offered
3	Classical training using the standard documentation. The aim of this step was to allow trainees to understand the basics about the technology offered to them as a future career	Duration was 3–8 weeks. A self-study period
4	Selecting interested students. The selection criteria were based mostly on students' motivation. The idea was to weed-out those who were not motivated. This was achieved by registering the frequency and by a simple test	Until this step, all activities took place on the university's premises
5	Creating the intra-company learning network. This was the key step in this process. This was where social learning began to grow at a high speed. Only those persons who were motivated to participate could join the social network. A dedicated space was very important, as were the other six principles previously mentioned	Duration 1 week
6	Scenario-based intensive training. The last phase of the process was dedicated to transferring the tacit knowledge from person to person inside the social network that was growing. Network growth was critical to the success of the learning process. Scenarios were absolutely necessary to facilitate the transfer of episodic knowledge, which was the most important in the learning process	Duration 4–6 months
7	Gather statistical data. For each training session, relevant statistical data were collected at the end of the process	

Table 2
Steps in the training process.

#	Action according to methodology	Result
1	Advertise the training initiative	List of applicants
2	Select initial team	Initial team
3	Perform conventional training	Selected team
4	Perform scenario-based training	Junior consultants
5	Work peripherally on real remote projects	Experienced consultants
6	Interview with a client	Autonomous consultants

The difference between the process outlined above and the classical “shadowing” process is in step #4. Using scenarios as part of the process was the way in which we achieved speeding-up the training process. Additional details regarding the improvements are described in the next section of this paper.

The participants in the first two phases were university students who were enrolled in the Master’s program designed within the social network. Students who were willing to enter the learning network and were not members of the dedicated Master’s program were also accepted if there was space available. At this stage, students participated in the training during their free time. During a conventional training process, students were not hired. Training took place on the university’s premises.

At the end of step 3, a test was given and, based on the results of this test, students were selected to continue training. All of the trainees who were selected after this test were hired by the company participating in this process. Once they were hired, they spent most of their time within the business environment.

Most of the learning occurred during the second phase. Groups of 3–4 trainees were assigned to a senior consultant. The senior consultants used an e-learning platform to provide scenarios to be solved by the trainees. The assistance of the dedicated senior personnel, who also worked as consultants on a real project, was only partial and insufficient for solving the scenarios. The trainees had to navigate by themselves inside the social network and negotiate the support of their colleagues and other senior personnel in order to solve the requirements of the scenarios.

Negotiating the support of other senior personnel or colleagues was neither simple nor easy. Because each senior consultant had a very limited amount of time available, the assistance they could provide was also limited. This also applied to the peer trainees who had already acquired the requisite knowledge. This motivated the trainees to be proactive and acquire as much information as possible during their contact with a savvy person.

For efficient navigation inside the social network, the trainees spent much of their time socializing. In this case, learning was a process of establishing ties and using them to acquire tacit knowledge. Weak links between community members proved to be very important when attempting to facilitate contacts with other persons who happened to have the time and the mental acumen to help a trainee.

To encourage the effort needed to establish ties, the trainees were hired and senior personnel were paid extra bonuses based on their success. This was an important part of the model.

A critical moment was the last step when the trainees got the opportunity to talk to a real client. For those unfamiliar with the procedures of the consulting business, before a client of a company such as IBM accepts a person to work on their projects, the client will intensively interview that person.

This happens mostly because, in the IT consulting business, clients pay on a Time and Material basis, so they have to check that the consultants the system integrators assign to their projects have the skills that are required. These interviews are very serious and often take place in stages. It is not easy for a consultant to qualify because their knowledge is carefully checked. Because there is money involved, clients do not use formal tests.

Because of these tests, we could validate that our training methods actually worked. By passing such an interview, a consultant becomes billable for working on the premises of the client, which then certifies that the learning process has sped up the transfer of tacit knowledge.

Providing feedback knowledge to the decision makers is an important part of any knowledge sharing platform (Huang & Lin, 2010). During our process, feedback was provided during regular meetings with the decision makers. These meetings were part of the community rhythm.

8. Results and statistical data

Between 2004 and 2011, 12 training sessions were conducted with various set-ups. The technologies chosen for these sessions were SAP ABAP and Hyperion (subsequently acquired by ORACLE). The choice of these technologies was made by the partner company based on the needs of the market.

In 2004, both ABAP and Hyperion had shortages of skilled consultants for international markets. This remains valid today based on the perceptions of the companies involved in the research project described in this paper. Both software platforms require numerous skills that are limited in terms of availability and are very specialized.

The 12 training sessions were registered by us with the aid of an online platform used for applications and for training. The first author of this paper maintained continuous connections with the business environment where the training took place during the duration of this study. A dedicated office was established within the business environment where the authors could follow the process and participate in meetings.

Meeting the clients was an important task for us as we had to explain the model to the managers of client companies. During the 7 years of this study, we travelled all over Europe in order to explain the learning process to the managers who were supposed to interview the consultants that came out of the process described above.

The statistical data that we accumulated are summarized in Table 3.

The column labeled “Average time until successful interview” indicates the number of months a trainee needed to pass an interview with a client of the consulting company. To get to this point, a trainee had to study, solve scenarios, and work remotely under the direct assistance of a senior consultant (Table 4).

This time would have been 24 months based on the classical way of training new consultants by “shadowing”. It will be noted that these average values range from 3.7 to 15 months, with most sessions being successful after 5–6 months. For the consulting companies that were involved, this was a highly successful reduction in the time needed for training, from 24 months to roughly 5–6 months. This allowed IBM and its partner to successfully deliver highly demanded skilled resources to the consulting market.

However, not all of the sessions were efficient. It will be noted that the 2 sessions that took place in the city of Chisinau were much longer than the sessions that took place in Bucharest. In one of these sessions, the trainees needed about 16 months to get to a level where they were able to pass interviews with clients, while in the other session the time was nearly 15 months. In these two cases, we think that the differences resulted because the students came from a different university that did not have the same quality in terms of education delivered to the students. These students also had difficulties with speaking foreign languages. This can explain why these sessions produced such different outcomes.

Another interesting observation is that, for the 5th session, the time needed was considerably shorter than for the others. During

Table 3

Statistical data recorded during this study.

#	Technology	Size of initial team	Size of selected team	Average time until successful interview (months)	Location	Incentives
1	ABAP	50	15	6	Bucharest	Employment
2	ABAP	38	10	5.6	Bucharest	Employment
3	ABAP	36	10	6.2	Bucharest	Employment
4	ABAP	80	26	5.4	Bucharest	Employment
5	ABAP	376	106	3.7	Bucharest	Employment + Car
6	ABAP	90	35	15.8	Chisinau	Employment + Moto
7	HYPERION	25	7	7.2	Bucharest	Employment
8	HYPERION	23	5	6.4	Bucharest	Employment
9	ABAP	70	16	5.2	Bucharest	Employment
10	ABAP	65	14	5.6	Bucharest	Employment
11	ABAP	73	20	5.4	Bucharest	Employment
12	ABAP	38	8	14.6	Chisinau	Employment + Computers

Table 4

Efficiency indicators determined for the different sessions.

Session#	Technology	Size of initial team	Size of selected team	T	P%	E
2	ABAP	38	10	5.6	26.32	2.29
3	ABAP	36	10	6.2	27.78	2.22
4	ABAP	80	26	5.4	32.50	2.57
5	ABAP	376	106	3.7	28.19	3.04
6	ABAP	90	35	15.8	38.89	1.91
7	HYPERION	25	7	7.2	28.00	2.06
8	HYPERION	23	5	6.4	21.74	1.96
9	ABAP	70	16	5.2	22.86	2.27
10	ABAP	65	14	5.6	21.54	2.12
11	ABAP	73	20	5.4	27.40	2.38
12	ABAP	38	8	14.6	21.05	1.29

this session, the organizers decided to offer a car as a prize to the student who achieved the highest grade on the test given after the initial k-sharing knowledge (see Fig. 1). The car was exhibited in front of the university for 3 months and motivated the students to study harder during the first phase. This prize also attracted better students, which increased the efficiency of the learning process.

The size of the selected team relative to the size of the initial team is also important because it provides an indication of how many people must be attracted for a certain session to make certain that, at the end of the session, there will be enough trainees to go onto the second stage.

9. Impact on productivity

As mentioned previously, the productivity of professional companies requires different models than those developed for the manufacturing industry (Nachum, 1999). In the area of software development, individuals learn from experience and this learning process improves the productivity of software companies (Boh, Slaughter, & Espinosa, 2007). Knowledge sharing has a direct influence on a firm's performance (Wang & Wang, 2012). Because of this, we consider that it is important to analyze its impact on productivity.

In the specialty literature, it is accepted that learning is very important for software companies (Kukko & Helander, 2012; Was-tell, 1999). According to Boh et al. (2007), "Developers' current productivity in completing individual work on a system is positively related to the number of different developers that they have worked with in previous MR's". An important aspect to be kept in mind when defining a model for measuring productivity in software development is the learning curve. In a classical industrial environment, tasks are repetitive (Argote, 1999). This is not the

case for the software industry, which is more about problem-solving and is based on certain methodologies.

The methods to measure the productivity of a knowledge worker are still being debated. There are no universally accepted means to measure productivity in professional companies (Ramírez & Nembhard, 2004). For the companies with which we were involved for this study (IBM and one of its core suppliers), individual productivity was defined as the utilization rate per person. The utilization rate was obtained by dividing the number of hours that clients were actually billed to the number of hours billable to clients per person. This was the way in which the managers of the partner companies computed their productivity. Idle time (also called "bench time") decreases the productivity of a software consulting company, as these hours are not billed to any client while costs are still paid because the consultants' payroll needs to be covered each month.

Our initial assumption was that the better the social network and the scenarios in our model were, the greater would be the increase in a company's productivity. We also defined the concept of global productivity as the total utilization rate during the time a person spends inside the company employing him(her). This takes into account that consultants may resign and join rival companies. For instance, a person may resign after five years of which (s)he had spent 5 months learning while being paid by the consulting company. The global productivity (utilization rate) of this person would be: $(60 - 5)/60 = 91\%$.

The annual turnover among the teams that we worked with at the beginning was around 20%. That is, each year 20% of the members of the team resigned and joined other companies that offered better salaries. This might seem surprising to those unfamiliar with the booming ERP consulting market of the mid-2000s. However, this was the reality at that time. This rate has decreased since the start of the financial crisis in 2008. However, as of 2012 in some Eastern European regions, it has remained high because many Western European corporations have opened offices in order to cut costs in their home countries.

This high turnover rate has an impact on the productivity of a company because each new hire will have to be trained, which produces idle time. A constant flow of new resources is a must for a consulting company that is active in an area with a shortage of resources. The time that a person spends learning is non-billable for the company that employs that person. For this reason and to increase productivity, a company must decrease its learning time while still offering a good level of services. Developing in complex software systems, such as ERPs, is a sophisticated activity that requires abstract, practical, and theoretical skills (Sacks, 1994). A developer has to acquire as many of these skills as possible in order to increase the company's productivity. Based on the above consideration, we considered that to increase a company's productivity,

we had to insert more resources in a shorter amount of time. We referred to this as “being efficient”. The more people that could be properly inserted over a shorter period of time, the better would be the results for the company.

To make more accurate observations regarding the impact of the social network, we formulated a mathematical model to measure the efficiency of each training session. This model estimated the efficiency of speeding-up the knowledge sharing process and was based on those considerations related to measuring productivity as the utilization rate. The efficiency indicator was defined in conjunction with the company managers. This indicator pointed out the reductions in training time provided by considering learning as a social act rather than following the classical learning models. The managers also wanted a financial indicator. However, we found no way to implement a financial performance index. In our discussions with managers, we observed that it was not easy to clearly show the company’s management the exact financial advantages of implementing knowledge management. As stated by Tseng (2008), measuring the performance of knowledge management should not be based on financial information.

This indicator defines a global efficiency indicator, E , which is dependent on two variables, T and P :

$$E = \frac{Tm}{T} + \frac{P}{Pm} \quad (1)$$

where,

T = The amount of time needed by a junior to reach intermediate status (considered to be two years of on-the-job training by the companies we worked with);

P = The percentage of students from the initial team that made the selected team after the test was given;

Tm = Average amount of time needed by a trainee to achieve the amount of knowledge that would normally require two years of on-the-job training;

Pm = Average percentage of students from the initial team that made the selected team after the test was given.

This formula is a simple, realistic measure of the efficiency of the process, as the indicator E increases as the time, T , becomes shorter and as the percentage, P , increases. This indicator is important because the time needed to achieve the desired result is not the only indicator. The number of trainees that reach the end of the process is also important. The models for evaluating knowledge sharing are limited (Kuah, Wong, & Tiwari, 2013) because managers try to think based on established management patterns. We found that judging the efficiency of knowledge sharing required new concepts and that, at least for the time being, managers are simply not ready to accept these. The current literature on this subject is also limited (Kuah et al., 2013) and was of little use. This is why we consider that the current efficiency indicator is a good starting point. We plan to extend our research by defining an efficiency indicator based on Social Network Analysis (SNA).

However, the statistical data that has been acquired thus far has to be extended. This will require a number of new sessions and a dedicated software tool that remains to be developed. SNA is probably the best means available to measure the efficiency of a learning network and it will be the subject of our future research.

10. Efficiency factors

The efficiency indicator described above can be used to measure the impact of various motivational factors on the success of the process. As is readily seen in the data in Table 3, the time needed to achieve the desired level of success varied greatly between sessions. In most sessions, the objective of reducing the training period was met. However, the results were different with some sessions.

Over the years, we have attempted to increase the efficiency of the knowledge sharing process by motivating the students to study harder in order to obtain quicker results. To improve the results, we have tested this method with two technologies in two different locations and added various incentives to motivate the students to enter the learning program.

There were two main types of motivational factors introduced during the various training sessions: prizes and interesting job opportunities. The aim of this analysis was to determine which motivational factors, technologies, and locations worked better and to compare them in terms of efficiency using the indicator defined in Eq. (1).

10.1. Motivational factors

We were interested in determining if students were more responsive to getting a good job or obtaining an interesting prize, such as a car. To solve this problem, we ran a comparative analysis by considering the various motivational factors used in the 12 sessions described in Table 1.

During the educational process, we used interesting job opportunities and attractive prizes to motivate students to study harder and to more quickly achieve the desired result. To make a comparative analysis between the motivational factors in terms of their efficiency, they were not used simultaneously. In each session, only one motivational factor was used.

Due to reasons of availability, during the first four sessions we used interesting job opportunities in the SAP community to attract students and to motivate them to learn. This was changed during the 5th session when a high value prize (a full options car) was offered as a prize for the best student. The car was placed in front of the computer science department for a period of four months, after which the students were evaluated. No job opportunities were promised to make the comparative analysis more accurate.

The data needed for this analysis were recorded over the years and are summarized in Table 1. It essentially represents the input for the efficiency indicator, E , described in Eq. (1).

Using the model defined in Eq. (1), we computed the efficiency indicator E for each session and compared these results.

The values of the parameters Pm and Tm are readily determined as they are the average values of the last two columns of Table 1: $Pm = 26.93$ and $Tm = 7.37$. It will be noted that the average value of the efficiency indicator is >2 for most of the sessions. However, session number 5 is a notable exception because its efficiency indicator is >3 (3.04). This is quite a remarkable result because no jobs were promised to the students during this session.

This was a clear indication that high value prizes attracted and motivated students better than interesting job opportunities and, according to our indicator, the efficiency of high value prizes was much higher.

10.2. Effect of location

Based on the data in Table 2, the training during sessions 6 and 12 had much lower efficiency indicators than the other sessions. The average of the indicators for these 2 sessions was 1.6, while the other 10 sessions had an average efficiency indicator of 2.32.

The difference between these sessions was the location where they were held. Sessions 6 and 12 took place in a different city than the other 10 sessions. The students came from different educational institutions, although all of them had degrees in computer science.

The educational models that were applied were identical and the incentives were even stronger for these two sessions with poorer results. This supports the assumption that the quality of the students that are put into the process is very important. The

training process may produce very different results depending on the university from which the students graduated.

The two universities involved were very different in terms of their traditions. The university in Bucharest had well-established processes that were developed starting in the late 1950s when the first computer science courses were taught. The second university was much newer and was created during the 1990s based on the paradigms of that period. Although formally the two universities had more or less the same processes, the reality was different. In the second university, the staff was less experienced and the actual implementation of the processes was not optimal.

This suggests that this method works best with students from universities with certain traditions that run according to well-established and matured internal processes. The authors plan to run similar sessions in other countries (Cyprus is in preparation) and the new data will allow for further analysis of this issue.

A question that was raised more than once was whether or not the method could be applied to current employees. That is, could one use social learning networks to re-train employees from sectors with uncertain futures? We must point out that about 50% of our students were employed and were attending so-called professional master programs. Based on the Bologna process, professional master programs were introduced in European universities. These are programs that are generally designed for persons who want to make career changes. The young bank employees who wanted to make careers in IT were frequent participants in our training sessions. These people were 25–30 years of age and were eager to start a career in IT.

It should also be noted that in Eastern Europe, jobs in IT are very popular because the salaries are considerably higher than those in other economic sectors. This is mostly because Eastern Europe is less developed than Western Europe and many companies from Western Europe outsource their activities in the Eastern part of the EU.

So, the answer to the question as to whether the method would also work for employed persons is yes. In our case, it worked for young people. We still have to investigate the influence of age on this method. It is unclear whether the efficiency will remain the same if the participants' ages increase.

10.3. Effect of technology

Looking at sessions 7 and 8 that focused on HYPERION while the others focused on ABAP, one can observe minor differences in terms of efficiency. The average efficiency indicator for sessions 7 and 8 was 2.01. The other 10 sessions that used ABAP as a technology had an average efficiency indicator of 2.23.

Thus, it was a little more efficient to apply this method using ABAP rather than Hyperion. However, the difference in efficiency is small. The software technology used has some effect on the efficiency of the training process, but this effect is minimal. For our data, the effect of location was much greater than the effect of the technology used.

11. Comments and criticism

The first results of the current research, which began in 2004, were initially communicated at a scientific conference held in 2006. A number of other results that were obtained during subsequent years were communicated at similar conferences held in Europe and Asia. Over time, we received various comments and criticisms from different sources. We will address the major critiques here.

- Why is two years the standard industry training time? Perhaps shortening the allotted time was all that was needed? Our response: In a normal consulting company, the time

required by a junior member to achieve an intermediate level was standardized at this value, which was communicated to us by the partner companies and confirmed by all others. For those familiar with the ERP consulting industry, this reality is well known. We found no scientific reports in the literature that could confirm this exact value, but it is found in the taxonomies of major consulting companies like IBM. In some isolated cases, consultants can speed up the transfer of knowledge due to exceptional personal skills. However, the average time for a larger team to get from a junior to an intermediate level is still considered to be 2 years. By using a social learning network, we could reduce this time from two years to roughly six months.

- In what way was the original education/training set-up not a proper social system? What was wrong with it? Our response: As stated previously, the social systems used before by the companies involved in this study were social learning networks that had developed without any specific design. These were communities of practice that had developed based on the industrial idea that education means transferring knowledge to the brain just like one transfers a substance to a container. The major change was that we introduced a carefully designed learning network, which subsequently reduced the learning time significantly.
- What is the general effectiveness of this method? Why was this method applied only to a selected team? Our response: selection was necessary to eliminate those who were not really interested in the technology. Because the initial training session was short (2–3 weeks), it delivered no effective knowledge. It was more a presentation to allow the students to understand the content of the course that would follow. This study demonstrates that this method works for those who are interested in becoming consultants in the field of ERP and BI development.

12. Conclusions

The first important conclusion is that using a carefully designed social learning network reduces the learning time in the some ERP technologies. By choosing the design presented here, this reduction in training time can be highly significant from 24 months, which is the “classical” industry standard, to 6–7 months. This is an important means to bridge the gap between the level of the average university graduate and the skills demanded by international consulting companies.

A second conclusion is that the efficiency of this process is greatly influenced by factors that can be controlled by the organizers. Incentives are very important in this process and choosing them carefully has a major impact on the efficiency of the process. Seductive prizes appear to have the greatest impact on efficiency. A possible explanation for this is that learning occurs by interacting within a social network and such prizes stimulate the creation of networks.

These results can be used to design strategies for accelerating the educational process of young software developers who are needed in today's rapidly growing knowledge-based economy. We are now in discussions with many large companies that are interested in creating teams of skilled software experts in Eastern Europe.

Other investigators who want to use this method are advised to pay attention to the participants' ages, the maturity of the computer science department of the university they are partnering with, and ensure that there is active support by consulting companies where the students will acquire their episodic knowledge. The sponsorship of high level management is mandatory for successfully creating a social learning network. A social learning network

is like a living organism, as it needs to be cared for on a daily basis to ensure its proper growth.

Regarding future research, we intend to use different motivational factors and test the efficiency of these sessions. Another area of interest is to use a tool for mapping the links that are created between the members of the learning community. This tool should track the messages exchanged by the members and identify the relationships among them.

Acknowledgment

Research support from IBM Global Delivery Center and Crystal System, an IBM core supplier, is gratefully acknowledged. We wish to thank Mr. Canzio Dovigo and Mr. Hary Oлару for their support.

References

- Argote, L. (1999). *Organizational learning: Creating, retaining, and transferring knowledge*. Norwell: Kluwer.
- Argyris, C., & Schon, D. A. (1995). *Organizational learning II – theory, method and practice*. Reading, Mass.: Addison Wesley. ISBN 9780201629835.
- Awad, M. E., & Ghaziri, M. H. (2004). *Knowledge management*. Upper Saddle River, NJ: Prentice Hall. ISBN 9780130348203.
- Bjorlin, C. (2008). SAP seeks to overcome skills shortage. Available at <<http://searchsap.techtarget.com/news/1314379/SAP-seeks-to-overcome-skills-shortage>>, last accessed May 2011.
- Boh, W. F., Slaughter, S., & Espinosa, J. A. (2007). Learning from experience in software development. *Management Science*, 53, 1315–1331.
- Chang, K. C. (2011). Close but not committed? The multiple dimensions of relational embeddedness. *Social Science Research*, 40(4), 1214–1235.
- Chang, T.-C., & Chuang, S.-H. (2011). Performance implications of knowledge management processes: Examining the roles of infrastructure capability and business strategy. *Expert Systems with Applications*, 38(5), 6170–6178.
- Cross, R., Parker, A., Prusak, L., & Borgatti, S. P. (2001). Knowing what we know: Supporting knowledge creation and sharing in social networks. *Organizational Dynamics*, 30, 100–120.
- Cyert, R. M., & March, J. G. (1992). *A behavioral theory of the firm* (2nd ed.). Cambridge, MA: Blackwell. ISBN 9780631174516.
- Easterby-Smith, M., Crossan, M., & Nicolini, D. (2000). Organizational learning: Debates past, present and future. *Journal of Management Studies*, 37, 783–796.
- Easterby-Smith, M., & Lyles, M. A. (Eds.). (2011). *Handbook of organizational learning and knowledge management*. Wiley.
- Geus, A. P. (1988). Planning as learning. *Harvard Business Review*, 88, 70–74.
- Girvan, M., & Newman, M. E. J. (2002). Community structure in social and biological networks. *PNAS*, 99, 7821–7826.
- Granovetter, M. S. (1973). The strength of weak ties. *American Journal of Sociology*, 78, 1360–1380.
- Huang, C. C., & Lin, S. H. (2010). Sharing knowledge in a supply chain using the semantic web. *Expert Systems with Applications*, 37(4), 3145–3161.
- Joia, L. A., & Lemos, B. (2010). Relevant factors for tacit knowledge transfer within organizations. *Journal of Knowledge Management*, 14(3), 410–427.
- Kim, S., Suh, E., & Jun, Y. (2011). Building a knowledge brokering system using social network analysis: A case study of the Korean financial industry. *Expert Systems with Applications*, 38(12), 14633–14649.
- Kothari, A., Rudman, D., Dobbins, M., Rouse, M., Sibbald, S., & Edwards, N. (2012). The use of tacit and explicit knowledge in public health: A qualitative study. *Implementation Science*, 7(1), 20.
- Kuah, C. T., Wong, K. Y., & Tiwari, M. K. (2013). Knowledge sharing assessment: An ant colony system based data envelopment analysis approach. *Expert Systems with Applications*, 40(8), 3137–3144.
- Kukko, M., & Helander, N. (2012). Knowledge sharing barriers in growing software companies. In *45th Hawaii international conference on System science (HICSS) IEEE 2012* (pp. 3756–3765).
- Kumar, V., Jones, E., Venkatesan, R., & Leone, R. P. (2011). Is market orientation a source of sustainable competitive advantage or simply the cost of competing? *Journal of Marketing*, 75(1), 16–30.
- Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. New York: Cambridge University Press.
- Lazowska, E. (2011). Red hot: The computer science job market. Available at <<http://www.xconomy.com/seattle/2011/05/17/red-hot-the-computer-science-job-market/>>, last accessed May 2012.
- Lin, H. F. (2006). Impact of organizational support on organizational intention to facilitate knowledge sharing. *Knowledge Management Research & Practice*, 4, 26–35.
- Lopez-Nicolas, C., & Merono Cerdan, A. (2011). Strategic knowledge management, innovation and performance. *International Journal of Information Management*, 31, 502–509.
- Nachum, L. (1999). Measurement of productivity of professional services: An illustration on Swedish management consulting firms. *International Journal of Operations & Production Management*, 19, 922–950.
- Nonaka, I., & Takeuchi, H. (1995). *The knowledge-creating company*. London: Oxford University Press.
- Okafor, E. C., & Osuagwu, C. C. (2006). The underlying issues in knowledge elicitation. *Interdisciplinary Journal of Information, Knowledge, and Management*, 1, 96–108.
- Ramirez, Y. W., & Nembhard, D. A. (2004). Measuring knowledge worker productivity: A taxonomy. *Journal of Intellectual Capital*, 5, 602–628.
- Rogers, E. (1995). *Diffusion of innovations*. Ann Arbor: University of Michigan Free Press.
- Sacks, M. (1994). *On-the-job learning in the software industry*. Westport: Quorum Books.
- Salamat, H. S., & Choudrie, J. (2007). Using meta-abilities and tacit knowledge for developing learning based systems. A case study approach. *The Learning Organization*, 14, 321–344.
- Skerlavaj, M., Dimovski, V., Mrvar, A., & Parhor, M. (2010). Intra-organizational learning networks within knowledge intensive learning environments. *Interactive Learning Environments*, 18, 39–63.
- Skyrme, D. (2012). *Knowledge networking: Creating the collaborative enterprise*. Routledge.
- Toffler, A. (1980). *The third wave*. New York: Bantam Books.
- Tsai, C. M. (2009). The knowledge diffusion model associated with innovative knowledge. *Expert Systems with Applications*, 36(8), 11323–11331.
- Tseng, S. M. (2008). Knowledge management system performance measure index. *Expert Systems with Applications*, 34(1), 734–745.
- Tsunguang, W. (2009). Rethinking teaching with information and communication technologies (ICTs) in architectural education. *Teaching and Teacher Education*, 25(8), 1132–1140. ISSN 0742-051X.
- Van der Leij, M., & Sanjeev, G. (2011). Strong ties in a small world. Review of Network Economics, 10. doi: 10.2202/1446-9022.1278. Available at <<http://www.bepress.com/rne/vol10/iss2/1>>.
- Von Krogh, G., Ichijo, K., & Nonaka, I. (2000). *Enabling knowledge creation*. New York: Oxford University Press.
- Wang, Z., & Wang, N. (2012). Knowledge sharing, innovation and firm performance. *Expert Systems with Applications*, 39, 8899–8908.
- Wastell, D. (1999). Learning dysfunctions in information system development: Overcoming the social defenses with transitional objects. *MIS Quarterly*, 23, 581–600.
- Wenger, E. (2012). <http://wenger-trayner.com/wp-content/uploads/2012/01/06-Brief-introduction-to-communities-of-practice.pdf>, last accessed May 2012.
- Wenger, E., McDermott, R., & Snyder, W. (2002). *A guide to managing knowledge: Cultivating communities of practice*. Boston: Harvard Business School.
- Xiang, R., Neville, J., & Rogati, M. (2010). Modeling relationship strength in online social networks. In *Proceedings of the 19th international conference on World Wide Web* (pp. 981–990). ACM.