

Association for Information Systems AIS Electronic Library (AISeL)

AMCIS 2009 Proceedings

Americas Conference on Information Systems
(AMCIS)

2009

SimKnowledge - Multi agent-based simulation of knowledge sharing

Rene Peinl

IPI GmbH, rene.peinl@hotmail.com

Ronald Maier

University of Innsbruck, ronald.maier@uibk.ac.at

Follow this and additional works at: <http://aisel.aisnet.org/amcis2009>

Recommended Citation

Peinl, Rene and Maier, Ronald, "SimKnowledge - Multi agent-based simulation of knowledge sharing" (2009). *AMCIS 2009 Proceedings*. 790.

<http://aisel.aisnet.org/amcis2009/790>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2009 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

SimKnowledge - Multi agent-based simulation of knowledge sharing

René Peinl

IPI GmbH, Germany
rene.peinl@hotmail.de

Ronald Maier

Dept. of Information Systems, University of
Innsbruck, Austria
ronald.maier@uibk.ac.at

ABSTRACT

Even though there is abundant literature on successful cases of organizations applying knowledge management (KM) instruments, many KM initiatives have failed to achieve their knowledge and business goals. In order to foster decisions about the design of such initiatives, information is required on success factors and barriers when selecting KM instruments. Multi-agent based simulation (MABS) is suggested as an instrument to investigate potential effects of KM instruments on dependent variables such as knowledge goals, e.g., sharing of knowledge in organizations, or business goals, e.g., business performance. For such a simulation, the concept of knowledge sharing, influencing factors and their impact on business and knowledge goals have to be operationalized. This paper presents such a model which is based on an extensive multi-disciplinary literature survey. The model is implemented in a MABS tool used for a series of experiments contrasting results with/without KM instruments. Finally, the paper discusses results, limitations, further enhancements and practical implications of the simulation.

Keywords

multi agent-based simulation, knowledge sharing, knowledge management, software industry, team work

INTRODUCTION

Work in organizations is increasingly information- and knowledge-intensive and the share of knowledge work has risen continuously during the last decades (Wolff, 2005) so that organizations are striving to improve productivity of knowledge work (Drucker, 1994). During the last twenty years, KM has been suggested to help solving these issues. Numerous fragmented KM measures or instruments have been proposed which claim to solve particular knowledge-related problems (Alavi/Leidner 2001, Maier et al. 2009) implemented in many organizations with the help of KM initiatives (Davenport/Probst 2002). However, many of these have failed to achieve their goals (Bishop et al. 2008). Information is rare on how KM measures will impact business performance or performance with respect to knowledge goals, thus rendering the decision difficult about which KM measure to pick in a certain organizational situation. This paper contributes to the information base upon which this can be decided by offering a tool to analyze the consequences of KM measures for knowledge sharing, i.e. for one of the most prominent goals found in many KM initiatives (Maier 2007, p. 476). Thus, the paper presents SimKnowledge, a multi agent-based simulation tool for studying behavior of knowledge workers and for analyzing effects of KM measures on individual and collective behavior. Central to the simulation model is a process model of knowledge sharing developed by the authors on the basis of an extensive interdisciplinary literature study grounded on psychological, social and economic theories in order to create a consistent explanation basis. After several steps to detail the model, reduce complexity, overcome incommensurable theoretical concepts and operationalize them, a simulation model arose which was implemented in a self-developed simulation system based on Java. This system was then used for a series of simulation studies analyzing the effects of KM measures the results of which are also reported in the paper.

MODEL OF KNOWLEDGE SHARING

According to semiotics, data is symbols with certain syntax and information is data with a certain semantic (Graefe 2002, p.134f). Knowledge can be seen as information in a context that allows people to interpret the information, sometimes it is also defined as information that enables people to act (Maier 2007, p.60f). Boer et al. identify three main interpretations of knowledge: knowledge as an object, knowledge in the minds of people, and knowledge as social practice of knowing (Boer et al. 2002, p.2). This paper follows interpretation two, thus only people can interpret information and act accordingly and a single person can create new knowledge by experimenting and inference. Nevertheless, having a shared understanding about their environment and expressing knowledge in written or oral form must be considered crucial for knowledge sharing. Following such a strict definition, knowledge becomes information as soon as it is explicated. Knowledge is reconstructed from information by interpretation in the context of the situation and the person's prior knowledge.

Knowledge sharing is defined as the process of one person (source) (1) deciding to share knowledge, (2) remembering a portion of knowledge, (3) explicating it to information on a medium, (4) directly or indirectly transferring it to another person (recipient), which (5) perceives the information and (6) interprets it in the given context so that the knowledge is re-constructed and integrated into the person's knowledge base. A final step is (7) the evaluation of the newly acquired knowledge by the recipient (Figure 1).

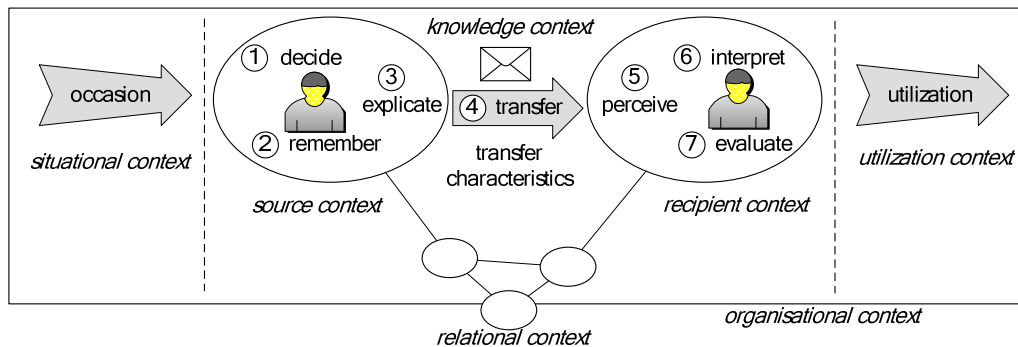


Figure 1: Process model of knowledge sharing

This definition is oriented at the model for knowledge sharing in (Maier 2007, p. 71) and the people-bound knowledge definition that leads to knowledge sharing as exchange of information to yield knowledge (Carley 2002, p.2). Compare also the similar, but less comprehensive models presented in (Meredith & Burstein 2000, p.5ff) and (Nissen 2002, p.253f). Both, antecedents and application are often not regarded as parts of the knowledge sharing process (Probst et al. 1998, Nissen, p.255f), but have to be considered in the knowledge sharing model used here.

An occasion gives the source the chance to initiate a knowledge sharing activity (see left hand side of Figure 1). In organizational contexts, utilization of acquired knowledge is important as knowledge is not accumulated for its own sake. This is reflected in the right-hand side of Figure 1. Source, recipient, channel, message and context are the five basic elements of knowledge transfer (Szulanski 1996) depicted in Figure 1 by persons (source and recipient), transfer arrow (channel), letter (message) as well as relational, organizational, situational and utilization context (context).

The decision to share knowledge in the given situation with a specific or anonymous recipient starts the process. The initiating occasion for knowledge sharing is an aspect seldom considered in literature. (Berends et al 2006) differentiate between the initiator of the knowledge sharing activity (push vs. pull). Identification of a partner for knowledge sharing is a complex process not included in the model. A number of 'knowledge transfer mechanisms' that can serve as occasions were collected by (Argote et al. 2000). These are personnel movement, training, observation, technology transfer, scientific publications and presentations and interactions with suppliers and customers (Balzert 2001, p.3).

Then the person remembers the required part of her knowledge. This remembering process has two side effects. It prohibits oblivion and may lead to a changed interpretation of the knowledge in the source because it cannot simply be retrieved, but has to be reconstructed (Cohen 1998, p.30ff). Explication converts knowledge bound to people into contextualized information directly perceivable by others. The more explicit the knowledge, the easier it can be explicated. It is difficult to explicate tacit or implicit knowledge, although it may still be explicable e.g., by demonstrating actions associated with it.

Transfer can happen directly, e.g., face-to-face, or indirectly, e.g., with the help of a document, personally or anonymously, synchronously or asynchronously. It can have various multiplicity (1-1, 1-n, m-n) and different media can be used (air, paper, electronic media). Thus, the model is based on a wide interpretation of knowledge sharing. Writing a book that another person reads afterwards in order to re-construct the author's knowledge is as much a knowledge sharing activity as answering a request for help from a colleague.

Interpretation is influenced by the recipient's context, especially her cognitive state. After interpretation, the recipient knows the meaning of the information perceived. The newly acquired knowledge is then evaluated by the recipient. Prior knowledge and new knowledge have to be aligned and potential contradictions have to be resolved. Applying the knowledge in a business process or project are common forms of knowledge utilization. Oral or written explication and further sharing of knowledge can also be seen as knowledge utilization.

RELATED WORK

Experiences from related simulation projects were reused extensively. The identified studies were drawn from multi-agent based simulation (MABS) systems simulating people at work on the one hand and non-agent simulations dealing with KM, software development or team work on the other hand. First MABS that fall into this category already appeared in the early 90s like (Mi, Scacchi 1990) modeling software development processes. Two studies analyze the impact of different team formations on completion of technical projects on time and budget (Martinez-Miranda & Aldea 2002, Moreno et al. 2003). Team members differ in cost, experience and willingness to collaborate. Projects are modeled as a number of work packages with given predecessor and successor. Both publications do not describe the results. A simulation study closely related to this paper is (Inuzuka 2003) which examines whether codification or personalization strategies result in higher economic success for an organization. Knowledge is modeled as discrete knowledge objects which can be learned from colleagues or a database. Newly acquired knowledge is documented to the database with a given probability. The results show that a personalization strategy achieves the same success with lower costs. The introduction of so-called gate keepers who act as knowledge broker increases the advantage over codification strategies.

In non-agent simulations, KM issues were also already investigated in the early 90s, e.g., Carley (1992) models organizational learning and personnel turnover in a Monte Carlo simulation. After 2000, the number of publications in this area has risen substantially. (Anjewierden et al. 2002) use a System Dynamics approach to simulate the relationship between knowledge and business processes on a business unit level. KM interventions affect how fast and efficiently knowledge processes can be executed and therefore how effectively the business units act. (Hanakawa et al. 2002) use micro simulation to model learning and productivity of a single software developer. Knowledge is modeled in interdependent areas that are more or less important in different development projects. Activities consist of a sequence of subtasks that require different knowledge. (Nissen, Levitt 2004) simulate knowledge flows between software developers in a team. They build on the models of project work developed in several years of research and implemented in a tool called Virtual Design Team. There is no description of results regarding knowledge flows, but the tool showed in several studies in aircraft and vehicle construction that it is able to forecast bottlenecks and project success quite well. Finally, (Elliman et al. 2005) simulate knowledge-intensive work in a lawyer's office using discrete event simulation. They explicitly consider priority of tasks and interruptions.

SIMULATION MODEL

The simulation model is based on the knowledge sharing model presented above and implements an agent architecture derived from InterRaP (Müller 1996), which allows of modeling the micro level while analyzing effects on the macro-(organization) and meso level (teams). It uses a utility function to determine best actions for selected goals and models internal states according to the PECS architecture (Physis, Emotion, Cognition and Status, Urban & Schmidt 2001). The software manufacturer is modeled as a project-based organization led by a CEO who gets job offers from potential customers, accepts them based on free capacities and delegates the job to a project manager. The latter searches for available developers with skills required for completing the work packages in the project. Existing skills of developers in contrast to skills required by their work package determine whether and how fast it can be completed. If developers lack knowledge to complete a work package, they attempt to learn from a colleague or documented knowledge. Company success is determined by earnings through completed projects minus personnel cost. Overall knowledge of employees is used as control variable. Figure 2 visualizes the primary influence relationships between main concepts used in the operationalized simulation model which are explained in the following.

External factors for the simulation model are potential customers that offer projects and competitors that compete not only regarding project offers, but also regarding potential employees and therefore knowledge on the labor market. Finally, there are external knowledge sources available that can be used to learn from. It is not distinguished in the current simulation model between external trainers, books and information available on the Internet. Competitors are modeled only implicitly by adjusting the rate of project offers and resources available on the labor market. The more projects, the fewer employees are available for completing them. More qualified and competent people on the labor market lead to overall decreasing salaries, but increasing knowledge in the company when hired. Learning increases the employee knowledge and knowledge furthers learning. Time allocated for learning cannot be spent on project work which results in a negative relationship to project progress. However, there is a positive relation via employee knowledge. KM instruments require investments and therefore cause training cost. Earnings are calculated by turnover, increased by progressing or completed projects, minus personnel cost, related to employee salaries and training cost which reflects the focus on knowledge handling by employees used in this paper. Knowledge total is calculated as sum of all employees' knowledge.

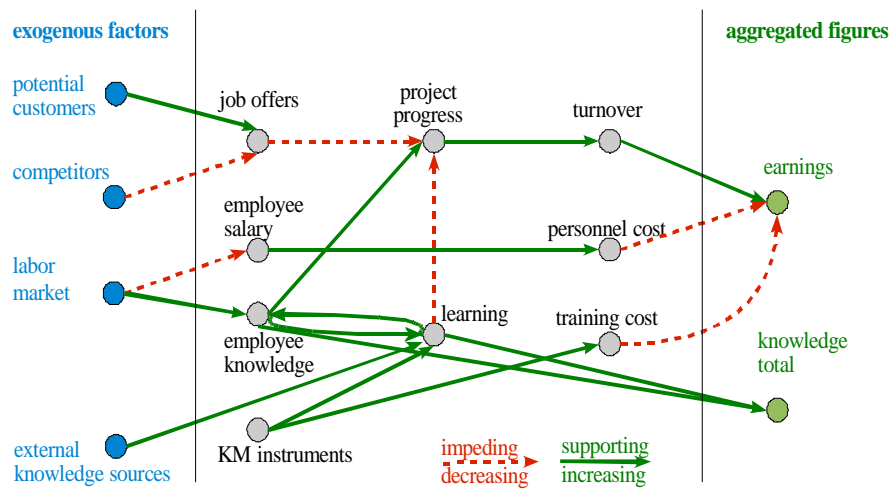


Figure 2: Model of influencing factors

A KM instrument is a collection of organizational, human resources and ICT measures that are aligned, clearly defined and can be deployed purposefully in an intervention into an organizational knowledge base in order to achieve knowledge-related goals. Applying a KM instrument as part of a KM initiative should affect learning positively as depicted in Figure 2. The main purpose of this simulation model is to study the impact of applying KM instruments to knowledge sharing and in turn business performance. There is a large number of KM instruments, e.g., case debriefings, communities, competence management, experience management, expert advice, good/best practices, idea and proposal management, knowledge maps, lessons learned or semantic content management. These can be structured according to the two primary KM strategies (Hansen et al. 1999) into codification or document-oriented KM instruments, e.g., experience management, versus personalization or human-oriented KM instruments, e.g., competence management.

Accepted projects compete for knowledgeable employees that can complete them. The project manager impersonates this by searching for project members to complete work packages. The employee's knowledge increases by working on work packages that are demanding (not too hard and not too easy) or learning from knowledge sources. Furthermore, an employee can learn from colleagues by asking about certain topics. KM instruments can influence learning in all three areas. The salary of an employee is constant over one year and can increase then, if her knowledge has increased over the year and she has a more than average rate of completed work packages. In the first models, probability of increasing salary was set to zero for reasons of simplicity. Work packages progress, if the skill of the employee is 0.5 less than the skill required by the work package or more (on a zero to five skill scale). A skill present equal to skill required means a progress factor of one. For skills present less than required the progress factor decreases linearly, for higher skills it increases.

IMPLEMENTATION

The model was implemented by one of the authors using Java 1.5 in a self-developed multi agent-based simulation system. Each agent is modeled as a separate thread. A static main class SimulationEngine handles creation, parameterization and management of the agent threads. Every round consists of perceiving senses from the world (audio and visual), calculating the next actions and executing them. Before every round, sequence of agent activation is mixed so that no agent has the advantage of being first in line. Agents are derived from a common base class and are implemented as three specializations CEO, project manager and (knowledge) worker, with the latter one being the most comprehensive one. Agent actions are implemented as inner classes with a common interface, so that they have full access to the agent's inner states. Selecting an action is done by calculating utility values of possible actions using reflection to instantiate the action classes.

CALIBRATION AND VALIDATION

Parameters were chosen in real life units wherever possible in order to ease parameterization and later on interpretation of results. Therefore, salaries are stated in Euros and skills are modeled as continuous variables on a zero to five scale. (Lethbridge 1999) served as the main basis for identifying relevant skills and initial values. One of the authors conducted an empirical study at a German software company with around fifty employees which came to similar results and served as a second basis for validation. Salary information came from German studies conducted by computer magazines (c't 2002-2006)

and a labor union (IG Metall). Dice data¹ from 2003 and 2006 were used to compare the German results with US American data.

In all studies, salaries were increasing slightly over the years, and years of job experience led to higher salaries up to a certain degree (up to 8 years in the Dice studies, later on more experience does not lead to higher salaries). Average salaries were around 52t€ (thousand Euros) in Germany for software engineers II (with a few years of job experience), 40t€ for beginners and 60t€ for professionals with several years of experience. Project managers got between 67 and 71 t€ US data was in similar regions with 53t\$ for beginners to 76t\$ for employees with 8 years and more of experience.

Employees in our own empirical study were between 20 and 35 years old and had an average of 3.5 years of job experience of which they were 2.8 years with the company. They participated in 5.8 projects in these 2.8 years with an average time to completion of 10 months. Thus, only few projects were processed in parallel. Most employees worked in small teams with two to five members (avg. 3.3). In the simulation model, three software developers and one project manager were used for simplicity reasons.

In order to calibrate the simulation, the model was extended step by step from one with only project work without learning, to a more sophisticated one, that allows learning by doing, learning from documented knowledge and learning from colleagues as well as forgetting knowledge that was not used for some time, quit processing of a work package if no progress is made for some time and meta-knowledge about project members that helps for team selection and finding colleagues to learn from.

Sensitivity analyses were conducted in order to find reasonable values for parameters that could not be determined empirically. The atomic time interval for the round-based simulation was set to 15 minutes, based on the actions that should be conducted within a round (e.g., ask colleague for help). The number of knowledge areas or skills was first deduced from literature (16 skill areas found in Lethbridge 1999) and then tested in the simulation model. Best results for a simple and comprehensible model were achieved with four knowledge areas altogether and one skill needed per work package. Learning rate was tested from a theoretical minimum based on literature stating that experts need 10 years with 2-4 hours a day (equals 0.00018) up to 0.02 and fixed at 0.01. Forgetting rate was tested with constant rates of 0.0004 up to 0.0010 and adaptive rates of a 15,000th up to a 35,000th per round the skill was not used and set to a 25,000th with 25 days delay before forgetting.

Skill values after 3 years in the simulation were close to those found in (Lethbridge 1999) and our own study for maximum and average skills (3.7 vs. 3.8 and 2.4 vs. 2.4), but showed higher values for minimum skills (1.2 vs. 0.8) and less deviation than empirical data (0.2-0.3 vs. 0.7-0.8). The lower minimal values may result from a higher number of skill areas in reality than in the simulation (only 4). Standard deviation is partly influenced by the different minimum values. Turnover and number of completed projects is higher in the simulation than in our empirical study (4 Mio € vs. 2.2Mio € and 22.6 projects vs. 6.3). This is due to good order situation (always enough project offers) and shorter project runtimes (4 vs. 10 months).

SIMULATION STUDIES AND RESULTS

Two simulation studies were conducted that analyze the influences of one human- and one document-oriented KM measure, i.e. skill management systems (competence management) and documentation of learning experiences (experience management). All experiments were conducted with 49, 77 and 101 agents² and a 3 years time frame and repeated 50 times each.

In the skill management study, project managers could use a skill management system (SkMS) to find developers with the required skills, whereas in the base scenario, they had to learn about developers' skills from personal contacts and acquired team members on that basis which soon is outdated, since developers constantly learn and forget. In a third experiment, the SkMS was also available for developers in order to find a suitable colleague from whom to learn.

Hypothesis 1: Usage of SkMSs decreases time for completing work packages.

This hypothesis is based on the assumption that the assignment of developers to work packages leads to a better mapping of actual and required skill. It was found that SkMSs do not automatically result in quicker completion of work packages (see Table 1), but its success largely depends on the strategy for assigning work packages to workers. If work packages are assigned randomly to workers whose skills differ least from the required ones, the organization's profit does not increase significantly. However, learning from colleagues ($\alpha=0,001$, $t=0,00035$ bis $0,00065$) and during project work rises significantly ($\alpha=0,001$, $t=0,0000$ bis $0,00057$). Thus, Hypothesis 1 was rejected.

¹ http://dice.salary.com/salarywizard/layoutscripts/swzl_titleselect.asp

² Uneven numbers result from the design with 1 CEO, X project managers and 3*X software developers

Agents	KM instrument	Earnings	Turnover	Skill max	Skill min	Skill avg	Skill total	WP duration
49	none	4.300 Mio €	20.889 Mio €	3.66	1.29	2.38	428.86	17.6
	SkMgmt f. PMs	4.217 Mio €	20.806 Mio €	3.72	1.17	2.33	419.7	18.18
	SkMgmt f. all	4.109 Mio €	20.698 Mio €	3.71	1.19	2.35	423.6	18.26
77	none	5.080 Mio €	31.346 Mio €	3.68	1.56	2.64	751.11	18.5
	SkMgmt f. PMs	4.607 Mio €	30.873 Mio €	3.64	1.51	2.57	732.2	18.88
	SkMgmt f. all	4.476 Mio €	30.742 Mio €	3.63	1.52	2.57	730.6	19.01
101	none	9.170 Mio €	43.730 Mio €	3.71	1.22	2.39	896.40	17.7
	SkMgmt f. PMs	9.105 Mio €	43.665 Mio €	3.67	1.16	2.30	863.2	18.31
	SkMgmt f. all	8.859 Mio €	43.419 Mio €	3.63	1.11	2.30	861.2	18.33

Table 1: Detailed results for skill management study

For experience documentation, developers could spend time on documenting their knowledge and making it available on the Intranet, so that other developers can learn from that. Initially, the maximum skill value that can be learned from documented knowledge was set to 2.5. This value increases with more agents documenting their knowledge presuming their knowledge is higher than the one already present in the Intranet. It is argued that developers are able to express all relevant knowledge since the domain is highly explicit, whereas in other domains this might not be the case. Two experiments were compared with the base scenario (no experience documentation). In the personal experience documentation experiment, every developer documented his own knowledge. In the project debriefing scenario, all team members meet at the end of a project, talk about their experiences and jointly document them.

Hypothesis 2: Usage of experience documentation increases average knowledge of employees.

Results show that systematic documentation of learning experiences leads to a significant rise in the average knowledge of employees ($\alpha=0,01$, $t=0,00940$ for 49 and $t=0,00400$ for 77 agents). Thus, Hypothesis 2 was supported, although no significant rise was found for 101 agents and personal experience documentation. Especially interesting is that documentation on a personal level leads to employees becoming more specialized in a single knowledge area, whereas documentation on a team level fosters generalists with medium to high knowledge in several knowledge areas (see Table 2). This is due to the fact that learning during project debriefings has a greater impact than learning from documented knowledge.

Agents	KM instrument	Earnings	Turnover	Skill max	Skill min	Skill avg	Skill total	WP duration
49	none	4.251 Mio €	20.711 Mio €	3.71	1.18	2.24	402.86	18.8
	Personal documentation	5.165 Mio €	21.624 Mio €	4.76	1.00	2.35	423.07	16.39
	Project reviews	4.876 Mio €	21.336 Mio €	4.36	1.29	2.73	411.79	17.01
77	none	4.654 Mio €	30.790 Mio €	3.67	1.47	2.44	695.19	19.3
	Personal documentation	6.066 Mio €	32.202 Mio €	4.89	1.04	2.59	737.06	16.78
	Project reviews	5.045 Mio €	31.181 Mio €	4.47	1.54	3.17	633.45	18.07
101	none	8.392 Mio €	42.823 Mio €	3.70	1.07	2.29	859.74	18.3
	Personal documentation	12.244 Mio €	46.675 Mio €	4.72	1.21	2.29	848.86	15.82
	Project reviews	9.868 Mio €	44.299 Mio €	4.49	1.21	2.83	861.08	17.30

Table 2: Detailed results for experience documentation

LIMITATIONS AND FUTURE DEVELOPMENTS

A dependency on the number of employees was not found which is surprising in both studies, since more employees lead both to fewer chances of knowing about the skills of all colleagues and to more people contributing to the experience base. This may be due to the fact that already with 49 agents the social network is highly concentrated on the local neighborhood

and agents only have an average of 2.7 contacts which is significantly less than the 5.5 contacts average found in our empirical study. This leads to the interpretation that the simulation model not yet captures the complex social structure that is present in real organizations. The often cited coffee talks that foster knowledge sharing do not happen in the simulation. In order to extend the model in that direction, the atomic time interval presumably has to be shortened, since otherwise going to the canteen, talking and going back would take at least four rounds, i.e. one hour.

In order to create more interactions between developers, dependencies could be introduced between work packages in form of project-specific knowledge, e.g., software architecture or data model of the application in development, so that employees need to learn from each other. This would require developers to work on more than one project, because otherwise waiting time rises too much. Parallel work on several projects would also be required to rise the number of skill areas to the value in literature without forgetting too much. Another possible extension is to turn knowledge areas into dynamic ones, so that experts in one area have still some new knowledge left to learn after a while. An easy implementation would be to raise the upper limit over time. It seems more realistic, though, to model knowledge as discreet variable, e.g., a bit string, and introduce additional bits. Finally, errors during coding could be introduced so that differences between experts and novices could be stressed.

CONCLUSION

This paper presented results of applying a multi-agent based simulation tool for two KM instruments, skill management and experience management which led the authors to the conclusion that multi agent-based simulations are well-suited not only to research knowledge sharing in organizations, but also to raise awareness of deciders for contextual factors needed to increase probability of a successful KM initiative. Keeping in mind both the high numbers of successful and unsuccessful KM interventions reported in the literature, this tool should help to transfer knowledge about success factors and barriers of applying KM instruments into a form that can be easily deployed by theorists and practitioners alike.

REFERENCES

1. Alavi, M., Leidner, D. E. (2001): Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues, *Management Information Systems Quarterly*, 25, 1, 107-136.
2. Anjewierden, A., Shostak, I., de Hoog, R. (2002): KMsim - A Meta-Modelling Approach and Environment for Creating Process-Oriented Knowledge Management Simulations, *13th European Conference on Knowledge Acquisition, Management and Modelling (EKAW)*, Sigüenza, 65-79.
3. Argote, L., et al. (2000): Knowledge Transfer in Organizations - Learning from the Experience of Others. *Organizational Behavior and Human Decision Processes*, 2000. 82, 1, 1-8.
4. Balzert, H. (2001): *Lehrbuch der Software-Technik*, 2 ed., Spektrum Akad. Verl., Heidelberg.
5. Berends, H., van der Bij, H., Debackere, K. & Weggeman, M. (2006): Knowledge sharing mechanisms in industrial research. *R&D Management*, 36(1), January 2006, Blackwell Publishers
6. Bishop, J., Bouchlaghem, D., Glass, J., Matsumoto, I.: Ensuring the effectiveness of a knowledge management initiative, in: *Journal of Knowledge Management*, Vol. 12, No. 4, 2008, 16-29
7. Boer, N.-I., Berends, H. (2003): The relational dimension of knowledge sharing – an empirical study of an industrial research group, *4th European Conference on Organizational Knowledge, Learning and Capabilities (OKLC 2003)*, Barcelona, Spain.
8. Boer, N.-I., P.J. van Baalen, and K. Kumar (2002): An Activity Theory Approach for Studying the Situatedness of Knowledge Sharing. *35th Hawaii International Conference on System Sciences (HICSS 2002)*. 2002. Big Island, Hawaii.
9. Carley, K. M. (1992): Organizational learning and personnel turnover, *Organization Science*, 3, 1, 20-46.
10. Carley, K.M. (2002): Information Technology and Knowledge Distribution in C3I teams. *Command and Control Research and Technology Symposium*. 2002. Naval Postgraduate School, Monterey, CA.
11. Cohen, D. (1998): Toward a Knowledge Context - Report on the First Annual U.C. Berkeley Forum on Knowledge and the Firm. *California Management Review*, 1998. 40, 3, 22-39.
12. Davenport, T. H., Probst, G. J. B. (2002): *Knowledge Management Case Book. Best Practises*, 2nd ed., Erlangen.
13. Drucker, P. F. (1994). The age of social transformation. *The Atlantic Monthly*, 274, 5, 53-80.
14. Elliman, T., Eatock, J., Spencer, N. (2005): Modelling knowledge worker behaviour in business process studies, *Journal of Enterprise Information Management*, 18, 1, 79-94.
15. Graefe, G. (2003): Incredible Information on the Internet - Biased Information Provision and a Lack of Credibility as a

- Cause of Insufficient Information Quality, *8th International Conference on Information Quality (ICIQ 2003)*, Cambridge, MA.
16. Hanakawa, N., Matsumoto, K.-I., Torii, K. (2002): A Knowledge-Based Software Process Simulation Model, *Annals of Software Engineering*, 14, S. 383–406.
 17. Hansen, M. T., Nohria, N., Tierney, T. (1999): What's Your Strategy for Managing Knowledge?, *Harvard Business Review*, 77, 3-4, 1999, 106-116.
 18. Inuzuka, A. (2003): How to Share Knowledge Effectively?, T. Terano, H. Deguchi, K. Takadama (Hrsg.): Meeting the Challenge of Social Problems via Agent-Based Simulation, Springer, Tokio et al., 123-137
 19. Lang, J., Torre, L. v. d., Weydert, E. (2000): Utilitarian Desires, *Autonomous Agents and Multi-Agent Systems* 5(3), September 2002, 329-363
 20. Lethbridge, T. C. (1999): The Relevance of Education to Software Practitioners – Data from the 1998 Survey, Technical Report TR-99-06, School of Information Technology and Engineering, University of Ottawa.
 21. Maier, R. (2007): Knowledge Management Systems - Information and Communication Technologies for Knowledge Management. 3 ed., Berlin: Springer.
 22. Maier, R., Hädrich, T. & Peinl, R. (2009). Enterprise knowledge infrastructures. 2 ed., Berlin, Springer.
 23. Martinez-Miranda, J., Aldea, A. (2002): A Social Agent Model to Simulate Human Behaviour in Teamwork, *3rd Workshop on Agent-Based Simulation*, Passau, 18-23
 24. Meredith, R. and F. Burstein (2000): Getting the Message across with Communicative Knowledge Management. *Australian Conference on Knowledge Management and Intelligent Decision Support (ACKMIDS)*. 2000. Monash University, Australia.
 25. Mi, P., Scacchi, W. (1990): A Knowledge-Based Environment for Modeling and Simulating Software Engineering Processes, *IEEE Transactions on Knowledge and Data Engineering* 2, 3, 283-294
 26. Moreno, A., Valls, A., Marín, M. (2003): Multi-agent simulation of work teams, *3rd International/Central and Eastern European Conference on Multi-Agent Systems*, CEEMAS, Prague, Czech Republic, 281-291.
 27. Müller, J. P. (1996): The design of intelligent agents - a layered approach, Springer, Berlin.
 28. Nissen, M.E. (2002): An extended model of knowledge-flow dynamics. *Communications of the Association for Information Systems*, 2002, 8 (online), 251-266.
 29. Nissen, M. E., Levitt, R. E. (2004): Agent-Based Modeling of Knowledge Flows: Illustration from the Domain of Information Systems Design, *37th Hawaii International Conference on System Sciences (HICSS 2004)*, Big Island, Hawaii.
 30. Szulanski, G. (2000): The process of knowledge transfer - a diachronic analysis of stickiness, in: *Organizational Behavior and Human Decision Processes*, 82(1), S. 9-27
 31. Urban, C., Schmidt, B. (2001): PECS – Agent-Based Modelling of Human Behaviour, Emotional and Intelligent II - The Tangled Knot of Social Cognition, *AAAI Fall Symposium 2001*, North Falmouth, MA.
 32. Wolff, E. N. (2005). The growth of information workers. *Communications of the ACM*, 48, 10, 37-42.