Accelerating the Sharing of Knowledge in Order To Speed Up the Process of Enlarging Software Development Teams - a Practical Example

Razvan Bologa
Ana Ramona Lupu
Computer Science Department
Academy of Economic Studies
Calea Dorobanti 15-17, Bucharest
Romania
http://www.ase.ro

Abstract: In today's fast growing IT industry many companies are constantly experiencing shortages of software developers, a key resource in the computer-based world that we live in. This frustrating reality comes from the fact that training a competitive software developer is a long process that may take years which is incompatible with the much faster ups and downs of the software industry. As a consequence many managers from software development and consulting companies are attempting to accelerate the process of training software developers. The current article presents an example of knowledge sharing practice in which young graduates have been trained as ABAP programmers with the help of an accelerated knowledge sharing method that has enabled them to reach in only 4-6 months a level of expertise that has made them acceptable for top system integrators. Traditionally, for reaching such a level there would have been necessary almost 2 years of training on the job. The article focuses on the architecture of the knowledge sharing system and the roles of the persons involved. A simplified mathematical model of the success rate is also described. As a conclusion the article demonstrates that the time needed to acquire expertise as an ABAP programmer can be reduced from 2 years to 4-6 months with the help of knowledge sharing methods.

Keywords: knowledge sharing, abap, knowledge management, training

1. Introduction

The process of knowledge sharing has been studied since the ancient Greeks have established the famous Athenian School. Along the centuries there have been countless investigations of the process of knowledge creation and sharing. The perspective of those who studied the concept knowledge has always depended on the needs of the societies they lived in. Plato, Aristotle, Descartes, Leibniz, Locke and Kant have approached knowledge philosophically. Their intention was to investigate knowledge in a subjective way and to integrate it in a view of the world. The psychologists and the scientists of the 19th and 20th century, such as Wilhelm Wundt, John B. Watson and George Miller, focused on the practical goal of using knowledge in an attempt to optimize the industrial world they were living in. They explored knowledge from an interdisciplinary perspective[1].

Neither the philosophers nor the psychologists and the scientists had a major interest in accelerating the process of knowledge creation and sharing. This happened only after computers started to irreversibly change our world a few decades ago. Knowledge is becoming nowadays the main source of competitive advantages and companies are chasing it. Acquiring valuable knowledge sooner then competitors is a key advantage for a company. This is why managers and shareholders are starting to engage in a fierce battle for having knowledge first which leads to the need to optimize and accelerate the process of knowledge creation and sharing. Over the last decade it has been proven that organizations can use the sharing process in order to increase their knowledge base and competitiveness (see [2], [3], [4] and [5]).

The following parts of this article will present a successful experiment in which knowledge sharing has been used to increase the competitiveness of a consulting team specialized on SAP technologies.

For those involved in the multi-billion business of SAP consulting it is a known problem that it's almost impossible to have the right number of experts when needed. The reason for this is the fact that among the project managers of SAP projects there is the idea that a consultant must have at least 2-3 years of experience with SAP in order to have the right expertise. Since SAP business is constantly expanding and changing the consultants are either not enough or lacking the needed specialization. Training new consultants is a difficult and slow process as junior SAP experts and not welcome in large numbers by the demanding companies that choose to install SAP. This is a really bootstrapping situation that often resulted in large gaps between the offer and the demand of labor force that cause problems both to the companies and to consultants. ABAP is the programming language in which most of SAP was written and ABAP programmers, like other SAP experts, are scarce, difficult to train and badly needed.

At beginning of 2004 an Italian partner of IBM(the world's largest SAP system integrator), has approached the Computer Science Department of Romania's largest university with the bold idea of setting up a knowledge sharing system that would turn young graduates into internationally competitive ABAP programmers in a matter of months. The idea was greeted with skepticism in the academic and professional SAP communities from Italy and Romania as it was generally considered that a programmer in ABAP must be trained on the job for 2-3 years on various projects in order to have a reliable level of expertise. However, in spite of the initial skepticism, the managers from IBM remained open to the experiment as they were experiencing painful shortages of ABAP programmers.

2. Problem Formulation

Our aim is to find a way to accelerate the sharing of knowledge between the members of an existing ABAP team of programmers and a number of trainees so that the trainees reach a good level of expertise in 4-6 months instead of the 2-3 years required by the traditional training on the job.

ABAP is a simple programming language that has been in use since the '70s. The programming language in itself is not difficult to learn. The main problem in training ABAP programmers comes from the fact that there are thousands of SAP tables containing data that an ABAP programmer must manipulate in order to successfully complete a task. For manipulating this data a programmer must be familiar with the structure of the tables he or she must use.

Having deep knowledge of the structure and the data stored in these tables is the key to being a successful ABAP programmer. Acquiring such knowledge is a learning by doing process. This knowledge is episodic. Episodic knowledge is based on experimental information, or episodes [8]. It is difficult to recall and to verbalize. An ABAP programmer needs a lot of hands-on experience in order to be able to successfully create forms and user exits based on the huge collection of tables of SAP. Apart from the tables, there is also the fact that an ABAP programmer often has to customize code that is supplied with the SAP system and a programmer can get familiar with this code only by "playing" with it.

The challenge is to identify a knowledge sharing method that would accelerate the sharing of ABAP knowledge between expert and senior programmers and the trainees. Unlike conventional knowledge management system, the current system is intended to enable only the sharing of knowledge and does not aim to capture existing tacit knowledge and place it in a central repository. The objective of the knowledge management system is only to accelerate the process of knowledge sharing.

3. Problem Solution

3.1. Identifying the Knowledge to be Shared

The ABAP knowledge to be shared can be divided in two main categories: tacit and explicit. The two categories of knowledge are well known to the knowledge management community(see [6] and [7]) and represent an efficient way of dealing with human knowledge when designing knowledge management systems.

Explicit knowledge is knowledge that can be found in books, reports, training materials, reviews, documentation and discussion forums. It is easy for an ABAP programmer to absorb explicit knowledge. However, it should be noted that the explicit knowledge about ABAP can be acquired only if certain perquisites are met. The main perquisite is to have a well established algorithmic thinking and to be familiar with the basic programming concepts and database technologies. ABAP makes extensive use both of databases and of programming logic and it is impossible for someone who does not have such knowledge to start the process of studying ABAP.

Tacit knowledge on the other hand is the knowledge embedded in the human mind through practical experience. It includes intuitions and other elements that are hard to verbalize but are extremely useful. This is the most important knowledge to be shared.

In the case of ABAP, the explicit knowledge is very well organized in courses provided by SAP AG. For a programmer it is only a matter of time and effort to acquire this knowledge which is also considerably less complicated than in the case of other programming languages.

The difficult task of training an ABAP programmer comes from the fact that there is a lot of vital tacit knowledge to be shared. Such knowledge can be found only in minds of expert programmers. The tacit knowledge of an expert ABAP programmer allows him or her to deal with the various tasks in a rather automated way. Thanks to tacit knowledge, an expert programmer would know where to look and what to do in an instinctive manner.

Turning this tacit knowledge into explicit knowledge has been a main area of interest in the ABAP community ever since SAP started to become a major company. Nowadays, after decades of endeavor, there is a limit that has been reached and almost all the knowledge that could have been captured has already been obtained. Unfortunately, the extensive and well organized ABAP knowledge base that has been produced by the SAP community is not good enough for training a programmer quickly. After consulting this knowledge base a programmer would still have to spend 2-3 years of training o the job until reaching a level of expertise that would make him acceptable in top consulting companies.

Tacit to tacit communication is a key activity in the process of sharing ABAP knowledge. Such knowledge sharing is mostly about socialization and often produces no explicit knowledge. Experience is generally shared in face-to-face meetings and technology plays a marginal role. Conversion of knowledge between tacit and explicit forms is presented in the figure bellow[9]:

Tacit to Tacit *Socialization*	**Tacit to Explicit** E*xternalization*
Explicit to Tacit *Internalization*	**Explicit to Explicit** *Communication*

Fig. 1 Conversion of knowledge between tacit and explicit forms

As discussed above, the conversion of tacit to explicit knowledge in ABAP is unlikely to take place. All other three forms of knowledge conversion would have to be addressed by the knowledge sharing system with a particular attention paid to the tacit to tacit conversion.

3.2. Designing the knowledge sharing system

The basic idea in designing the knowledge sharing system iwas to insist on the sharing of tacit knowledge between expert ABAP programmers and the trainees. Such a knowledge sharing must be based on examples. Sharing explicit knowledge was also important. However, explicit ABAP knowledge was easily shared using conventional classroom training methods based on the well organized courses coming from SAP AG.

For knowledge to be shared, it must first be deployed in a format acceptable to the user. We have seen in the previous section that most of the tacit knowledge in ABAP has an episodic nature. In the process of conveying episodic knowledge, the expert explains by examples, or scenarios [8]. The trainee must be exposed to a large number of such examples.

The examples that were used for training the ABAP programmers were older tasks that expert programmers had to solve in the past. Basically, the trainees got a set of real specifications from older projects and were asked to solve the task as of it was a real one. They were assisted by the expert programmers and were offered the solution code as a reference.

A set of scenarios based on real tasks works more efficiently than artificially constructed examples as they allow the expert programmers who act as trainers to recall easily the episodic tacit knowledge they have to pass to the trainees.

The knowledge sharing system had two components as indicated in fig. 2: the explicit knowledge sharing component and the tacit knowledge sharing component.

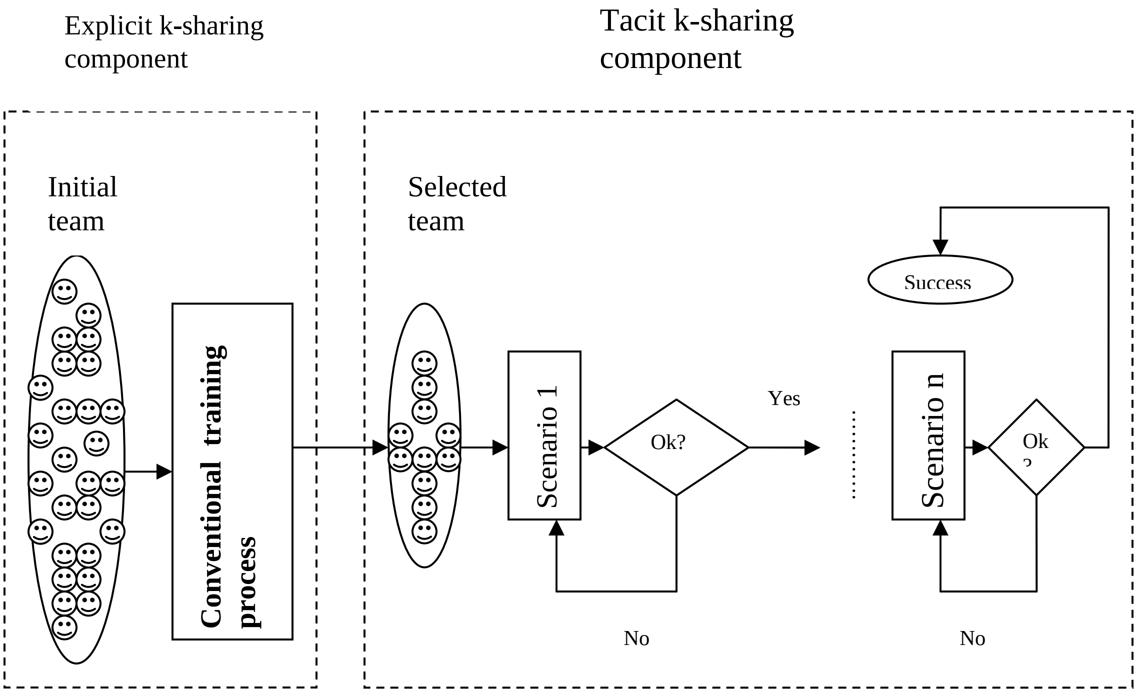
Explicit k-sharing component
Tacit k-sharing component
Initial team
Conventional training process
Selected team
Scenario 1
Ok?
Yes
No
Success
Scenario n
Ok ?
No

Fig. 2 The knowledge sharing system

The initial team was made of the trainees who entered the process of training. They were mostly graduates or soon to graduate students that were interested in learning to program in ABAP. The students were selected based on interest and they were checked to make sure that they have a good basic training in computer science. Good programming skills and knowledge of database systems were a perquisite of undergoing a standard ABAP training process.

This team went through a conventional training based on the standard ABAP SAP documentation. The duration of the courses was the standard duration of SAP (about 5 weeks). During the conventional training students were gathered in a classroom where they were explained the basic aspects on ABAP and were required to solve the standard exercises provided by SAP.

The initial team was reduced to a smaller selected team after the conventional process of training. The selection was made based on a test they had to take. The number of selected students also depended on business factors such as the capacity of the consulting team to insert new ABAP programmers in its projects once they were trained.

After that, the members of the selected team went through a process of intensive training based on the scenarios. As explained above, the scenarios were older projects already completed by the consulting team. The older projects were an excellent training base because the expert programmers could easily recall the episodic knowledge they were supposed to transmit to the trainees.

In the process of building the scenarios it was very important to choose scenarios that had a suitable level of difficulty (smaller initially and higher later on) and also to cover all functional areas of ABAP. It was also important to make repetitions as trainees needed to encounter a situation at least twice in order to be able to address it.

A scenario took about 2 weeks to complete. The expert programmers and the trainees had to share the same room as socialization is a key factor in transferring tacit knowledge. Socialization activities were also substantially encouraged with teambuilding activities that ranged from classical corporate parties to setting up to more elaborate parties.

Once the trainees completed the work demanded by the specifications of the scenario they asked the expert to evaluate their work. If the expert programmer considered that they the level of knowledge accumulated was not satisfactory he or she could ask the trainees to re-run the scenario. Such situations were very frequent. At the end of the scenario the expert made a final evaluation and based on that he or she decided whether or not the trainees had to re-run the current scenario or to pass to a new one. The duration of a complete training process was 4-6 months.

3.3. Estimating the size of the initial team

Answering the question: "How many persons should be in the initial team in order to make sure that we have enough in the selected team?" was an important issue that had to be addressed in order to make the system work. In the very first session the organizers had to take a guess but after that the decision could be based on previous data.

There were four sessions of ABAP training organized in Bucharest according to the principles described above starting with 2004. The size of the initial team and the selected team is presented in table 1 for all of the four cases.

Session#	Size of initial team	Size of selected team
1	50	15
2	38	10
3	36	10
4	80	26

Table 1. Team sizes for ABAP training

If we consider that the size of the selected team(ST) depends on the initial team(IT) according to the following formula:

IT=a ST, where a is a constant that can be estimated with the least squares method.

Applying the least square methods we determine a=3.24. This result tells us that based on the experimental results obtained the selected team is about 30% of the initial team.

3.4. Testing the system

The only way the efficiency of such a system could be tested was by sending the trainees to real projects once they had completed the process of knowledge sharing. Out of the 35 ABAP programmers trained in the first three sessions only two were rejected by the project managers of IBM. The rest of them were accepted in top projects all over Europe. In one case, nine programmers, out of which six had no other experience then the one obtained during knowledge sharing process described above, succeeded in coding a reference project of one of the world's largest tobacco companies.

4. Conclusion

We can conclude by stating that using scenarios based on previous projects will speed-up the process of training new ABAP programmers. Such scenarios work much better and faster because the expert programmers that act as trainers can easily recall their episodic knowledge. Socialization is also an important part of the process of knowledge sharing as social activities act as a catalyst in the process of sharing knowledge. By employing an accelerated knowledge sharing method an ABAP programmer can be trained in only 4-6 to a level that would take 2-3 years of traditional training on the job.

References:

[1] Thagard, P. . *Mind : Introduction to Cognitive Science.* Cambridge, MA: The MIT Press, 2005

[2] Davenport T. and Prusak L.. *Working Knowledge.* Harvard Business School Press, Cambridge, MA, 1998

[3] Andrews K.M. and Delahaye B. L. . *Influences on Knowledge Processes in Organizational Learning: the Psychosocial Filter.* Journal of Management Studies 37(6), 797-810, 2000

[4] Tsai W. *Knowledge Transfer in Intraorganizational Networks: Effects of Ntework Position and Absorptive Capacity on Business Unit Innovation and Performance.* Academy of management Journal 44(5), 996-1004, 2001

[5] Lin H.F *Impact of organizational support on organizational intention to facilitate knowledge sharing.* Knowledge Management Research & Practice 4/2006, 26-35, 2006

[6] Nonaka I and Takeuchi H. *The Knowledge-Creating Company.* Oxford University Press, 1995

[7] Leonard D. and Sensiper S. *The role of Tacit Knowledge in Group Innovation.* California Management Review 40(3), 112-132, 1998

[8] Awad M.E. and Ghaziri M. H. *Knowledge Management*, Pearson Education, New Jersey, 2004

[9] Marwick A. D. *Knowledge Management Technology*. IBM Systems Journal, vol. 40, no. 4/2001, 814-830, 2001