Journal of Computer Science 7 (2): 298-303, 2011
ISSN 1549-3636

An Approach from Knowledge Dust to Gems (Knowledge Management)

Praveen Kumar and Satish Chhokar
Department of Computer Science, Raj Kumar Goel Institute of Technology,
UP Technical University, Ghaziabad, India

Abstract: Problem statement: Most businesses rely on the fact that their employees possess relevant knowledge and that they can apply it to the task at hand. The problem is that this knowledge is not owned by the organization. It is owned and controlled by its employees. Maintaining an appropriate level of knowledge in the organization is a very important issue. It is, however, not an easy task for most organizations and it is particularly problematic for software organizations, which are human and knowledge intensive. Knowledge management is a relatively new area that has attempted to address these problems. **Conclusion/Recommendations:** This study introduces an approach called the Knowledge Dust to Gems approach. This approach addresses some of the issues with knowledge management by providing low-barrier mechanisms to “jump start” the experience base. This approach allows the experience base to become more useful more quickly than traditional approaches.

Key words: Knowledge management, knowledge dust, software organizations, Quality Improvement Paradigm (QIP), Experience Factory (EF), knowledge gems

INTRODUCTION

The business world tends to be more and more knowledge-oriented. While organizations in the past were more focused on expensive machines to produce their products, organizations today face a much more competitive environment that is highly based on knowledge. In order to be successful and to excel, businesses of today need to focus on getting the right knowledge at the right time. Most businesses rely on the fact that their employees possess relevant knowledge and that they can apply it to the task at hand. The problem is that this knowledge is not owned by the organization as such. The knowledge is owned and controlled by its employees. The following statements illustrate this situation:

- “Our knowledge is moving regularly”
- “It is not the case experience is moving out, there is inexperience also walking in the door”

Maintaining an appropriate level of knowledge in the organization is a very important issue. It is, however, not an easy task for most organizations and it is particularly problematic for software organizations, which are human and knowledge intensive. The employees are the main assets of software organizations. The software industry is characterized by frequent technology changes, which calls for a continuous stream of new knowledge. In addition, the turnover rate in software organizations is often high, which makes the problem of maintaining knowledge worse.

The problem of maintaining an appropriate level of knowledge consists of many different sub problems. Examples of these sub-problems are:

- When Experience moves out, there is loss of knowledge because new one will take time to reach up to that
- When new entrant comes, he start learns instead of supporting existing one because of lack of knowledge
- Trainer cannot devote a good plenty of time to new employee because trainers may be busy in other work load
- Location of knowledge is not known. A common problem is to locate knowledge needed to solve a particular problem. The knowledge is probably already present in many organizations; the problem is to identify where it is or who has it

Knowledge Management (KM) is a relatively new area that addresses business problems by capturing and sharing knowledge. There are many software packages on the market that help organizations with this task

Corresponding Author: Praveen Kumar, Department of Computer Science, Raj Kumar Goel Institute of Technology, UP Technical University, Ghaziabad, India

(Lindvall *et al*., 2001; Jabar *et al*., 2010). Knowledge management can be applied by any kind of organization that needs to manage its knowledge. The experience factory (Basili *et al*., 1994) is an example of a knowledge management approach especially designed for software organizations. The Experience Factory recognizes that organizations need to learn from their past experiences in order to deliver products faster, cheaper and with higher quality than before.

If knowledge management is the answer to the problems of maintaining knowledge, why aren't all organizations doing knowledge management?:

- The reason may be Knowledge management is hard
- Selling may be difficult

When management is approached to invest in knowledge management they will ask the critical questions:

- What is the return on the investment?
- What is the payoff time?

The problem is that knowledge management is sold to management mainly as a long-term investment, meaning that the organization invests now and harvests later. Even if there are early benefits, these are often not evident because they are hard to see and measure. This is mainly the same problem that the software process improvement activities face: The benefits from these activities are often intangible and it is hard to assess.

The critical question that individual employees will ask is: what do I get out of changing my work processes and doing additional work? Knowledge management is based on the contribution of the employees. It is built on the fact that knowledgeable employees capture and share their knowledge with less knowledgeable employees. When knowledge management is sold to individual employees the message they hear is: You invest now, someone else might harvest later. This answer is often not satisfactory enough to motivate them.

Because it takes time to get the benefits from knowledge management and because it is hard to measure the benefits, knowledge management can be seen as a risky activity. Knowledge management requires a relatively large investment and a fundamental commitment to change the organizational culture to a sharing one. The risk lies in the fact that it takes a long time to notice if the wrong approach was selected or to find out that another direction would have been more successful.

These risks manifest in many ways. A classical example is when the organization spends a long time developing an enhanced taxonomy before worrying about the content of the experience base. This can be compared to designing a library system for classifying books without having books to fill the library. When the organization realizes that content is needed it starts to author books for the electronic library. If it takes long to develop the taxonomy, it will take even longer to populate the experience base eventually, when the organization has spent all its efforts on authoring electronic books, the intended users have already lost interest because they perceive the experience base as limited in providing them the support they expected.

The knowledge dust and gems approach: Based on our experience we set out to define a less risky approach that would be more appealing to both management and individual employees. The new approach is influenced by the ideas of the Quality Improvement Paradigm (QIP) (McGarry, 1994) a model for process improvement in software organizations. QIP uses the notions of Continuous improvement and iterations as the main vehicle for planning, executing, evaluating and improving processes.

These concepts led us to define an approach that leads the organization to knowledge management gradually and improve step by step. In short, it allows the organization to invest less now and also harvest some now. It enables the organization to evaluate the approach and improve based on the results. When the organization is ready to advance, it can invest more and harvest more. This leads to a situation where individual workers see benefits much sooner and allows the overall direction of the initiative to be adjusted quicker.

Our main approach is the Experience Factory (EF), which establishes a learning organization. The EF approach is beneficial for software organizations that need to learn from their past experience. The EF is a sophisticated approach that satisfies an organization's long-term needs of sharing experience. We searched the literature for a complementary approach that would satisfy the short-term needs of an organization. The complementary approach we selected to base our new approach on was the AnswerGarden (Ackerman, 1990).

The AnswerGarden (AG) (Ackerman, 1990) approach was developed due to the needs of sharing knowledge in customer support organizations and helpdesks. Such organizations receive requests for help from customers and users of a system and the organization helps them overcome the problems encountered with that system. The nature of this business is highly repetitive; the same request for help occurs frequently and the same help strategy can be used many times:

- The first challenge help-desk organizations face is how to capture and share the knowledge so that known answers can be quickly dispatched to customers
- The second challenge is to establish a process that allows experts share their knowledge with each other and with novices in an efficient way

The AG approach lets employees store and organize questions and answers as they are received and answered by the organization. By storing questions and corresponding answers in a common repository, the knowledge can easily be spread throughout the organization. By making the repository open to customers, they can even try to serve themselves first, by searching for existing questions and answers. Only when they do not find an answer, they need to take one step further and contact the helpdesk. Examples of such approaches, including ones where customers help each other can be found on many companies' web sites.

Our approach, the Knowledge Dust to Gems, combines and makes use of benefits both from the AnswerGarden (which represents Knowledge Dust) and the Experience Factory (which represents Knowledge Gems). The AG serves short-term needs; it uses an ad-hoc methodology and, it enables the collection of fine-granular items that lead to organic growth. Organic growth is a desirable property of an experience base as it lets the experience base grow in areas where employees search for knowledge, i.e., there is a need and demand for knowledge. The EF serves long-term needs, it is based on a sophisticated analysis and synthesis methodology; it uses feedback loops and recognizes the need of a separate organization that is responsible for the analysis and synthesis-the EF group.

The new approach that we have implemented captures the knowledge dust that employees use and exchange on a daily basis and immediately, with minimal modifications, makes it available throughout the organization. This process is accomplished by creating a system that supports peer-to-peer activities; i.e., the employees of the organization help each other and fulfill the short-term return goals of a knowledge capturing and sharing approach. In parallel, the knowledge dust is analyzed and synthesized and transformed into knowledge Gems, which represent more sophisticated, refined and valuable knowledge items that take longer time to produce. This study is often complex and needs to be done by a separate organization: the EF group.

The new experience factory model: Figure 1 shows the Experience Factory (EF) (Basili *et al*., 2001) Model and its main feedback loop. The EF group collects data produced by the project organization. The data goes through an extensive analysis phase and is synthesized into higher levels of knowledge and packaged in the form of experience packages. The experience packages are stored in the experience base and are made available to the project organization mainly in the form of business support provided by the EF group.

The full feedback loop generates valuable experience packages, but due to the sophisticated analysis and synthesis processes, it takes a relatively long time to generate experience packages and make them available for the project organization. In order to deliver benefits from the EF sooner, we defined a way to speed up this feedback process. The new approach adds a new and shorter feedback loop to the EF, which can be seen in Fig. 2.

In the new approach the EF group defines a number of ways to let the project organization collect data about itself while it is conducting its core activities. This data, the dust, goes through a minimal analysis phase turning it into mini-gems.

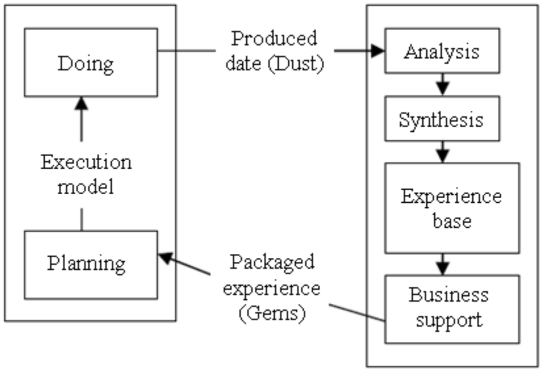

Fig. 1: The experience factory model

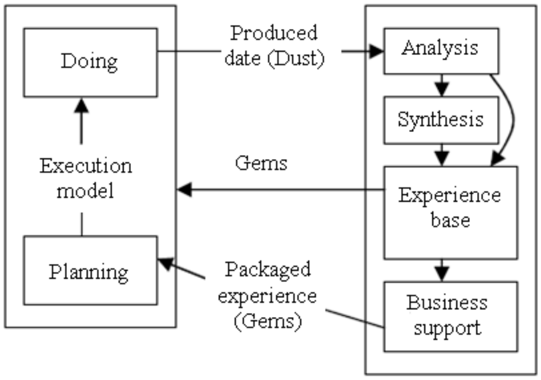

Fig. 2: The experience factory model with the new feedback loop

The mini-gems entirely bypass the synthesis phase and are stored in the experience base. The mini-Gems in the experience base are made available to the project organization almost immediately after collection. In this way, the organization receives benefits from the EF as soon as the dust collection process is established.

Knowledge dust: The main characteristic of knowledge dust is that individuals produce it during their core activities. Thus knowledge dust is a nice side effect produced during activities that have to be done anyway. This does not mean that it takes no time to produce the dust or that it is worthless in another context. The point is that by collecting the information that is produced already and applying it to a broader context, widely extends its utilization. In essence, collecting dust requires that the organization asks employees for a minor change in behavior. An example of such a small behavior change is to capture tacit knowledge while they are being exchanged between two employees, or by making explicit but private knowledge available to a larger group of employees. On other occasions, the organization must ask the employees for more information. An example is to ask for more information to be added to bug reports so that they can be more easily analyzed later. Knowledge Dust is much smaller and more fine-granular from a knowledge management perspective. Each dust particle may not convey a lot of information, but becomes useful when gathered, published and made searchable.

Knowledge gems: Knowledge Gems are larger sets of dust that are analyzed and evolved into higher levels of knowledge. Knowledge Gems often start out as smaller Gems that grow larger and larger over time. Cultivating knowledge dust into Gems is a continuous improvement process. As more dust is collected, more extensive analysis can be conducted, which results in more complete and better organized Gems. Cultivating knowledge Gems can be a relatively sophisticated activity that can take time. Therefore, it is conducted by the Experience Factory (Basili *et al*., 1994) group which specializes in analysis and synthesis techniques, such as qualitative and quantitative analysis, statistical methods and models based on empirical data. The Gems are fed back into the experience base for future use by both the project organization and the EF group. As the Gems are used by the organization, feedback about this usage is collected and analyzed and the Gems are improved based on this feedback.

Figure 3 shows the stages of the Knowledge Dust evolution until it turns into Knowledge Gems. The cultivation of dust to Gems is very inexpensive early in the process. The project organization provides the dust with a minimal effort. The EF group spends minimal amount of time on the analysis of dust, turning it into mini Gems and then making them available to the project organization.

More sophisticated Gems require more sophisticated analysis. This more sophisticated analysis that is done by the EF group is more costly and takes more time, but the result is also much more useful to the organization.

Cultivating dust into gems at a company: The knowledge dust to Gems approach has been applied in several different ways, for various different projects. This section describes some of these activities and method for cultivating Gems.

Many times, employees communicate over email when asking and answering questions. These questions and answers are stored in their FAQ system that can be used by all employees. The FAQ puts structure and makes public questions and answers exchanged by employees. Periodically, the EF group analyses the FAQs and identifies pieces that are rich enough to be turned into Gems. The related FAQs are put together in a more refined document, for instance, a process description that is stored in the experience base to be later used by the project organization.

When an incident occurs, employees are encouraged to record them in our Lessons Learned database. This database can be used by employees to search for such incidents and to apply the lessons learned to the task at hand. When the EF group realizes that a critical mass of lessons learned are collected on a certain topic, the EF group analyses and synthesizes the lessons learned into a best practices document that is stored into the experience base and is provided to the project organization.

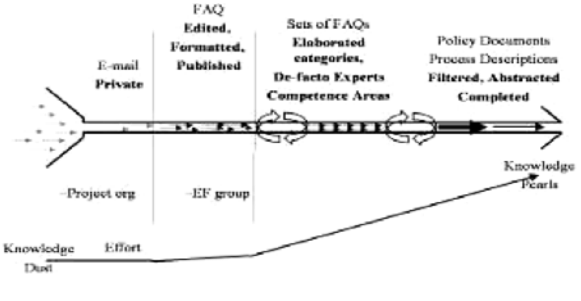

Fig. 3: Cultivating knowledge dust into knowledge

We may have a software package for documenting and tracking the defects during software development. This can be accessed by all employees and can help identify who are the experts on certain technologies. The S/w package may provide fields where developers and designers document their decisions and issues, thus keeping a history of the facts. Based on the reports generated by this, it may be possible to identify problem areas and proposed solutions. The EF group analyzes the bugs reported and establishes development practices that can be used by the project organization.

We may use chat software that for conducting guided discussions that are captured and analyzed. The chat logs and summaries can be used by the project organization immediately. Also, the EF group makes a detailed analysis about both the content of the discussion and the process to generate Best Practices and Lessons Learned documents that are stored in the experience base and can be used by the project organization.

One method that we are investigating for synthesizing knowledge dust into Gems involves the use of qualitative data analysis methods. Qualitative data is data in the form of text and pictures, as opposed to numbers or discrete categories and can be either objective or subjective. Qualitative analysis techniques are generally designed to find trends, commonalities and "stories" in textual data. Some of the more commonly used qualitative data collection techniques include interviewing and observation, while qualitative analysis techniques include the constant comparison method and cross-case analysis. The qualitative analysis method we have chosen to model our dust-to-Gems technique is the constant comparison method (first proposed by (Glaser and Strauss, 1967) and is described in more practical terms by (Miles and Huberman, 1994)). The constant comparison method was designed to generate grounded theory, in other words to reveal hypotheses (based on trends and patterns,) that are grounded in the qualitative data collected. The basic process is to first code a set of qualitative data (a textual narrative resulting from an interview and an observation) by assigning codes, or labels, to passages of text indicating the subject or theme of that passage. Then similarly coded passages are grouped together, in context, to reveal trends and patterns. A field memo is then written, describing the finding and referencing the data that supports it. Writing and refining field memos eventually lead to the formation of hypotheses. This process is repeated in parallel with continued data collection, thus refining and shaping the hypotheses as patterns in the data emerge.

Our process for synthesizing knowledge dust will be involved in examining a set of experience packages for similar, contradictory, or related experiences, resulting in a comprehensive "story" or description of a phenomenon or a higher-level statement that is supported by experience. For example, suppose the experience repository contains 10 project history reports on any projects (i.e., development projects in which commercial components were integrated into the final system). Suppose further that 6 of these cite problems working with "young" component vendors, e.g., the vendor going out of business, a vendor is being unable to properly support the product and low quality components. The vendors cited in these reports ranged in "age" from 6 months to 4 years. The remaining 4 projects, whose reports did not mention problems with vendors, worked exclusively with "older" vendors. Synthesizing the dust from this set of textual experience would result in a higher-level package that says, "Working with vendors who have been in business less than 4 years is problematic" and then would go on to describe the types of problems that typically arise, discuss various contextual issues of these problems and finally cite the original experience packages from which this information came. Such a higher-level package would be useful to future projects not only because it constitutes an experience gems, but also because it saves the future project personnel time in reading through all the previous project history reports (which they would be unlikely to do any way).

The proposed process for qualitative analysis of experience dust is as follows:

- A set of experience packages (suspected of containing important experience dust) is identified. In the example above, this would be the set of project history reports
- Then, a set of keywords is chosen that describe the topic or issue that needs to be investigated (in the above example, such keywords might be "vendor" or "problem")
- Then the text in the experience packages would be searched for occurrences of the keywords and the passages found would be coded accordingly
- The next step would be the intellectual task of grouping the coded passages and looking for trends, resulting in a
- "story", or a hypothesis, called a gems
- And finally, a new experience package would be created to summarize the newly created knowledge

CONCLUSION

Knowledge is increasingly becoming the main asset of many organizations. This is particularly true for software organizations. Knowledge Management is the proposed solution to the problem and is built on the capturing and sharing of employees' knowledge. Capturing and sharing knowledge is hard and risky. The time it takes to capture, organize, package and distribute knowledge is often to long for managers to be willing to invest in knowledge management. Employees hesitate to participate because they have to change their study processes and do additional work while their personal benefits are intangible and unclear.

We address these problems by defining a new approach called the Knowledge Dust to Gems approach. This approach adds the short-term-oriented features from the AnswerGarden (Ackerman, 1990) approach to the long-term and sophisticated features of the Experience Factory approach. The result is a new, shorter feedback loops that "jumpstarts" the creation and evolvement of Experience Factories in general and Experience Bases in particular. The Knowledge Dust to Gems approach is a low-barrier and less-risky approach that helps organizations quickly capture, share and use daily knowledge and evolve it into more complex knowledge over time. The approach gives the organization a starting point for building an Experience Factory and a Learning Organization. It helps change the corporate culture into a sharing one by making visible how sharing knowledge adds value sooner. The approach provides support for the project organization as well as for the experience factory organization. The project organization gets a system that enables them to share knowledge and help each other in a peer-to-peer manner. This peer-to-peer sharing is helpful for the EF group as well because it relieves them from the pressure to deliver immediately complex experience packages, which take time to develop. As time goes by, dust is collected in the experience base. This dust becomes the basis for the analysis and synthesis of the EF group, which turns the dust into Gems and delivers them to the project organization.

REFERENCES

Ackerman, M.S., 1990. Answer garden: A tool for growing organizational memory. Proceedings of the ACM SIGOIS and IEEE CS TC-OA Conference on Office Information Systems, (COCS'90), ACM New York, NY, USA., pp: 31-39. DOI: 10.1145/91474.91485

Basili, V.R., G. Caldiera and D.H. Rombach, 1994. The experience factory. Encyclopedia Software Eng., 2: 511-519.

Basili, V.R., M. Lindvall and P. Costa, 2001. Implementing the experience factory concepts as a set of experience bases. Proceedings of 13th International Conference on Software Engineering and Knowledge Engineering, (ICSEKE'01), Knowledge Systems Institute, USA., pp: 102-109.

Glaser, B.G. and A.L. Strauss, 1967. The Discovery of Grounded Theory: Strategies for Qualitative Research. 1st Edn., Aldine Transaction, USA., ISBN-10: 0202302601, pp: 271.

Jabar, M.A., F. Sidi and M.H. Selamat, 2010. Tacit knowledge codification. J. Comput. Sci., 6: 1170-1176. DOI: 10.3844/jcssp.2010.1170.1176

Lindvall, M., I. Rus, R. Jammalamadaka and R. Thakker, 2001. Software tools for knowledge management: A DACS State-of-the-art report. 2001, Fraunhofer Center for Experimental Software Engineering Maryland and The University of Maryland: Maryland. http://portal.acm.org/citation.cfm?id=1632160

McGarry, F., 1994. Software Process Improvement in the NASA Software Engineering Laboratory, CMU/SEI-95-TR-22, University of Maryland. http://www.sei.cmu.edu/library/abstracts/reports/94tr022.cfm

Miles, M.B. and A.M. Huberman, 1994. Qualitative Data Analysis: An Expanded Sourcebook. 2nd Edn., Sage Publications, USA., ISBN-10: 0803955405, pp: 352.