Overview of configurators as effective tools for corporate knowledge management

Gerhard Fleischanderl

Siemens Austria, Program and Systems Engineering
Erdberger Laende 26
A-1030 Vienna, Austria
gerhard.fleischanderl@siemens.at

Abstract

Knowledge management is widely discussed, but scarcely supported with tools and information systems. Configurators capture knowledge from product design, marketing and sales. By that configurators support three of the core processes of knowledge management: distribution, usage and preservation of knowledge. Constraint-based configurators and other model-based reasoning systems provide the technology which suits the requirements of knowledge management. Ease of maintenance of a configurator's knowledge base corresponds to its usefulness for knowledge management.

Introduction

Knowledge management in organizations has become a buzz-word in management literature. Organizational culture and an attitude for learning are prerequisites for the success of knowledge management. Besides that it needs tools for supporting its implementation in organizations.

Knowledge management has distinct aspects. Some of them lend themselves to the application of knowledge-based systems, in particular configurators. Knowledge-bases capture part of the know-how which is held collectively in an organization. For instance, configurators implement knowledge about configuring complex products. Furthermore, knowledge-based systems help distribute their focused knowledge to users who may be anywhere in the organization.

Configurators are showpieces for handling know-how in an organization and are, thus, relevant to knowledge management. This article presents an overview and analyzes the relationship between configurators and knowledge management. Technologies for configurators are discussed with respect to their role in knowledge management.

Knowledge management

Knowledge

Knowledge is broader, deeper, and richer than data or information. Davenport and Prusak (1998) give a working definition of knowledge.

> Knowledge is a fluid mix of framed experience, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information. It originates and is applied in the minds of knowers. In organizations, it often becomes embedded not only in documents or repositories but also in organizational routines, processes, practices, and norms.

There are two types of knowledge:

- *Tacit* knowledge is in the heads of people and cannot be captured directly.
- *Explicit* knowledge is formulated, e.g. in a document.

In any organization (a company or a nonprofit organization), knowledge assets become more and more valuable, yet are hard to pin down.

Knowledge management

Knowledge management is a management function. It can be characterized by its core processes (based on Probst et al., 1998).

1. *Identification* of knowledge: analyzing the environment.
   How does the organization obtain a general view on existing knowledge internally and externally?
2. *Purchase* of knowledge: buying instead of making.
   What knowledge does the organization buy externally? Who is recruited for the organization?

3. *Development* of knowledge: using the creativity of the people in the organization.

How does the organization create new knowledge?

4. *Distribution* of knowledge: bringing knowledge from the creators to the users.

How does the organization transfer knowledge to the place where it is needed?

5. *Usage* of knowledge: transforming knowledge into results.

How does the organization make sure that knowledge is actually used in daily work?

6. *Preservation* of knowledge: selecting, storing and updating.

How does the organization protect itself from losing knowledge?

7. *Coordination* of the knowledge management process. How does the organization control its knowledge-creating activities?

These core processes correspond to the basic knowledge management activities that are arranged around an organizational memory (Abecker et al., 1998).

Knowledge management can be summarized by knowledge sharing among the people in an organization. Putting existing knowledge to use and generating new knowledge is at the heart of knowledge management.

Within intellectual capital (Stewart, 1997), the purpose of structural capital is 'to codify bodies of knowledge that can be transferred and to preserve the recipes that might otherwise be lost'. This describes one aim of knowledge management which is particularly relevant for the application of software systems.

The repositories for explicit knowledge are either broad (e.g. documents, files on the Internet) or focused (e.g. well-structured knowledge-bases). Focused knowledge requires more effort to prepare it, but is better suited for using it in software systems. This is where knowledge-based systems like configurators come into play.

The role of configurators

Configurators are software systems for supporting the configuration process of complex products. Parts of that process are handled automatically by configurators. Configurators make tacit knowledge explicit by codifying and modeling it in knowledge-bases (e.g. rules) or other parts of software systems (e.g. user interfaces). That knowledge is accessible to the knowledge engineers, who create and maintain the knowledge-base. In a configurator, knowledge is transformed into an application that is made available to many users. The normal users employ a configurator mainly as black-box and do not usually extract knowledge from the configurator.

The role of configurators in knowledge management is mainly in these core processes:

- distribution
- usage
- preservation

Before the distribution of knowledge, pieces of knowledge from various sources have to be combined. People from various business functions and departments contribute to the configurator, e.g. sales and product development. By testing and validating pieces that are added to the knowledge-base, a configurator helps to build a knowledge-base that provides value to the users.

The software application that contains valuable know-how about the configurable product and the configuration process is made available to the users of the configurator. The embedded knowledge is used implicitly or becomes visible occasionally, e.g. for generating explanations in case of contradictions between knowledge-base and product description.

To make preservation of knowledge work, the knowledge-base has to be understandable even after a long time and for different people. In other words, a configurator ought to support long-term maintenance.
In the spiral of knowledge (Nonaka, 1991) people should be able to re-engineer the knowledge from a configurator's knowledge-base and use it for new tacit knowledge. The extended or refined knowledge will then be captured again in the knowledge-base.

Jobs become ever more specialized, and the communication between knowledgeable specialists gets more important than the organizational structure (Drucker, 1988). Therefore, there is greater need for integration of knowledge from various sources. Knowledge-based tools can help to coordinate and distribute explicit knowledge by capturing it in systems and software.

When configurators are applied successfully in a company, they are useful tools also for knowledge management. Yet, configurators are not the universal tools for all the aspects of knowledge management.

Currently, the wide-spread usage of configurators is in well-defined business functions like quotations and detailed configurations done by sales people. In the future, the technology of configurators may be used in other areas, e.g. configuration and re-configuration of computer networks during operation.

Configurator technology

For knowledge management in general, several expert system and artificial intelligence technologies may be used (Davenport and Prusak, 1998; Sabin and Weigel, 1998):

- Neural networks: are mainly used for classification and pattern recognition. The neural network remains a black box and cannot be converted back to explicit knowledge. This technology is only marginally important for configuration tasks.
- Case-based reasoning tools: extract knowledge from cases (Kolodner, 1993). They are mainly used for customer-service applications, e.g. at help desks and in call-centers. The problem characteristics and solutions of the underlying cases are still retrievable. For configurations, a case from a small collection of standard products can be used and modified to fulfill a customer requirement.
- Rule-based expert systems: represent the knowledge as production rules. Over time, they tend to be difficult to maintain or add knowledge to because the interrelationships between the rules become more complex. The most well-known application is Digital's R1/XCON system (McDermott, 1982; Bachant and McDermott, 1984).
- Constraint-based systems: use a generic model for configuration tasks (Mittal and Frayman, 1989). They employ constraints to restrict the ways various components can be combined and use object-oriented models underneath. Interactions between constraints are less complex than in rule-based systems. Extensions to the classical constraint-satisfaction problem paradigm, e.g. generic constraint-satisfaction (Fleischanderl et al., 1998), add expressiveness to constraint-based configurators.
- Other model-based reasoning systems are similar to constraint-based systems concerning their usefulness for knowledge management: description logic-based systems (McGuinness and Wright, 1998); resource-based systems (Juengst and Heinrich, 1998).

With respect to knowledge management, constraint-based and other model-based configurators look more useful because they provide the opportunity to use the knowledge-base as a repository that can be maintained efficiently over time.
Besides the technology of the configurator itself, its integration into the wider information system, e.g. enterprise resource planning (Haag, 1998), is crucial for the successful deployment of a configurator.

A configurator application example

Using the telecom switching systems configurator Lava (Fleischanderl et al., 1998) as an example, we demonstrate the relationships to knowledge management. Lava has been in production use for more than two years.
Lava was implemented with the platform Cocos that uses constraint-based reasoning system and a knowledge-base with class descriptions and constraints. The products to be configured with Lava are large switching systems, comprising up to 50,000 components approximately.

When the Lava project was launched, knowledge management was not an explicit requirement. Looking back we recognize several *advantages* brought about by Lava for the goals of knowledge management.
- The requirements on the configurable product range are made explicit and captured in the knowledge-base in a declarative manner.
- Knowledge from various sources in the company is combined, e.g. from sales, marketing, and product development.
- The knowledge-base prevents redundancies which makes maintenance faster. Every constraint can be used for three purposes, namely generating (new components), checking (existing components), and explaining (inconsistencies in a configured system).
- Maintaining the knowledge-base over the years proved to be easy. Due to the structure of the knowledge-base (class hierarchy and constraints) the bulk of updates did not touch the existing parts, but were added with new classes or constraints.

With respect to knowledge management, Lava shows *drawbacks* of knowledge-based systems:
- Creating and maintaining the knowledge-base for complex products requires experts. Thus, the benefits of explicit knowledge in a 'good' knowledge-base are restricted to a small group of persons.
- The constraint-based platform Cocos requires modeling on an abstract level. Cocos produces explanations by presenting instantiated constraints. There are situations where occasional users want more specific explanations that guide them how to repair an inconsistency.
- Cocos uses a class description and constraint language which requires transformation of data that are fed from external sources into the knowledge-base. (However, there is no universal knowledge description language available, and might never be.)

In a nutshell, Lava proved easy to maintain and thus fulfills an important criterion for knowledge management suitability. Lava's constraint knowledge-base can be used for purposes beyond configuration.

Bringing it together

Experiences with knowledge management showed that several requirements are crucial for its success in industrial practice (Abecker et al., 1998).
- Collection and systematic organization of information from various sources.
- Minimization of up-front knowledge engineering. A knowledge management system must provide benefits quickly.
- Exploiting user feedback for maintenance and evolution. Maintenance efforts must be minimized.
- Integration into existing work environment.
- Active presentation of relevant information.

The first and fourth criteria are also crucial for the success of configurator applications.

The suitability of a configurator for knowledge management ought to be a criterion for selecting the configurator technology for several reasons:

- The success criteria for knowledge management initiatives are similar to those for configurator applications.
- The process of capturing the relevant knowledge, i.e. transforming tacit knowledge into an explicit knowledge-base, is related to knowledge sharing, which is at the heart of knowledge management.

Ease of maintenance of a configurator's knowledge base corresponds to its usefulness for knowledge management.

Configurators (if in production use) are success stories for handling knowledge in an organization, thus promoting activities in knowledge management.
Yet, knowledge management reaches beyond specialized tasks like the configuring of products. Focusing on tools and technology alone is a pitfall not unknown to information systems. 'Soft' issues like organizational culture must not be neglected in knowledge management.

Further work ought to explore how configurators can be improved to provide more benefits to knowledge management in organizations.

Conclusion

Configurators play an important part in the knowledge management of organizations. By helping to turn tacit knowledge about configuring complex products into usable software applications, an organization's knowledge is put to use. Explicit knowledge captured in well-structured knowledge-bases is accessible to maintenance during the use of the configurator.
To maximize the benefits of configurators for corporate knowledge management, the technology of a configurator and its interfaces to other software systems must be considered thoroughly. The long-term success of a configurator application corresponds to ease of maintenance and is to a large degree determined by its suitability for knowledge management.

References

Abecker, A.; Bernardi, A.; Hinkelmann, K.; Kühn, O.; and Sintek, M. 1998. Toward a technology for organizational memories. *IEEE Intelligent Systems & their applications* 13(3):40-48, May/June.

Bachant, J.; and McDermott, J. 1984. R1 revisited: four years in the trenches. *AI Magazine* 5(3):21-32, Sep.

Davenport, T. H.; and Prusak, L. 1998. *Working knowledge: how organizations manage what they know.* Boston, Mass.: Harvard Business School Press.

Drucker, P. F. 1988. The coming of the new organization. *Harvard Business Review* 66(1):45-53, Jan-Feb.

Fleischanderl, G.; Friedrich, G. E.; Haselböck, A.; Schreiner, H.; and Stumptner, M. 1998. Configuring large systems using generative constraint satisfaction. *IEEE Intelligent Systems & their applications* 13(4):59-68, July/Aug.

Haag, A. 1998. Sales configuration in business processes. *IEEE Intelligent Systems & their applications* 13(4):78-85, July/Aug.

Juengst, W. E.; and Heinrich, M. 1998. Using resource balancing to configure modular systems. *IEEE Intelligent Systems & their applications* 13(4):50-58, July/Aug.

Kolodner, J. 1993. *Case-based reasoning.* San Mateo, Cal.: Morgan Kaufman Publishers.

McDermott, J. 1982. A rule-based configurer of computer systems. *Artificial Intelligence* 19(1):39-88.

McGuinness, D. L.; and Wright, J. R. 1998. An industrial-strength description logic-based configurator platform. *IEEE Intelligent Systems & their applications* 13(4):69-77, July/Aug.

Mittal, S.; and Frayman, F. 1989. Towards a generic model of configuration tasks. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence,* 1395-1401. San Mateo, Cal.: Morgan Kaufman Publishers.

Nonaka, I. 1991. The knowledge-creating company. *Harvard Business Review* 69(6):96-104, Nov-Dec.

Probst, G.; Raub, S.; and Romhardt, K. 1998. *Wissen managen: Wie Unternehmen ihre wertvollste Ressource optimal nutzen* (Managing knowledge: How companies use their most important resource optimally, 2nd edn., in German). Frankfurt: Gabler.

Sabin, D.; and Weigel, R. 1998. Product configuration frameworks - a survey. *IEEE Intelligent Systems & their applications* 13(4):42-49, July/Aug.

Stewart, T. A. 1997. *Intellectual capital: the new wealth of organizations.* London: Nicholas Brealey.