Chapter 9
Capability Maturity

Alfs T. Berztiss
University of Pittsburgh, USA

Category: Organizational Aspects of Knowledge Management

INTRODUCTION

The dependence of any organization on knowledge management is clearly understood. Actually, we should distinguish between knowledge management (KM) and knowledge engineering (KE): KM is to define and support organizational structure, allocate personnel to tasks, and monitor knowledge engineering activities; KE is concerned with technical matters, such as tools for knowledge acquisition, knowledge representation, and data mining. We shall use the designation KMKE for knowledge management and knowledge engineering collectively. KM is a very young area—the three articles termed "classic works" in Morey, Maybury, and Thuraisingham (2000) date from 1990, 1995, and 1996, respectively. We could regard 1991 as the start of institutionalized KM. This is when the Skandia AFS insurance company appointed a director of intellectual capital. KE has a longer history—expert systems have been in place for many years. Because of its recent origin, KMKE is characterized by rapid change. To deal with the change, we need to come to a good understanding of the nature of KMKE.

One of the lasting contributions of the business reengineering movement is the view that an enterprise is to be regarded as a set of well-defined processes (Davenport, 1993; Berztiss, 1996). This implies that KMKE also should be a process. Implementation of a process has two aspects: there is need for a procedural definition, and for an understanding of the resources and capabilities needed to implement the procedures and manage the process. Here, we will not be considering the procedures. Our purpose is to set up a model that identifies the capabilities needed to define, implement, and maintain the KMKE process.

The Background section of this article introduces capability models. In the Focus section, we define a capability model for KMKE in general terms and look at the management and engineering sides of this model. Then, we look into the future and offer a conclusion.

BACKGROUND: CAPABILITY MATURITY AND SOFTWARE

One area that has had long experience with processes is software engineering, and we turn to it for

DOI: 10.4018/978-1-59904-931-1.ch009

guidance on how to construct a capability model for KMKE. The software Capability Maturity Model (CMM-SW) was introduced by Humphrey (1989) and elaborated by a team of researchers at the Software Engineering Institute (1995). A later development is CMMI, which stands for CMM Integration. This is a suite of models where CMMI-SW (CMMI Product Team, 2002) is the model for software development. We shall be guided by the original model for two main reasons: First, there is greater familiarity with CMM-SW than with CMMI; second, the original CMM-SW has inspired a number of models that address the specific capabilities needed for specialized applications. Thus, there are CMMs for reuse (Davis, 1993), formal specification (Fraser & Vaishnavi, 1997), maintenance (Kajko-Mattson, 2001), an initial version for KM (Berztiss, 2002a), e-commerce (Berztiss, 2002b), and data quality management (Berztiss, 2004). An investigation of how to adapt CMM-SW for such nontraditional projects as product-line development, database development, and schedule-driven development also has been undertaken (Johnson & Brodman, 2000). Considerable evidence exists on the effectiveness of CMM-SW and CMMI for improving quality and reducing costs (Goldenson & Gibson, 2003).

The CMM-SW has five maturity levels. Level 1 is the base from which an organization moves upward by satisfying a set of requirements expressed as key process areas (KPAs). This level structure with the total of 18 KPAs is shown in Table 1. All KPAs of Level 2 relate to management, those of Level 3 to management and engineering, and those of Levels 4 and 5 relate primarily to engineering.

In CMM-SW, the definition of a KPA starts with a statement of it "goals," a "commitment to perform," which is essentially a policy statement committing the organization to the satisfaction of these goals, and an "ability to perform" statement, which lists the resources that have to be allocated. Next comes a list of activities that need to be performed in order to achieve the goals of the KPA. This can be regarded as a requirements statement that tells what is to be done without going into details of how the activities are to be

Table 1. Key process areas of CMM-SW

Level 3	Level 5
Organizational process focus Organizational process definition Training program Integrated software management Software product engineering Intergroup coordination Peer reviews	Defect prevention Technology change management Process change management
Level 2	Level 4
Requirements management Software project planning Software project tracking and oversight Software subcontractor management Software quality assurance Software configuration management	Quantitative process management Software quality management