Integrated Analysis and Design of Knowledge Systems and Processes

MARK NISSEN, Naval Postgraduate School
MAGDI KAMEL, Naval Postgraduate School
KISHORE SENGUPTA, Naval Postgraduate School

Although knowledge management has been investigated in the context of decision support and expert systems for over a decade, interest in and attention to this topic have exploded recently. But integration of knowledge process design with knowledge system design is strangely missing from the knowledge management literature and practice. The research described in this chapter focuses on knowledge management and system design from three integrated perspectives: 1) reengineering process innovation, 2) expert systems knowledge acquisition and representation, and 3) information systems analysis and design. Through careful analysis and discussion, we integrate these three perspectives in a systematic manner, beginning with analysis and design of the enterprise process of interest, progressively moving into knowledge capture and formalization, and then system design and implementation. Thus, we develop an integrated approach that covers the gamut of design considerations from the enterprise process in the large, through alternative classes of knowledge in the middle, and on to specific systems in the detail. We show how this integrated methodology is more complete than existing developmental approaches and illustrate the use and utility of the approach through a specific enterprise example, which addresses many factors widely considered important in the knowledge management environment. Using the integrated methodology that we develop and illustrate in this chapter, the reader can see how to identify, select, compose and integrate the many component applications and technologies required for effective knowledge system and process design.

KNOWLEDGE MANAGEMENT AND SYSTEM DESIGN

The power of knowledge has long been ascribed to successful individuals in the organization, but today it is recognized and pursued at the enterprise level through a practice known as knowledge management (see Davenport and Prusak 1998). Although knowledge management has been investigated in the context of decision support systems (DSS) and expert systems (ES) for over a decade (e.g., see Shen 1987), interest in and attention to this topic have exploded recently. For example, knowledge capital is commonly discussed as a factor of no less importance than the traditional economic inputs of labor and finance (Forbes 1997), and the concept *knowledge equity* is now receiving theoretical treatment through research (e.g., see Glazer 1998).

Many prominent technology firms now depend upon knowledge-work processes to compete through innovation more than production and service (McCartney 1998), and Drucker (1995, p. 271) writes, "knowledge has become the key economic resource and the dominant—and perhaps even the only—source of comparative advantage." This follows his assertion that increasing knowledge-work productivity represents the great management task of this century, on par with the innovation and productivity improvements made through industrialization of manual-work processes (Drucker 1978). Brown and Duguid (1998, p. 90) add, "organizational knowledge provides synergistic advantage not replicable in the marketplace." Indeed, some forecasts suggest knowledge work (e.g., performed by professionals and managers) will account for nearly 25% of the workforce soon after the 21st century begins (Labor 1991). And partly in anticipation, fully 40% of Fortune-1000 companies claim to have established the role of Chief Knowledge Officer (CKO) in their companies (Roberts 1996). Miles et al. (1998, p. 281) caution, however, "knowledge, despite its increasing abundance, may

Manuscript originally submitted March 2, 1999; Revised June 8, 1999; Accepted August 1 1, 1999 for publication.

elude managerial approaches created in 20th century mindsets and methods."

In fact, knowledge is proving difficult to manage, and knowledge work has been stubbornly resistant to reengineering and process innovation (Davenport 1995). For one thing, Nonaka (1994) describes knowledge-creation as primarily an individual activity, performed by knowledge workers that are mostly professional, well-educated and relatively autonomous, often with substantial responsibility in the organization. They tend to seek and value their relative autonomy and often resist perceived interference by management in knowledge-work activities (Davenport et al. 1996). Moreover, substantial, important knowledge is tacit, unstructured (Nonaka 1994) and external to the organization (Frappaolo 1998). This can greatly impede the identification, acquisition, interpretation and application of such knowledge. Also, corporate knowledge has historically been stored on paper and in the minds of people (O'Leary 1998). Paper is notoriously difficult to access in quantity and keep current on a distributed basis, and knowledge kept in the minds of workers is vulnerable to loss through employee turnover and attrition. Vulnerability to such loss of knowledge is exacerbated by recent waves of downsizing associated with reengineering (McCartney 1998) and the constrained labor markets affecting many professions (esp. information technology and software engineering).

Moreover, most information technology (IT) employed to enable knowledge work appears to target data and information, as opposed to knowledge itself (cf. Ruggles 1997). We feel this contributes to difficulties experienced with knowledge management to date. Knowledge, almost by definition, lies at the center of knowledge work, yet it is noted as being quite distinct from data and information (e.g., see Davenport et al. 1998, Nonaka 1994, Teece 1998). Drawing from Arrow (1962) and others, we understand that even information economics has many important differences from standard economic theory (e.g., negligible marginal costs, network externalities, consumption without loss of use), but our understanding of *knowledge economics* is entirely "primitive" (Teece 1998).

Further, extant IT used to support knowledge management is limited primarily to conventional database management systems (DBMS), data warehouses and mining tools (DW/DM), intranets/extranets and groupware (O'Leary 1998). Arguably, just looking at the word "data" in the names of many "knowledge management tools" (e.g., DBMS, DW/DM), we are not even working at the level of information, much less knowledge. And (esp. Web-based) Internet tools applied within and between organizations provide a common, machine-independent medium for the distribution and linkage of multimedia documents, but extant intranet and extranet applications focus principally on the management and distribution of information, not knowledge per se. Although a great improvement over previous stove-piped systems, islands of automation and other information systems maladies, as Nonaka (1994, p. 15) states, such "information is [just] a flow of messages," not knowledge.

Groupware offers infrastructural support for knowledge work and enhances the environment in which knowledge artifacts are created and managed, but the management of knowledge itself remains indirect. For instance, groupware is widely noted as helpful in the virtual office environment (e.g., when geographically-dispersed knowledge workers must collaborate remotely) and provides networked tools such as shared, indexed and replicated document databases and discussion threads (e.g., Lotus Notes applications), as well as shared "white boards," joint document editing capabilities and full-duplex, multimedia communication features. These tools serve to mitigate collaborative losses that can arise when rich, face-to-face joint work is not practical or feasible, and groupware can facilitate the reuse of knowledge-work artifacts (e.g., successful consultant proposals, presentations and analyses).

However, as we learned through the painful, expensive and failure-prone "first wave" of reengineering (see Cypress 1994), simply inserting IT into a process in no way guarantees performance improvement. Indeed, many otherwise successful and effective firms experienced process *degradation* as the result of reengineering (e.g., see Caron et al. 1994, Hammer and Champy 1993). This point is underscored by Hammer (1990), who colorfully refers to such practice as "paving the cowpaths" and "automating the mess" (e.g., making a broken process simply operate broken faster).

Drawing all the way back to Leavitt (1965) and others (e.g., Davenport 1993, Nissen 1998), new IT needs to be integrated with the design of the *process* it supports, which includes consideration of the organization, people, procedures, culture and other key factors, in addition to technology. Such integration of knowledge process design with knowledge system design is strangely missing from the knowledge management literature and practice. And what about the information systems (IS) methodologies, techniques and tools used to design and implement knowledge systems? Are they the same, familiar ones employed over the decades for databases, transaction process systems, expert systems, groupware and other applications? Should they be? These are some of the critical knowledge management questions addressed through this paper.

The research described in this chapter is focused on knowledge management and system design from three integrated perspectives: 1) reengineering process innovation, 2) expert systems knowledge acquisition and representation, and 3) information systems analysis and design. We integrate these three perspectives in a systematic manner, beginning with analysis and design of the enterprise process of interest, progressively moving into knowledge capture and formalization, and then system design and implementation. Thus, we offer an integrated approach that covers the gamut of design considerations from the enterprise process in the large, through alternative classes of knowledge in the middle, and on to specific systems in the detail.

The central premise of this work is, although knowledge management represents a phenomenon of relatively new widespread interest and attention in research and practice, many of its underlying elements are actually quite familiar and have been effectively addressed for many years (decades in some cases) through work in process redesign (e.g., integration of information technology enablers with organizational design, human resources, information availability, inter-organizational alliance, workflow modification and other process transformations), artificial intelligence (e.g., knowledge capture and formalization, distributed inference, knowledgebase design) and information systems (e.g., structured and object-oriented analysis & design, database development, decision support systems). At this stage of our research, we have developed many compelling examples of well-established methodologies, techniques and tools being composed to support integrated analysis and design of knowledge systems and processes.

In the sections that follow, we outline a three-tier framework for examining alternative methodologies, techniques and tools and employ this framework to provide a high-level overview of well-established approaches from each of the areas above. Drawing from the literature, we examine a number of extant knowledge management systems and practices to classify and analyze current developmental methodologies, techniques and tools. We then outline a contingent feature space of specific elements, levels and stages comprising knowledge management and use this to develop an integrated analysis and design approach tailored to each key aspect of knowledge system and process design. With this, we develop a set of contextual factors (e.g., organizational environment, knowledge characteristics) that draw insight into strengths and limitations of various approaches. This represents a central contribution of the chapter, as it reveals the underlying components of knowledge management and prescribes design guidance specific to each. We then discuss how to employ the design approach developed above through a specific enterprise example, which addresses many factors widely considered important in the knowledge management environment (e.g., cross-functional virtual teams, collaborative work, distributed tacit and explicit knowledge, both routine and non-routine work processes, a dynamic market/organizational environment) and illustrates the use and utility of our integrated approach to analysis and design of knowledge systems and processes. The final section closes with key conclusions and implications for practice, in addition to a focused agenda for future research along these lines.

EXTANT APPROACHES TO SYSTEM AND PROCESS DESIGN

In this section, we outline a three-tier framework for examining alternative methodologies, techniques and tools employed to develop systems and processes, and we present substantive discussion of methods from reengineering, expert systems and information systems domains. This provides background necessary to understand these diverse but overlapping approaches and to appreciate their respective and integrative applicability and potential in the context of knowledge systems and processes.

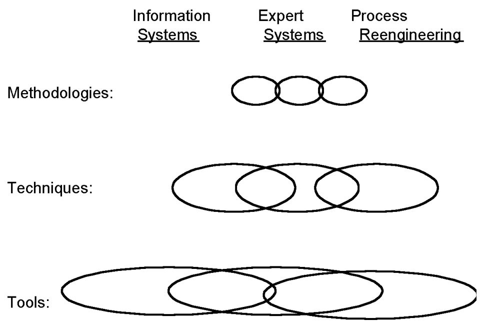
Analytical Framework

Central to information system and process development are the concepts *methodologies*, *techniques*, and *tools*. These three concepts work together to provide a solid framework for facilitating the development of information systems, expert systems and enterprise processes. Methodologies are comprehensive, step-by-step approaches that guide the development of a system or process. They provide guidance on what should be done, when it should be done, how it should be done and who should do it. Examples of methodologies used in information system development include structured analysis, information engineering, rapid prototyping, object-oriented development and others. Some aspects of these information system methodologies (e.g., prototyping) are also employed for expert systems development, and even reengineering shares some methodological commonality (e.g., through the BPR Life Cycle; see Kettinger et al. 1995). Each methodology generally includes several developmental techniques.

Techniques are specific processes used in conjunction with one or more methodologies that result in well thought-out, complete and comprehensible deliverables. Techniques provide support to a wide range of systems development activities from planning and analysis, through design and implementation, to system maintenance and retirement. Examples of techniques used for information systems development include interviewing, use case modeling, data flow modeling, entity-relationship modeling, structured design, object-oriented programming and others. More so than is the case with methodologies above, several of these information system techniques (e.g., interviewing, use cases) are also employed for expert systems development. But other expert system development techniques (e.g., knowledge acquisition, knowledge representation) focus on knowledge—as opposed to information or data—directly and are unique to the class of knowledge-based systems (e.g., including expert systems, intelligent agents). Likewise, reengineering involves several of the same techniques. But it too has a unique set (e.g., pathology diagnosis, transformation matching) at the techniques level.

Tools are computer programs that facilitate the implementation of techniques within the overall guidelines of a

Figure 1: Commonality of Methodologies, Techniques and Tools



particular development methodology. Examples of information system development tools include program editors, compilers and debuggers, modeling applications (e.g., for dataflow diagrams, entity-relationship diagrams, object models), configuration management modules, test simulators and others. Even more so than above, considerable commonality exists between the sets of tools used for information systems development with those employed for developing expert systems and reengineering engagements.

This relationship between methodologies, techniques and tools employed for the three classes of systems—information systems, expert systems and enterprise processes—is roughly depicted by the Venn Diagrams presented in Figure 1. Notice considerable uniqueness (e.g., very little overlap) at the level of methodologies, increasing commonality through levels of techniques, and substantial overlap at the tools level. The relative sizes of ovals in the figure also depict the relationship from above in terms of numbers; that is, there are relatively few qualitatively-different methodologies, but numerous unique techniques and a multitude of diverse tools are employed across the three system classes. From this early examination, we might expect knowledge management methodologies to be quite unique. But they are likely to involve several common techniques and abundant tools used for information systems, expert systems and reengineering. This provides insight into our development of an integrated approach to knowledge system and process development.

Examination of Established Approaches

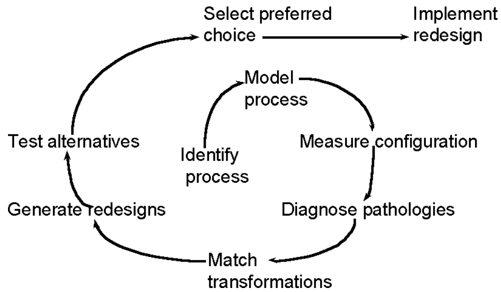
Here, we focus the discussion on examination of established approaches to analysis and design in the areas of reengineering, expert systems and information systems. The latter two areas serve to represent nearly all IS analysis and design activity—with expert systems approaches oriented directly toward knowledge itself—and incorporation of reengineering discussion serves to integrate both systems and processes into our analysis. This provides the substance for inclusion into our three-tier framework from above.

Reengineering methodologies. Business process reengineering (BPR) involves radical redesign of enterprise processes (Hammer and Champy 1993) intended to effect dramatic, order-of-magnitude performance improvement (Davenport 1993). A number of reengineering methodologies have been developed by BPR experts. They reflect a synthesis of many process redesign endeavors and are generally developed by BPR consultants who are widely acknowledged as the most knowledgeable experts in the field. Admittedly, some of these "methodologies" (e.g., Hammer and Champy 1993) appear to accomplish little more than motivating the case for BPR (Cole 1994) and are shown to have substantial room for improved analysis (Hansen 1994); in essence they answer the question of *whether* to reengineer. However, others (e.g., Davenport 1993) provide a start-to-finish guide to undertaking process improvement, answering questions such as *what* steps need to be taken and in which order. Additionally, a number of academic investigations build upon the kind of knowledge available through expert reengineering methodologies—for example contributing knowledge in terms of frameworks (e.g., Davidson 1993, Guha et al. 1994) and guidelines (e.g., Henderson and Venkatraman 1993, Klein 1994) that begin to answer operationalized questions such as *how* to accomplish the redesign steps from above.

Nissen (1998) describes reengineering in terms of process-redesign activities organized as an evolutionary spiral to denote increasing process knowledge and understanding as the reengineering activity progresses. This sequence of activities, delineated in Figure 2, represents a blend of expert reengineering methodologies—particularly those of Andrews and Stalick (1994), Davenport (1993), Hammer and Champy (1993), Harrington (1991) and Johansson et al. (1993)—synthesized together to compose an analytical method supporting measurement. Step one is to identify a target process for redesign. Next, a model is constructed to represent the baseline (i.e., "as is") configuration of this process, and configuration measurements then drive the diagnosis of process pathologies. The diagnostic results are used in turn to match the appropriate redesign transformations available to "treat" pathologies that are detected. This sequence of analytical activities leads systematically to the generation of one or more redesign alternatives, which most experts argue should be tested through some mechanism (esp. simulation) prior to selection of a preferred alternative for implementation.

The analysis and design phases of reengineering (generally called "redesign") are followed by implementation and maintenance in what is known as the BPR Life Cycle (Kettinger et al. 1995). And although the life cycle steps are often described as a sequence, their performance is generally very iterative. Generally, process redesign involves analytical activities, whereas implementation and maintenance require making physical and procedural changes in enterprise processes. Redesign requires understanding the objectives and strategies of an enterprise and generally entails process modeling and analysis that results in one or more (re)designs for the process in question. Implementation represents a key

Figure 2: General Redesign Process



stage of activities in the reengineering life cycle and represents a major area of risk in terms of BPR success. Change management is very important for effective implementation, and we now have the benefit of research results such as "preconditions for success" (Bashein et al. 1994), "tactics for managing radical change" (Stoddard and Jarvenpaa 1995), revelations of "reengineering myths" (Davenport and Stoddard 1994) and greater insight into implementation problems (Clemons et al. 1995, Grover et al. 1995).

There is considerable debate regarding how to define "reengineering maintenance." Some experts describe reengineering as an "all or nothing proposition" (Hammer 1990, p. 105), in which existing enterprise processes are radically transformed through "redesign with a blank sheet of paper" (Hammer and Champy 1993, p. 131) and "the proverbial clean slate" (Hammer and Stanton 1995, p. 4). And most researchers agree the radical nature of reengineering makes it an inherently discrete event (e.g., see Cole 1994, Davenport 1993, Nissen 1996). But reengineering shares many methodological elements with Total Quality Management (TQM; see Flood 1993, Harrington 1991, Hoffherr 1994), and a number of reengineering methodologies (e.g., see Andrews and Stalick 1994) include the TQM practice of continuous process improvement (CPI)—an incremental-change approach—as the effective "maintenance" phase of reengineering. Such reengineering maintenance through CPI is justified by Davenport (1993, p. 14): "Lest it slide back down the slippery slope of process degradation, [after process redesign] a firm should then pursue a program of continuous improvement for the post-innovation process." The maintenance phase thus completes the life cycle and often signals the beginning of a new analysis effort (e.g., additional process redesign).

Expert System Development Methodologies. Expert systems are computer programs that emulate the problem solving and experience of experts in specific domains. Expert systems thus provide a way to capture and apply human knowledge, expertise and experience via computer. Because they focus on knowledge directly, expert systems appear to offer particular promise in knowledge management. The process of developing an expert system is called knowledge engineering (Prerau 1990). Similar to the system development life cycle of traditional information systems development, the knowledge engineering process consists of a number of phases, each consisting of several tasks. As with the reengineering life cycle above, although knowledge engineering phases and tasks are usually shown in sequence, in practice they are conducted iteratively. And like the life cycle from above, expert system development can similarly be portrayed in terms of an evolutionary-spiral process. The following is a summary of the essential activities conducted in each of six phases: 1) problem assessment, 2) knowledge acquisition, 3) knowledge representation, 4) system implementation, 5) verification and validation, and 6) maintenance.

Problem assessment pertains to the applicability and feasibility of an expert system solution to a particular problem. A good business case is often required in this first phase. As implied by the name, the knowledge acquisition phase involves acquisition of knowledge from a domain expert and/or other sources of knowledge (Kamel 1999). It also involves interpreting, analyzing and documenting the acquired knowledge. It is well understood that tacit knowledge is more difficult to acquire than its explicit counterpart, and many human experts are truly outstanding at what they do but unable to clearly articulate *how* they accomplish their knowledge work. Knowledge representation involves the selection of a knowledge representational scheme and control strategy. Acquired knowledge is represented using one of several ontologies and representational formalisms (e.g., rules, frames, scripts).

The expert system implementation phase is very much like its reengineering counterpart above and information system counterpart below. This phase involves coding the knowledge acquired as above using appropriate expert system development software (e.g., a development "shell") and one or more of the selected representational formalisms. Verification and validation (V&V) ensures the developed system correctly implements its initial specification and performs at an acceptable level of expertise. This step shares considerable similarity with information systems V&V, except that *knowledge* validation is unique to expert systems development. For example, the Turing Test represents a textbook approach to validating expert knowledge (see Turban and Aronson 1998). Very briefly, if an informed person cannot tell the difference in knowledge-work performance between an expert human and an expert system, such system is deemed validated according to the Turing Test. Finally, maintenance represents an ongoing phase that corrects system errors and deficiencies. It also updates the system knowledge as requirements evolve and completes the development cycle. As with reengineering above, the expert system maintenance phase often signals the beginning of a new analysis effort (e.g., additional knowledge acquisition and representation).

Information System Development Methodologies. A widely used information system development methodology is the system development life cycle (SDLC). The SDLC is a common methodology for system development that consists of phases, sub-phases and tasks to guide the system analysis and design effort. As above, although the SDLC appears to be a sequential set of phases, its implementation is usually highly iterative. Almost every organization uses a slightly different life cycle model, with a varying number of identifiable phases. Here, we consider a SDLC that consists of six phases: 1) information system planning, 2) project initiation and planning, 3) system analysis, 4) logical and physical design, 5) system implementation, and 6) maintenance.

Information systems planning is usually part of the organization's corporate and systems planning process. It identifies the information needs of the organization as a whole and the potential projects to meet these needs. The project initiation and planning phase defines the scope of the proposed system and specifies the time and resources needed for its implementation. It generally includes an economic feasibility study to ensure the benefits provided by the proposed system outweigh the costs of its development. The main goal of system analysis is to specify complete and detailed requirements of the proposed system. This is accomplished by working closely with current and future system users and by careful study of existing manual or computerized systems (e.g., enterprise processes, legacy information systems). In addition to application requirements, this phase specifies other requirements such as performance, reliability, security, interfaces and more.

The design phase converts the description of system requirements into coherent, well-organized specifications that can be implemented through computer code. The design phase maps the "what" of requirements into the "how" of design specifications, enabling the implementation which follows. Design specifications include all aspects of the system, from databases to software module logic, input and output forms and reports. Design specifications are in turn converted into code, the latter of which is tested and installed in the implementation phase. In addition to coding, testing and installation, this phase includes other activities such as finalizing documentation, user training and system conversion. Finally, maintenance represents an ongoing activity that corrects system problems and adds new functionality as needed. In some sense, maintenance is not a separate phase but a repetition of the other life cycle phases required to analyze, design and implement the needed changes. Thus, as with reengineering and expert system life cycles, maintenance completes the cycle and often signals the beginning of a new analysis effort (e.g., a return to IS planning).

Summary. To summarize this examination of established approaches, reengineering, expert systems and information systems all involve some kind of developmental life cycle. Each respective life cycle begins with some planning and analytical tasks (e.g., reengineering process identification, expert system problem assessment, IS planning). The life cycles proceed through relatively diverse design activities (e.g., reengineering pathology diagnosis and transformation matching, expert system knowledge acquisition and representation, IS logical and physical design). Then each life cycle prescribes some physical and procedural changes through implementation (e.g., change management, V&V, coding) and transitions into a maintenance phase to complete the life cycle.

Despite structural similarities between the three life cycle models, however, the underlying steps and focuses of the corresponding methodologies are quite distinct. For instance, whereas reengineering is oriented toward the enterprise *process*, expert systems methodologies directly address *knowledge*, and IS methodologies focus on systems for *information* processing. Alternatively, one can argue information represents a necessary component of knowledge, which in turn represents a key element of any enterprise process. Thus, the three areas of methodological focus are tightly linked, and a strong argument can be made that all three aspects—process, knowledge and information—should be addressed together when designing knowledge systems and processes. This represents one of the central premises of the chapter.

KNOWLEDGE MANAGEMENT FEATURE SPACE

In this section, we outline a feature space of specific activities and stages comprising knowledge management as a process. We use this to classify and analyze a number of existing systems and practices, drawn principally from the literature, currently employed for knowledge management. The classification elucidates several informative similarities and differences between the diverse sets of systems and practices, and the analysis interrelates the various classes back to the three methodological approaches examined above (i.e., reengineering, expert systems and information systems). This represents a central contribution of the chapter, as it reveals the underlying process elements, levels and phases of knowledge management and links them to methods available for knowledge system and process development. We begin by drawing from the literature to integrate a number of various life cycle models emerging for managing knowledge.

Knowledge Management Life Cycle

Drawing from Nissen (1999), we begin to observe a sense of process flow or a life cycle associated with knowledge management. With some similarity to the developmental life cycles discussed above, although we describe the knowledge management life cycle as a sequence of activities, in practice their performance is generally iterative. Building upon this notion, we outline key elements of several life cycle models drawn from the recent knowledge management literature to develop an amalgamated, general knowledge management process model. We then combine this amalgamated model with other key dimensions and exemplars from the literature to compose a knowledge management feature space for analysis. Results from this analysis are used to make observations pertaining to the current state of the practice in knowledge management and integrate our discussion of extant system development approaches.

Life Cycle Models. In Table 1, we compare the knowledge management life cycles proposed by several researchers (e.g., Nissen 1999, Despres and Chauvel 1999, Gartner Group 1999, Davenport and Prusak 1998), which all share considerable similarities. For instance, most of the four life cycle models begin with a "create" or "generate" phase; only the Nissen model begins with knowledge capture, an activity appearing in the *third* phase of the Gartner Group model. The second phase pertains to the organization, mapping or bundling of knowledge; Davenport and Prusak omit this organization phase from their model, but it appears very prominently in all the others. Phase three uses different terms across the models, but they all address some mechanism for making knowledge formal or explicit. Likewise, the fourth phase uses different terms but addresses the ability to share or distribute knowledge in the enterprise. Three of the four models include a fifth phase for application or (re)use of knowledge for problem solving or decision making in the organization. Only the Despres and Chauvel model includes a sixth phase for knowledge evolution.

The Amalgamated model integrates the key concepts and terms from the four life cycle models. Comparing the steps above proposed by Nissen (1999) with this Amalgamated model, notice from Table 1 the latter life cycle model makes a distinction between knowledge creation (e.g., as proposed by Despres and Chauvel and Gartner Group) and its capture or formalization (i.e., Phase 3). Whereas knowledge creation involves discovery and the development of new knowledge, knowledge capture requires only that the knowledge be new to a particular individual or organization, and formalization involves the conversion of existing knowledge from tacit to explicit form. The Amalgamated model therefore seems more complete with its beginning at the creation step. Similarly, the Amalgamated model also adopts the evolution step from the Despres and Chauvel model.

These amalgamated life cycle phases are repeated across the tops of Tables 2-5 for reference. The cells of Tables 2-4 contain examples of current knowledge management systems and practices drawn from the literature (e.g., Davenport et al. 1996, Davenport and Prusak 1998, Gartner Group 1999, Despres and Chauvel 1999, others). We use these exemplars from current practice not only to populate the table cells but to interrelate its underlying dimensions. We have already discussed the life cycle dimension. But higher dimensionality may be required to map the more dynamic knowledge management activities summarized in Table 1. One important dimension along these lines is *knowledge management level*, which draws from Nonaka (1994) and others (e.g., Despres and Chauvel 1999). The knowledge management level includes both individual and collective entities, the latter of which are further distinguished between groups (e.g., of relatively small collections such as work teams or functional departments) and organizations (e.g., relatively large collections such as enterprises or corporations). This dimension pertains to the scale of knowledge management and extends from a single person, through work groups, to an enterprise as a whole. Combined with the life cycle steps from above, we employ these levels to classify extant knowledge management applications.

Table 2 in particular pertains to organization-level knowledge management, which we differentiate from that occurring at the group and individual levels. Tables 3 and 4 are presented to incorporate systems and practices applied at these latter knowledge management levels. In discussing

Table 1: Knowledge Management Life Cycle Models

Model	Phase 1	Phase 2	Phase 3	Phase 4	Phase 5	Phase 6
Nissen	Capture	Organize	Formalize	Distribute	Apply	
Despres and Chauvel	Create	Map/bundle	Store	Share/transfer	Reuse	Evolve
Gartner Group	Create	Organize	Capture	Access	Use	
Davenport & Prusak	Generate		Codify	Transfer		
Amalgamated	Create	Organize	Formalize	Distribute	Apply	Evolve

Table 2: Organization Level Systems and Practices

Create	Organize	Formalize	Distribute	Apply	Evolve
Data mining	Knowledge map	Data warehouse	FAQs	BPR	
AI first principles	Semantic network	Reports	Best practices		
R&D	GrapeVines		Lessons learned		
Bench marking			Knowledge brokers		
Business intel			"Yellow Pages"		
			Web publication		
			Document search		

these tables, a number of points merit noting. First, arguably, knowledge creation represents a more difficult and uncertain process than its capture. Indeed, referring to Tables 2-4, few systems exist to support knowledge creation—data mining system conglomerates and artificial intelligence (AI) from first principles represent notable exceptions—but a number of enterprise practices (e.g., corporate research and development (R&D), benchmarking, competitive business intelligence) are widely employed for this purpose.

Second, referring back to Table 1, the Gartner Group collects the organize, formalize and distribute activities under the common heading "knowledge sharing." We use this grouping below to help classify and cluster extant knowledge management technologies and practices. Continuing across the rows of Table 2, examples of systems used for enterprise-wide knowledge organization include knowledge maps and semantic networks. And from Table 3, group-level implementations such as Chrylser's Engineering Book of Knowledge and Anderson Consulting's Knowledge Exchange are also noted in the literature. Table 4 reveals that at the individual level, systems that extract and cluster information by keyword are available, along with the online thesaurus to interrelate key terms and concepts in the enterprise. And as noted above in our discussion of expert systems methodologies, knowledge-based systems (KBS; e.g., expert systems, intelligent agents) address knowledge directly and employ a variety of knowledge representational techniques for its organization. Without going into great detail, notice a number of systems and practices listed under the knowledge formalization and distribution columns. Clearly, this represents the current emphasis of most knowledge management today. Alternatively, the application phase is relatively sparse in terms of supporting systems.

Third, we noted above the Despres and Chauvel life cycle includes a sixth element, called "evolution," to represent the refinement and continued development of existing knowledge. With a little thought, one can see such refinement and continued development is similar in many respects to the "maintenance" phases of the developmental life cycle models above. And as with the former life cycle models—which we note are laid-out sequentially but generally performed in an iterative manner—we can also say that knowledge evolution leads in turn to further knowledge creation, thereby completing the cycle and signaling the beginning of new knowledge capture and sharing (e.g., additional organization, formalization and distribution of knowledge).

The cyclical nature is more readily discernable when presented as a circle, as opposed to a linear sequence of activities, as depicted in Figure 3. Notice the three "sharing" activities from above—knowledge organization, formalization and distribution—are adjacent on the right-hand side of the cycle. From the tables above, we see these activities correspond with greater support from extant information technologies and hence represent more of a localized view of knowledge management; thus, the grouping under the "Class I" heading in the figure. We note such, localized knowledge management systems are inherently supportive in nature; that is, this class of implementations to organize, formalize and distribute knowledge in the enterprise *support* people in the loop, who in turn apply, evolve and create knowledge in the organization.

Alternatively, the latter three, non-sharing activities are adjacent on the left-hand side of the cycle. But from the tables we see these activities do not correspond well with support from extant information technologies and hence represent an expanded view of knowledge management; thus, the grouping under the "Class II" heading in the figure. We note such, expanded knowledge management systems are inherently performative in nature; that is, this class of implementations to apply, evolve and create knowledge in the enterprise

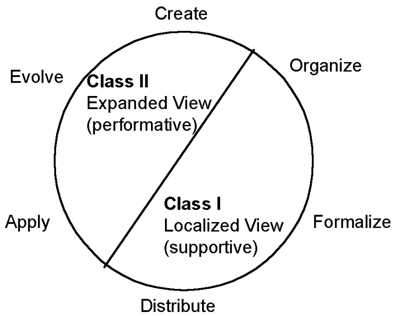
Table 3: Group Level Systems and Practices

Create	Organize	Formalize	Distribute	Apply	Evolve
	Engr BoK		Workflow	Group DSS	
	Knowledge Exchange		Groupware		
			Community of practice		
			Discussion groups		
			Document sharing		
			Workshops		
			Listservers		
			Tele conference		
			E-mail		
			Meetings		

Table 4: Individual Level Systems and Practices

Create	Organize	Formalize	Distribute	Apply	Evolve
	Keyword extract		Information retrieval	Data visualization	
	Online thesaurus		Document mgmt		
	KBS	KBS	KBS	KBS	

Figure 3: Knowledge Management Life Cycle



perform knowledge-management activities, either in conjunction with or in lieu of people in the organization. Referring back to the tables above, however, we note very few extant knowledge management systems currently capable of performing in this manner. This may highlight a promising area for future knowledge management research and system development.

Table 5 extends the classification of extant knowledge management systems and practices by identifying the principal information technologies (ITs) and redesign transformations used to enable systems and practices from the various classes and life cycle phases. Take the first technology listed under the "create" column, for instance. This entry corresponds with the data mining class of systems identified in Table 2 above. Here, Table 5 indicates most data mining applications are enabled by tools and techniques associated with sophisticated pattern matching. Similarly, AI first-principles reasoning is also listed under the "create" column in Table 2. Here, Table 5 indicates most first-principles applications are enabled through the kind of automated inference performed by KBS. As other instances from Tables 2 and 5, respectively, many corporate R&D processes are transformed through concurrent engineering methods, benchmarking is generally conducted through comparative process analysis, and the collection of most competitive business intelligence is predicated on business research. The other technologies and transformations follow accordingly for Tables 2-4 and should be self-explanatory to the likely reader of this chapter.

Notice, looking across the rows at the rightmost column, we identify one or more of the methodologies from above—reengineering (BPR), expert systems (ES) or information systems (IS)—used to develop each principal IT or redesign transformation. For instance, conventional IS methods are generally employed to develop association lists, and ES approaches are required for implementing most semantic nets. Several important points can be drawn from this table. First, notice the relative sparsity of applications outside the three "sharing" columns (i.e., beyond knowledge organization, formalization and distribution). At least as described in the literature, the great majority of knowledge management applications pertain to sharing knowledge that already exists in the enterprise. This supports our classification above in terms of a limited view of knowledge management, as it excludes knowledge creation, application and evolution, which are necessary to complete the life cycle. Indeed, we omit the "evolve" column in this table, because a dearth of technologies and practices is presently available for knowledge management at this step of the life cycle.

Second, every principal IT and transformation used to enable knowledge management systems and practices (i.e., listed in the table) is addressed by at least one current methodology. In other words, our current set of BPR, ES and IS methodologies provides the necessary capabilities to design and develop this entire collection of technologies and transformations used to enable extant knowledge management systems and processes. This supports our premise that

Table 5: Principal Enabling ITs and Transformations

Create	Organize	Formalize	Distribute	Apply	Developmental Approach
Pattern matching					IS, ES
Automatic inference	Automatic inference	Automatic inference	Automatic inference	Automatic inference	ES
Concurrent engr					IS, BPR
Process analysis					BPR
Business research					
	Association lists				IS
	Semantic net				ES
	Database	Database	Database		IS
	Web/Notes	Web/Notes	Web/Notes		IS
	Text search	Text index			IS
			Search engine		IS
			Workflow		IS, BPR
			Groupware		IS
			VTC		IS
			Listserver		IS
			GDSS		IS
			Graphics		IS

extant methodologies, techniques and tools can be employed to develop knowledge management systems.

Third, a number of principal ITs and transformations require *multiple* developmental approaches. For instance, sophisticated pattern-matching tools and techniques employed to develop data mining applications require a synthesis of IS and ES approaches, and transformation of an R&D process to support concurrent engineering involves both information systems and process redesign methodologies and techniques. Indeed, where certain technologies and transformations are developed *together* (e.g., combining concurrent engineering with workflow technology), data from Table 5 indicate all three developmental approaches (i.e., reengineering, expert systems and information systems) are required. The table thus suggests that no single methodology—or pair of methodologies—and related set of techniques and tools is sufficient to develop all knowledge management systems and processes. This supports our premise that an integrated approach to knowledge system and process design is necessary. We address this need in the following section, as we explicitly integrate a number of critical contextual factors into the analysis.

CONTEXTUAL INTEGRATION

The feature space of systems and technologies outlined above defines the broad design space for KM systems. This design space is further defined and constrained by a set of contextual factors that impinge on the implementation of these systems in organizations. We identify and elaborate on two primary factors: 1) the organization, and 2) the nature of knowledge underlying the task.

The Organization

In addressing contextual factors associated with the organization, we divide the discussion into three parts: 1) the role of organizational memory, 2) structure of the organization, and 3) organizational incentives. We discuss each in turn.

The role of organizational memory. Organizations have memory systems that maintain the lessons learned from experience (Levitt and March 1988). The knowledge resident in an organization's memory constitutes a core mechanism for preserving its history, routines and the lessons learned over time. An organization relies on its memory for maintaining continuity in a changing environment. The elements of an organization's memory are recorded in many ways, formally as well as informally. Formal recording is implemented through artifacts such as memos, reports, files, standard operating procedures and rule books. Informal mechanisms for organizational memory are constituted by individuals and communities of practice (Brown and Duguid 1991).

The design of knowledge management systems requires an awareness of three aspects of organizational memory. First, not everything that occurs in an organization's life is recorded. Since the amount of experience encountered by an organization is vast, its retention must be necessarily selective. Often, an organization's capacity for sense making and retention imposes a significant constraint toward retaining even knowledge that is vital to its survival and continuity (Weick 1995). Besides, some of the experiences are ambiguous wherein the link between action and outcome is not clear (Olsen and March 1975). In such situations, there is a lack of clarity about the value and content of the lessons learned, raising questions about the necessity of retaining such knowledge. The challenge for designers of knowledge management systems lies in devising ways to augment the existing repositories of organizational memory while at the same time ensuring that the additional knowledge captured is valuable to the organization. Tools such as "yellow pages" and knowledge maps can contribute significantly in this direction.

Second, it needs to be recognized that knowledge captured through informal mechanisms is often richer and more important to the organization than what is stored through formal mechanisms (Weick 1995). Such knowledge, which is often created outside the realm of institutionally mandated methods and procedures, can dissipate from organizational memory because of factors such as corporate reorganization, turnover in personnel and changes in technology (Brown and Duguid 1991). The loss of this knowledge can result in organizational de-skilling (e.g., through actions such as corporate downsizing). And such knowledge, once lost, cannot be re-created easily (Hutchins 1991). The key to building useful knowledge management systems lies in creating solutions that capture the essence of informal knowledge in a way that can be preserved and maintained. Tools such as lessons learned, handbooks of engineering knowledge and knowledge exchanges can perform valuable functions in capturing and distributing informal knowledge.

Finally, the putative effectiveness of knowledge management systems is predicated on the extent to which they are integrated with the nature and content of organizational memory and the underlying work practices. This requirement can be translated into a number of design desiderata. First, in building the conduits for *extracting* knowledge from resident organizational memory or *feeding* into it, designers need to consider whether the knowledge is resident in the formal or informal domains of memory. Often, knowledge management systems are designed to interact only with the formal repositories of organizational memory, in the process neglecting key aspects of knowledge resident in the informal repositories. This can result in a diminution in the actual and perceived relevance and utility of the systems. Second, knowledge management systems need to be integrated within the existing work practices of the organization and its repertoire of tools. Drawing from experiences with groupware (Grudin 1996), it can be argued that unless knowledge management systems are embedded within the larger context of organizational work practices, they are unlikely to be used. The absence of continued interaction can result in atrophy and obsolescence.

Structure of the organization. The structure of the organization has important implications for the creation, retention and dissemination of knowledge. As stated earlier, conventional organization structures rely heavily on informal networks and communities of practice for storing and disseminating knowledge. This is particularly the case for tacit and subtle aspects of organizational work practices (Teece 1996). Increasingly, however, organizational activities are being executed in the context of modified organizational forms enabled by information technology, such as virtual and networked organizations (Chesbrough and Teece 1996, Fulk and DeSanctis 1995). These organizations, which are typically *ad hoc* in nature, are formed for accomplishing specific objectives by combining multiple functional entities, often from two or more formal (i.e., non-virtual) organizations. Once the objective is accomplished, the virtual organization is disbanded or merged with a more-traditional organization.

The use of virtual organizations can create two specific problems with respect to the creation and retention of knowledge (Sengupta and Ramesh 1999). The *ad hoc* nature of a virtual organization means that it does not have an established structure or well-developed norms. The members are not familiar with each other, and the task at its initial stages has a high degree of equivocality. Therefore, prior to addressing the task at hand, the organizational participants must first develop a structure as well as routines and channels of communication (Finholt et al. 1990). Second, because of their transient nature, virtual organizations experience difficulties in maintaining continuity; thus, the history of relevant corporate knowledge is not easily available. The problem is perpetuated when the virtual organization is disbanded: the organizational memory is once again lost for future use (Chesbrough and Teece 1996).

These distinctions in the properties of conventional and virtual organizations can translate to differences in the use and efficacy of knowledge management tools. To illustrate, consider the use of video teleconferencing (VTC). Conventional organizations are usually characterized by clearly established patterns of structure, communication, culture and norms (Carley 1991). In such organizations, the principal use of VTC lies in facilitating *task-oriented* discussions among individuals who happen to be geographically dispersed. In virtual organizations, on the other hand, these properties of the organization must first be developed. Consequently, the utility of a VTC application in virtual organizations lies not only in aiding task-oriented activities, but also in enabling the more basic activity of *organization-building*.

This distinction-in-use, contingent on organizational structure, can be found in knowledge management technologies across the board. For example, when used in virtual organizations, template-oriented information dissemination tools such as “yellow pages” and knowledge maps serve an additional purpose of providing a scaffold of corporate knowledge around which the new organization can be built (Sengupta and Zhao 1998). This context-dependent use is often characterized as *bricolage* (Levi-Strauss 1966; e.g., people make do with what is available and use tools in ways not originally envisaged by the designer). Similarly, drawing from adaptive structuration theory (DeSanctis and Poole 1994), we argue the use of knowledge management tools can be “faithful” or “ironic”, depending on—among other factors—the organizational structure.

Organizational Incentives. In a knowledge management activity, the retention and updating of knowledge is essentially an upstream activity, whereas the benefits from its dissemination and use occur downstream (Davenport and Prusak 1996, Grudin 1996). This dichotomy in capture and use can create specific problems. Organizational realities dictate that high proportions of projects or ventures are terminated before they are completed. The decision to continue or terminate is often made on the basis of managerial factors, particularly cost and schedule. Activities such as the capture of knowledge can add to the cost of a project, contribute to its delay and thus increase its prospects for cancellation. Moreover, the benefits of the knowledge accrue only at a later point and typically to a different stakeholder.

The implication for developing tools and processes for knowledge management is that it is insufficient to focus on the technical aspects of application development. Once the technologies are developed, managers may not have adequate incentives to justify devoting resources to the retention and updating of knowledge. Since the utility of knowledge management systems depends on their content, such systems can only succeed if appropriate incentives are provided (e.g., modifying the evaluation procedures for ongoing projects).

The Nature of Knowledge Underlying the Task

Organizations frequently resort to codifying, or making explicit, the processes underlying tasks. The utility of codification is thought to be two-fold (Orr 1990). First, it is a convenient mechanism for the capture and transfer of useful knowledge resident in the organization (Chesbrough and Teece 1996). Second, it serves as a way of ensuring quality by prescribing uniformity in procedures and practices. Such *canonical* practices are usually contained in devices such as manuals, training programs, organization charts, job descriptions and standard operating procedures.

However, canonical prescriptions are often inadequate for solving complex problems (Bourdieu 1977). This inadequacy is primarily caused by three factors (Lave 1988, Lave and Wenger 1990). First, the amount of effort involved in the acquisition of knowledge required for performing a complex task effectively is non-trivial. Second, knowledge that is acquired and represented in the canonical methods can become obsolete with changes in the underlying technologies (Suchman 1987). Finally, there often are several different ways in which a task can be performed, and the best way of performing a task is contingent on the context in which it is performed.

Due to inadequacies in canonical descriptions, individuals who perform complex tasks may exercise improvisational skills and resort to *non-canonical* practices in order to bridge the gap between canonical approaches and effective work practices (Zuboff 1988). As a result, the manner in which complex problems are actually solved often differs from the prescriptive methods, in at least two ways. First, non-canonical practices can serve as extensions of canonical methods (e.g., in compensating for the omissions and inadequacies in prescriptive knowledge). Second, such practices can also modify canonical methods. This can happen if the canonical methods are considered to be obsolete or generally less effective than the corresponding non-canonical methods.

The interplay of prescription and practice results in the creation of a significant body of non-canonical knowledge in organizations. This knowledge is typically embedded in the relevant communities of practice. The knowledge can take various forms, such as vocabularies, knowledge of practical constraints and workarounds relating to specific constraints (Brown and Duguid 1991). The knowledge resides informally, often in the form of narratives such as “war stories” (Orr 1990) or simply in the heads of designers, engineers and managers (Lynn 1992). Such tacit knowledge, which is resistant to efforts at codification (Orr 1990), often forms the core of an organization’s distinctive competence (Chesbrough and Teece 1996) and is preserved and nurtured through communities of practice.

The principal implication of this discussion is, in developing knowledge management systems, a designer is faced with constraints on the types of knowledge that are amenable to capture. While the corpus of explicit knowledge that can be readily codified (and thus captured and distributed through systems) is undoubtedly useful, it is often accompanied by a vast complement of tacit knowledge, which is difficult to capture. This constraint is brought into focus by a recognition that in many instances (as in product design situations), it is the tacit knowledge that constitutes the mother-lode in terms of value.

As a practical matter, the line between explicit and tacit is blurred; often, the distinction is reflected in the way the knowledge is represented (Buckingham Shum 1994). Formal representations (e.g., in terms of design rules and procedures) lend themselves well to codification, but are difficult to capture. Informal representations (e.g., a videorecording of a design session) do not lend themselves to automated reasoning. However, such representations make the capture of tacit knowledge more feasible. Further, appropriately devised indexing schemes can make context-sensitive retrieval possible.

KNOWLEDGE MANAGEMENT EXAMPLE

To summarize from above, the contextual factors discussed in the preceding section play a critical contingency role in our examination of established approaches; that is, the context of the knowledge management environment and situation has a strong mediating effect on selection of appropriate methodologies, techniques and tools for knowledge system and process development. We illustrate this effect through examination of an example of a knowledge-work process from the literature to articulate, at a high level, our integrated approach to knowledge system and process design. The general example described in this section represents a composite of workflow activities adapted from two extremes of a process family. On the one hand, we have the well-known and relatively well-structured credit financing process described by Hammer and Champy (1993, pp. 36-39). This process is representative of many associated with knowledge and information work and should be familiar to most readers. On the other hand, we also describe its semi-structured counterpart that applies to venture-capital financing of startup technology companies. This latter process depicts the kind of difficulties that often arise with non-routine tasks and helps accentuate both benefits and limitations of our integrated approach to knowledge system and process development. We first outline key aspects of the general process and then discuss it in the context of integrated knowledge system and process development.

General Process

Credit financing represents a key subprocess in support of marketing and sales, as the ability to provide potential customers with in-house financing represents a strong selling point for any company. However, customer feedback suggests the process as practiced in many firms has a number of shortcomings and flaws. For example, it is often associated with long cycle time required to prepare a credit financing package, and many firms lack the ability to report on the status of a particular package while it is being processed. This holds for either routine credit packages (e.g., financing a washing machine) as well as non-routine processes (e.g., syndicating a commercial real estate loan).

Structured process. Drawing from Hammer and Champy (1993, pp. 36-39), we can characterize the general process through involvement of four value stream participants: 1) field sales groups with representatives that work to secure new customers, 2) the centralized credit financing organization, 3) a third party delivery company, and 4) the customers themselves. For purpose of this example, say the centralized credit financing unit is organized in terms of four functional departments, each of which is staffed with specialists for the functional areas: 1) credit check, 2) terms development, 3) financial pricing, and 4) quotation packaging.

As a relatively classic, bureaucratic organization, one can understand the process flow is often described as serial, beginning with a meeting between the field sales representative and potential customer. For instance, when customer interest is generated, the process continues with a telephone call from the field sales representative to a contact person in the financing unit, the latter of whom writes-down the relevant customer, product and financing information. The paper with this information is then carried to each separate functional department, where a functional manager assigns the job to a specialist from the department. This assignment is accomplished simply by placing the paper in the specialist's in-box. The specialist in each functional department retrieves the paper from his or her in-box and performs the functional tasks required for each potential customer. This functional work is accomplished using specialized, standalone computer equipment, but communications are conducted entirely through paper and face-to-face meetings. Once the specialist completes the required functional tasks, he or she writes-down the relevant facts and determinations on a separate piece of paper and reviews the results with the department manager.

This kind of functional activity proceeds in series from one department to another until the credit financing package is complete. When complete, the package is reviewed by the unit manager and then carried back to the contact representative, who arranges to have the third party delivery company transport the package to the field sales representative, generally via overnight air service. Once received, the field sales representative schedules an appointment with the potential customer to discuss the financing and other terms of the potential contract. Quality feedback loops at each department indicate packages can be returned for rework at any stage of the process, including return by the salesperson in the field.

Semi-structured process. The syndicated loan (e.g., for a startup company) proceeds through the same basic steps, but it is far less structured in nature; that is, although the same fundamental tasks need to be completed, the performance of each task may differ depending on the nature of the financing (e.g., amount of loan, type of loan, collateral offered), potential customer (e.g., length of customer relationship, size of customer institution, operating experience) and business proposition (e.g., stability of proposed venture, inherent technological risk, novelty of business model). Where the financing unit has underwritten one or more loans for similar purposes, the process begins with a search for a representative model to use as a starting point for customizing a loan package. This is a manner of case-based reasoning, which generally involves searching through archived documents and asking colleagues about their experiences. Where the requested financing package represents an unprecedented mix of attributes (e.g., financing, customer, proposition), workers from various fields of expertise (e.g., credit determination, contracts, risk assessment, pricing, loan packaging) must meet to determine how to approach the problem. And in the case in which the requested financing is routine, the process essentially collapses into the one described above, through which a package progresses in serial fashion from functional group to group.

Table 6: Process Measurements and Diagnoses

Configuration Measure	Value	Diagnosis
Parallelism	1.00*	Sequential process flows
Handoffs fraction	1.00	Process friction
Feedback fraction	0.29	Checking & complexity
IT support fraction	0.24	Manual process
IT communication fraction	0.00*	Paper-based process
IT automation fraction	0.00*	Labor-intensive process

* denotes theoretical extremum for a measure

Process Analysis

The first stage of knowledge system and process design involves process analysis. Until one understands the process—along with its various redesign opportunities and required knowledge—it makes little sense to begin designing systems. As noted above, many methodologies have been developed for process design. Here, we discuss the measurement-driven redesign method of Nissen (1998), which has been implemented via expert systems technology to automatically diagnose process pathologies and recommend redesign transformations.

Structured process. The pathologies and corresponding process measurements are shown in Table 6 for the first, structured credit financing process. From measured values presented in the table, one can see the baseline process suffers from a number of serious pathologies. Beginning with the parallelism measurement (1.00), this quantifies the sequential nature of the process and has adverse implications in terms of cycle time. The handoffs fraction measures the relative number of exchanges between workers performing different roles in the process. And the value (1.00) obtained for this process is exceptionally high for a process involving knowledge and information work such as this—on average, a specialist worker (e.g., credit manager, credit analyst, terms manager) performs only a *single* activity before passing work

Table 7: Redesign Transformations

Pathology	Transformation
Sequential process flows	De-linearize
Checking & complexity	Asynchronous reviews or empowerment
Process friction	Case manager or case team
Manual process	Integrated databases or workflow
Paper-based process	e-mail or workflow
Labor-intensive process	Expert systems or intelligent agents

along to the next process step. The associated pathology, process friction, also has adverse implications in term of cycle time, as work sitting in in-boxes and out-boxes, awaiting assignment, pausing for review and approval, undergoing transport and like situations consumes substantial process time. Closely related is the feedback fraction measure, which quantifies the relative number of review and approval steps in a process. The measured value (0.29) is relatively high for a knowledge-work process such as this, as a separate review/approval step is conducted at approximately every third activity. The associated pathology pertains to checking and complexity, which has adverse implications in terms of both cost and cycle time, and reveals relatively little autonomy for the knowledge workers involved in the process.

The three IT fractions are used to measure the relative use of information technology for support, communication and automation, respectively. The relatively low value (0.24) obtained for IT support indicates only one in four process activities is supported by information technology. And notice both the IT communication and automation measurements reflect theoretical minima. The associated pathologies listed in the table (i.e., a manual, paper-based, labor-intensive process) have adverse implications in terms of both cost and cycle time. Moreover, from a knowledge management perspective, knowledge in this baseline process is both tacit and explicit, resides in the minds of specialist workers as well as formal procedures, and where shared at all, it involves paper and face-to-face conversations. This represents a very primitive knowledge management environment.

Semi-structured process. The semi-structured process involves many of the same activities. And as noted above, it collapses into its structured process counterpart for routine financing requests. Indeed, at the process level, the activities are identical, except for the number of exceptions that affect the semi-structured version and various modes of problem solving used to perform different process activities. For example, the process proceeds along one flow if a prior financing package can be located for use as an exemplar and an entirely different one where such an exemplar cannot be found. Similarly, task performance is quite equivalent to that in the simple process above where specialists can work in relative isolation, but substantial face-to-face interaction is required where collaborative problem solving is required. For ease of discussion and to illustrate our techniques, we focus on the simpler, structured process in the analysis below. But for richness, we weave-in variations and differences to include its more complex, semi-structured counterpart.

Process redesign. Some representative redesign transformations are summarized for the structured process in Table 7. Beginning with the first pathology listed in the table, redesign transformations involving de-linearization (i.e., performing two or more process activities in parallel) offer good potential to treat this sequential process. Notice these transformations are not mutually exclusive, as de-linearization can also be applied to address the checking and complexity pathology through asynchronous reviews. This is an alternative to empowering analysts in the process to review their own work, fundamentally a TQM idea of building-in quality rather than verifying it through inspection.

The process friction pathology can be addressed through a case manager or case team. In such a transformation, the specialized division of labor and functional organization currently exhibited in the process are dissolved and replaced by small team—or even a single individual if sufficient knowledge support and expertise can be provided—that performs all the process activities. Manual processes can be addressed by a multitude of information technology transformations. Integrated databases and workflow systems represent good candidates here, and e-mail or workflow can also address the paper-based communications. Regarding the labor-intensive process resulting from negligible automation, expert systems or intelligent agents offer good potential to address these problems, provided the necessary knowledge and expertise can be effectively organized and formalized. Clearly, other redesign transformations also offer potential, and the point is not to exhaustively cover them. Rather, we want to show the importance of addressing the process in conjunction with knowledge and systems.

Additionally, many of these same redesign transformations also apply to the more complex, semi-structured process. The key difference is they apply to different versions or modalities of the latter process. For instance, we noted above the semi-structured process collapses into its structured counterpart for routine credit requests. In such a case, each of the redesign transformations listed in Table 7 also applies here. Alternatively, where more collaboration is required among workers, redesigns such as de-linearization and workflow do not apply as well. However, in this latter case, the group may effectively perform as a case team, essentially transforming itself in this manner as a separate mode of operation. Thus, alternative modalities of execution in the semi-structured process indicate the simultaneous existence of *multiple* redesign alternatives, each of which is instantiated at different times and occasions on the basis of contextual factors (e.g., task familiarity). And to the extent this latter case involves manual, paper-based, labor-intensive process activities—even though collaboratively performed—there exists good opportunity for the kinds of IT-based redesigns noted in Table 7 for the simpler process version above.

Knowledge Analysis

Knowledge analysis is in no way independent of the process analysis above. Rather, the former is fed directly by results of the latter. Looking at the redesign transformations identified in Table 7, for example, nearly all of them address knowledge in some way. For one, to effectively de-linearize process activities, workers such as the field sales agents need to know where work on the various elements of a credit financing packing is in the process. We can express this in terms of our knowledge management feature space above. From the amalgamated knowledge management life cycle model, this involves distributing explicit knowledge at the organization level across the credit unit. As another example, if line workers (i.e., not managers) are empowered to review and approve their own work, they require the kind of knowledge and experience possessed by managers today. From the knowledge management life cycle, this involves formalizing tacit knowledge at the individual level (e.g., the functional managers' expertise) and its subsequent distribution through each functional group.

The case team involves similar knowledge formalization, as generalist workers on a team require access to the detailed knowledge currently possessed by specialists in each functional organization. To work effectively as a team, the group will first need to organize this knowledge, and subsequently it will be distributed across the team members in the group. The case manager (i.e., a single individual) represents the extreme instantiation of a case team and places the most demands in terms of managing knowledge. Say, for instance, this single individual is the field sales agent; that is, instead of relying upon the centralized credit financing organization, field sales agents would become responsible for preparing their own credit packages. Deferring questions of technological feasibility and cognitive limitations for the time being, clearly, the field sales agents would need to be able to apply all the knowledge and reason with all the experience currently employed within the centralized credit organization; otherwise, performance will suffer (e.g., bad loans may increase or promising financing opportunities may be missed by mistake). Referring back to the knowledge management life cycle, no new knowledge needs to be created here, but existing—tacit and explicit—credit financing knowledge needs to be organized for understanding, formalized to be made explicit and distributed for remote application by sales agents in the field. Aside from the creation and evolution steps, this covers the entire knowledge management life cycle and involves knowledge from the level of an individual, through the functional groups, to the entire enterprise.

Contextual Analysis

Here, we draw from the contextual integration section to further refine the analysis of this knowledge management process and associated systems. Considering first the role of organizational memory in the credit financing process, we can identify two principal mechanisms for its recording. Explicit knowledge is well represented through standard operating procedures—though predominately in paper form. And tacit knowledge, which is required to handle novel, complex or unusual financing requests, resides in the minds of each specialist worker and is recorded through communities of practice for each of the four functional specialties (e.g., credit, contracts, pricing, risk).

Drawing from the discussion above, tools such as "yellow pages" and knowledge maps would be appropriate here to formalize and distribute the location and content of knowledge among all the functional workers. Clearly, this would apply more to redesigns such as de-linearization that preserve the functional organization of the process or to virtual coordination of workers who have not previously worked together. But such tools could also be made available to a single case manager, who may need expert assistance with particularly difficult problems. In addition to identifying *who* in the organization possesses various kinds of knowledge (e.g., through the "yellow pages") and *what* kinds of knowledge are possessed (e.g., through knowledge maps), it is also important to begin capturing the corresponding knowledge itself, particularly the tacit knowledge used by specialists to solve difficult problems. "Yellow pages" serve little purpose if the person identified as resident expert is no longer part of the organization (e.g., the de-skilling referred to above). This is an application for which expert systems may be relatively well-suited to make such tacit knowledge explicit and distributable. Other tools, such as lessons learned, handbooks and knowledge exchanges, can also serve a useful purpose by reducing the formalization burden often associated with expert systems.

Recalling the design desiderata from above, we have noted and addressed both tacit and explicit knowledge in the process and the need to develop conduits for feeding and extracting informal as well as formal sources of knowledge in the process. And by redesigning the process before designing knowledge management systems, we address the second design rule arguing such systems must be embedded into the underlying process as well. Here we begin to re-incorporate the methodologies-techniques-tools discussion from above.

Regarding organizational structure, most of the aforementioned redesigns preserve the relatively conventional organizational structure of the baseline credit unit. Although the case manager redesign is radical in terms of shifting responsibility from the organizational level (i.e., the credit unit) to an individual role (i.e., the field sales agent), even this streamlined organization is not particularly virtual in nature. Hence we contrast even the case manager redesign with the kinds of virtual design teams discussed by Sengupta and Ramesh (1999), for instance, and note how the semi-structured credit financing process from above can easily take-on a number of virtual characteristics (e.g., geographically-distributed coworkers, new teams formed for each syndicated loan). When considering VTC, for instance, use of this technology—say by field sales agents needing to interact with functional experts—can focus on task-oriented discussions. Alternatively, where new firms are coming together for the first time to syndicate financing for an Internet startup company, the additional demands and complexities of organization-building would *also* apply in this more complex, virtual case.

This brings us to the issue of incentives. Continuing with the case manager redesign, for instance, the field sales agent clearly represents a consumer of process knowledge and is highly motivated to seek it out. But say that (human) functional specialists are retained in a "help desk" approach to provide such assistance, among other duties. They must be incentivized, not only to provide the requested assistance, but to also be responsive, accurate and thorough in support of field sales agents' questions and problems. This is much more of a human relations issue than the technical questions traditionally addressed by information systems methodologies. But the example should make it clear how important such human relations issues (e.g., compensation, team building) can become in the context of knowledge system and process design. Moreover, say we are interested in capturing such specialist knowledge and experience through expert system technology. Here in particular, specialists require strong incentives to contribute knowledge. For aside from maintaining the resulting expert systems, perhaps, such specialists may effectively be replaced by the knowledge systems they help develop.

The second key contextual factor involves the nature of knowledge underlying each task. Beginning with the kinds of canonical practices from above—which we note are usually contained in devices such as manuals, training programs, organization charts, job descriptions and standard operating procedures—let's presume such practices are formally documented in the structured process. But drawing from the listed pathologies for the process (esp. manual, paper-based, labor-intensive), they are unlikely to exist in digital form. Tools to organize, capture and distribute such explicit knowledge include databases, online textual search and retrieval systems, hyperlinked intranet applications and even workflow integration (e.g., through contextualized online help and process information).

However, the non-canonical knowledge is considerably more difficult to manage. Where such tacit knowledge can be identified and articulated, conceivably expert system applications can be employed for its formalization and network technologies used for distribution. The kinds of experience-based knowledge used by managers to review the work of functional subordinates in the structured process, for instance, may fall into this category. Still, as noted above, the knowledge engineering required to develop and maintain such intelligent systems is non-trivial in terms of level and amount. Other kinds of tacit knowledge may be more difficult to identify, and even harder to articulate. Take, for example, problem solving knowledge used by functional specialists—perhaps in a virtual organization—to address novel, complex financing issues. A specialist may not even be aware of the corresponding knowledge until the situation arises, and many experts and professionals are notoriously poor at articulating the manner in which such non-routine problems are solved. For this kind of knowledge, we may have to settle for something of an "80/20" rule. In such a rule, one would strive to capture, formalize and distribute knowledge associated with the 80% of problems that are relatively routine and perfunctory through technology (e.g., expert systems). For the rest, one would instead rely upon the kinds of "yellow pages," knowledge maps, help desks and VTC links required to make such knowledge available among human problem solvers, in either a classical or virtual organization.

To summarize the section thus far, the methodology for designing knowledge systems and processes has progressed through three steps: 1) process analysis and redesign, 2) knowledge analysis, and 3) contextual analysis. Through examination of a general credit financing process, we have identified a number of alternative systems that offer potential to enhance knowledge management in this process context. These include, for instance, "yellow pages," knowledge maps, expert systems, VTC, databases, workflow, textual search and retrieval, intranet and other classes of systems and applications. Returning to one of our original themes in the paper, *individually*, each of these technological artifacts represents a relatively well-known and understood entity, for which extant methodologies, techniques and tools—used for reengineering, expert systems and information systems design—are readily available *at the application level*. In the discussion below, we close the section by briefly illustrating the kinds of methodologies, techniques and tools from above that can be employed to develop some of the knowledge management applications identified through this multi-level (e.g., process, knowledge, contextual) analysis.

Systems Analysis and Integration

At this final stage of analysis, we fix the discussion by focusing on one of the several redesign alternatives from above: case team. As noted above, this redesign is radical in nature—certainly with respect to the baseline, departmentalized organization—and places extreme demands on the designer in terms of knowledge management. The reader may recall the case team also corresponds to one modality associated with the semi-structured credit financing process. Through examination of the corresponding knowledge and contextual factors from above, we can identify several technologies and applications for integration with the case team process.

For instance, case team members need means for formalizing and distributing explicit knowledge (e.g., manuals, procedures, instructions), at the group and individual levels, and applications to help coordinate their respective activities. Drawing from associated group and individual level technologies above (e.g., Tables 3 and 4), knowledge exchange, workflow, groupware, document sharing, e-mail and text-retrieval applications can be implemented across the corresponding phases of the knowledge management life cycle. Individually, each of these applications is common and relatively straightforward to design and implement through conventional IS methodologies, techniques and tools. One could, for instance, iteratively employ the SDLC for this entire set of applications, perhaps using standard IS developmental techniques such as interviews, (e.g., for requirements determination and usage patterns), data-flow modeling (e.g., for mapping process information flows), entity-relationship modeling (e.g., for database design) and structured programming (e.g., for implementation). Associated tools could in turn include structured interview templates, a suite of modeling applications (e.g., as part of a CASE tool), program editors, compilers and debuggers.

Were we to select an alternative process redesign—such as the field sales agent case manager—involving separate knowledge (e.g., tacit managerial and specialist knowledge) and contextual factors (e.g., formal and informal organizational memory requirements, canonical and non-canonical tasks), the systems analysis and integration would necessarily focus on a different set of knowledge management applications (e.g., expert systems, intranets, VTC, others). Nonetheless, methodologies, techniques and tools that are readily available and widely understood can be compared and employed in a straightforward manner. For instance, the applications identified for this latter redesign might integrate expert system development methods with the information system analysis and design techniques and tools from above.

Thus, having reached this level of analysis, one can see traditional, well-understood IS methodologies, techniques and tools can be employed for knowledge management system development. This answers one of the key questions posed in the introductory section. The key to our integrated knowledge management methodology is, such systems are explicitly analyzed, selected and combined to help manage *knowledge* (e.g., explicit work practices), for a particular *process* design (e.g., case team) and set of *contextual factors* (e.g., organizational memory involving canonical knowledge). Most existing methodologies simply begin at this (final) system-development step without consideration of such process, knowledge or contextual factors. Only empirical evidence can confirm that our integrated methodology is in some ways *superior* to extant developmental approaches. But we can certainly argue this integrated knowledge system and process design methodology is more *complete*. This leads us to a number of conclusions and suggestions for future research along these lines.

CONCLUSIONS AND FUTURE RESEARCH

The research described in this chapter has focused on knowledge management and system design from three integrated perspectives: 1) reengineering process innovation, 2) expert systems knowledge acquisition and representation, and 3) information systems analysis and design. Through careful analysis and discussion, we integrated these three perspectives in a systematic manner, beginning with analysis and design of the enterprise process of interest, progressively moving into knowledge capture and formalization, and then system design and implementation. Thus, we have developed an integrated approach that covers the gamut of design considerations from the enterprise process in the large, through alternative classes of knowledge in the middle, and on to specific systems in the detail.

The central premise of this work is, although knowledge management represents a phenomenon of relatively new widespread interest and attention in research and practice, many of its underlying elements are actually quite familiar and have been effectively addressed for many years through work in information systems, artificial intelligence and process redesign. In the course of our discussion above, we outlined and employed a three-tier framework for examining alternative methodologies, techniques and tools. We then outlined a contingent feature space of specific elements, levels and stages comprising knowledge management, using it to compose an integrated analysis and design approach tailored to each of its key aspects. With this, we developed a set of contextual factors that draw insight into strengths and limitations of various approaches, and we illustrated the use and utility of integrated knowledge system and process design through an example. This represents a central contribution of the chapter, as it reveals the underlying components of knowledge management and prescribes design guidance specific to each.

Through our examination of established approaches, reengineering, expert systems and information systems all involve some kind of developmental life cycle. Each respective life cycle begins with some planning and analytical tasks, proceeds through relatively diverse design activities, prescribes some physical and procedural changes through implementation, and transitions into a maintenance phase. Despite structural similarities between the three life cycle models, however, the underlying steps and focuses of the corresponding methodologies are quite distinct. Yet, the three areas of methodological focus are tightly linked, and a strong argument can be made that all three aspects—process, knowledge and information—should be addressed together when designing knowledge systems and processes. This represents one of the central premises of the chapter.

A number of other important findings and conclusions emerge from this research. For one, despite the abundance of knowledge management life cycles that now appear in the literature, they all share considerable similarities and can be integrated into an amalgamated model to describe a broad diversity of knowledge management work in the enterprise. As another, if harnessed appropriately, the current repertoire of IT methodologies, technologies and tools has much to offer for the design and development of knowledge management systems. At the same time, as Tables 2-4 show, there are aspects of the knowledge management life cycle where the existing capabilities of IT are inadequate, most notably in the generation and application of knowledge at the organizational and group levels.

Further, contextual factors play a critical role in the design and implementation of knowledge systems and processes. We contend that effective knowledge management is a question of tailoring technical and process solutions to fit the exigencies of the context in which activities are being performed. And we illustrate how our integrated methodology for knowledge system and process design is more complete than existing developmental approaches, the latter of which simply begin at the (final) system development phase and ignore key process, knowledge and contextual factors. Future research along these lines may produce empirical evidence that our integrated method is also superior in some respects to extant approaches.

The research described in this chapter offers several other logical extensions for future research. One important extension would develop IT solutions that address support and performance of knowledge generation and application activities along the life cycle. Recall we found applications for such activities to be largely absent from tables representing the current state of technologies and practices. We may also identify the need for new techniques and tools to support development of the new solutions. Another useful extension would delineate the *contingent* nature of knowledge management. For instance, one could endeavor to specify in greater detail the interaction between information technologies and practices with organizational activities, thereby enabling designers to identify "families" of solutions that are likely to succeed as knowledge system and process implementations. Clearly, this represents only a short, partial list of future research topics. Knowledge management remains a relatively novel focus of research, and much work needs to be accomplished to advance our knowledge and technological level in this area. We hope to have contributed to such knowledge and level by illustrating how current methodologies, techniques and tools can be applied, in an integrated manner, for analysis and design of knowledge systems and processes.

REFERENCES

Anderson and APQC (1996). *The Knowledge Management Assessment Tool: External Benchmarking Version.*

Anderson, J.R. (1983). *The Architecture of Cognition*. Harvard University Press: Cambridge.

Andrews, D.C. and Stalick, S.K. (1994). *Business Reengineering: the Survival Guide* Yourdon Press Computing Series.

Arrow, K. (1962). "Economic Welfare and the Allocation of Resources for Invention," in: R. Nelson (Ed.), *The Rate and Direction of Inventive Activity*. Princeton University Press: Princeton, NJ.

Bashein, B.J., Markus, M.L., and Riley, P. (1994). "Preconditions for BPR success: and how to prevent failures," *Information Systems Management* (Spring), 7-13.

Bourdieu, P. (1977). *Outline of a Theory of Practice* Cambridge University Press.

Brown, J., and Duguid, P. (1991). "Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation," *Organization Science* 2(1), 40-57.

Brown, J.S. and Duguid, P. (1998). "Organizing Knowledge," *California Management Review* 40(3), 90-111.

Buckingham Shum, S. (1996). "Analyzing usability of a Design Rationale Notation,: In T. P. Moran and J. M. Carroll (Eds), *Design Rationale: Concepts, Techniques and Use* Lawrence Erlbaum Associates, Mahwah, NJ.

Carley, K. (1991). "A Theory of Group Stability," *American Sociological Review* 56, 331-354.

Caron, J.R., Jarvenpaa S.L., and Stoddard, D.B. (1991). "Business reengineering at CIGNA corporation: experiences and lessons learned from the first five years," *MIS Quarterly* (September), 233-250.

Chesbrough, H. and Teece, D.(1996). "When is Virtual Virtuous? Organizing for Innovation," *Harvard Business Review* (January-February), 65-73.

Clemons, E.K., Thatcher, M.E. and Row, M.C. (1995). "Identifying Sources of Reengineering Failures: A Study of the Behavioral Factors Contributing to Reengineering Risks," *Journal of Management Information Systems* 12(2), 9-36.

Cole, R.E. (1994). "Reengineering the corporation: a review essay," *Quality Management Journal1(4),* 77-85.

Cypress, H.L. (1994). "Reengineering - MS/OR imperative: make second generation of business process improvement mode work," *OR/MS Today* (February), 18-29.

Davenport, T.H.(1993). *Process Innovation: Reengineering Work through Information Technology,* Harvard Press, Boston, MA.

Davenport, T.H. (1995). "Business Process Reengineering: Where It's Been, Where It's Going," in *Business Process Change: Reengineering Concepts, Methods and Technologies,* V. Grover and W. Kettinger (eds.), Idea Group Publishing,Hershey, PA, 1-13.

Davenport, T.H., Jarvenpaa, S.L. and Beers, M.C. (1996). "Improving Knowledge Work Processes," *Sloan Management Review* (Summer).

Davenport, T.H., De Long, D.W. and Beers, M.C. (1998). "Successful Knowledge Management Projects," *Sloan Management Review* (Winter), 43-57.

Davenport, T.H. and Prusak, L. *(1998). Working Knowledge: How Organizations Manage what they Know*. Harvard Business School Press: Boston, MA.

Davenport T.H., and Stoddard, D.B.(1994). "Reengineering: business change of mythic proportions?" *MIS Quarterly* (June), 121-127.

Davidson, W.H. (1993), "Beyond re-engineering: the three phases of business transformation," *IBM Systems Journal*32(1), 65-79.

DeSanctis, G., and Poole, M.S. (1994). "Capturing the Complexity in Advanced Technology Use: Adaptive Structuration Theory," *Organization Science* 5, 121-147.

Despres, C. and Chauvel, D. (1999). "Mastering Information Management: Part Six – Knowledge Management," *Financial Times* (8 March), 4-6.

Drucker, P.F. (1978). *The Age of Discontinuity*. Harper and Row: New York, NY .

Drucker, P.F. (1995). *Managing in a Time of Great Change*. Truman Talley: New York, NY.

Earl, M.J. and Scott, I.A. (1999). "What is a Chief Knowledge Officer?" *Sloan Management Review* (Winter), 29-38.

Finholt, T., Sproull, L. and Kiesler, S. (1990). "Communication and Performance in Ad Hoc Task Groups," in: J. Galegher, R.

Kraut, and C. Egido (eds), *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work* Lawrence Erlbaum Associates, Hillsdale, N.J.

Flood, R.L. *(1993). Beyond TQM* Wiley, New York, NY.

Forbes (1997). Special Focus on Intellectual Capital, *Forbes ASAP* (7 April).

Frappaolo, C. (1998). "Defining Knowledge Management: Four Basic Functions," *Computerworld* 32(8).

Fulk, J. and DeSanctis, G. (1995). "Electronic Communications and Changing Organizational Forms," *Organization Science* 6, 337-349.

Gartner Group (1998). "Knowledge Management Scenario," conference presentation, SYM8KnowMan1098Kharris.

Glazer, R. (1998). "Measuring the Knower: Towards a Theory of Knowledge Equity," *California Management Review*40(3), 175-194.

Grover, V., Jeong, S.R., Kettinger, W.J. and Teng, J.T.C. (1995). "The Implementation of Business Process Reengineering," *Journal of Management Information Systems* 12(1), 109-144.

Grudin, J. (1996). "Evaluating opportunities for design capture,: in: T. P. Moran and J. M. Carroll (Eds), *Design Rationale: Concepts, Techniques and Use* Lawrence Erlbaum Associates, Mahwah, NJ.

Guha, S., Kettinger, W.J., and Teng, J.T.C.(1994). "Business process reengineering: building a comprehensive methodology," *Information Systems Management* (Spring), 13-22.

Hammer, M. (1990). "Reengineering work: don't automate, obliterate," *Harvard Business Review* (July/August), 104-112.

Hammer, M. and Champy J.(1993). *Reengineering the Corporation: A Manifesto for Business Revolution* Harper Business Press, New York, NY.

Hammer, M. and Stanton S. (1995). *The Reengineering Revolution: A Handbook* Harper Business Press, New York, NY.

Hansen, G. (1994). "A complex process: the case for automated assistance in business process Reengineering," *OR/MS Today* (August), 34-41.

Harrington, H.J.(1991). *Business Process Improvement: the Breakthrough Strategy for Total Quality, Productivity, and Competitiveness* McGraw-Hill, New York, NY.

Henderson, J.C., and Venkatraman. N.(1993). "Strategic alignment: leveraging information technology for transforming organizations," *IBM Systems Journal* 32(1), 4-16.

Hoffherr, G.D., Moran, J.D. and Nadler, G. (1994). *Breakthrough Thinking in Total Quality Management* Prentice-Hall, Englewood Cliffs, NJ.

Hutchins, E. (1991). "Organizing Work by Adaptation," *Organization Science* 2(1).

Johansson, H.J., McHugh, P., Pendlebury, A.J. and Wheeler, W.A. III, (1993). *Business Process Reengineering: Breakpoint Strategies for Market Dominance* Wiley, Chickster, UK.

Kamel, M. (1999). "Knowledge Acquisition," in: J.G. Webster (Ed.), *Wiley Encyclopedia of Electrical and Electronics Engineering* Vol. 11, New York: Wiley.

Kettinger, W.J., Guha, S. and Teng, J.T.C.(1995). "The Process Reengineering Life Cycle Methodology: A Case Study," in: V. Grover and W. Kettinger (eds.), *Business Process Change: Reengineering Concepts, Methods and Technologies*, Idea Group Publishing, Hershey PA, 211-244.

Klein, M.M. "Reengineering methodologies and tools: a prescription for enhancing Success," *Information Systems Management* (Spring 1994), 30-35.

Labor. U.S. Department of Labor Report (1991).

Lave, J. (1988). *Cognition in Practice: Mind, Mathematics, and Culture in Everyday Life* Cambridge University Press.

Lave, J., and Wenger, E. *(1990). Situated Learning: Legitimate Peripheral Participation* Cambridge University Press.

Leavitt, H.J. (1965)."Applying organizational change in industry: structural, technological and humanistic approaches," in: J. March (Ed.), *Handbook of Organizations* Chicago, IL: Rand McNally.

Levitt, B. and March, J. (1988). "Organizational Learning," *Annual Review of Sociology 14,* 319-340.

Levi-Strauss, C.(1966). *The Savage Mind* University of Chicago Press, Chicago.

Lynn, L. (1992). "Valuing Tradition while Changing: the Japanese Experience," in: Srivastva and Fry (eds.), *Executive and Organizational Continuity* Jossey-Bass, San Francisco.

McCartney, L. (1998). "Getting Smart about Knowledge Management," *Industry Week* (4 May).

Miles, G., Miles, R.E., Perrone, V. and Edvinsson, L. (1998). "Some Conceptual and Research Barriers to the Utilization of Knowledge," *California Management Review* 40(3), 281-288.

Nissen, M.E. (1997). "A Focused Review of the Reengineering Literature: Expert Frequently Asked Questions," *Quality Management Journal*, 3(3), 52-66.

Nissen, M.E. (1998), "Redesigning Reengineering through Measurement-Driven Inference," *MIS Quarterly* 22(4), 509-534

Nissen, M.E. (1999, forthcoming). "Knowledge-Based Knowledge Management in the Reengineering Domain," *Decision Support Systems* Special Issue on Knowledge Management.

Nonaka, I. (1994). "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science* 5(1), 14-37.

O'Leary, D.E. (1998). "Enterprise Knowledge Management," *Computer* (March).

Olsen, J. and March, J. (1975). "The Uncertainty of the Past: Organizational Learning under Ambiguity," *European Journal of Political Research* 3, , 147-171

Orr, J. (1990). *Talking About Machines: the Ethnography of a Modern Job* doctoral dissertation, Cornell University.

Prerau, D. (1990). *Developing and Managing Expert Systems: Proven Techniques for Business and Industry* New York: Addison-Wesley.

Roberts, B.(1996). "Internet as Knowledge Manager," *Web Week* (9 September), 30.

Ruggles, R. (1997). *Knowledge Management Tools* Butterworth-Heinemann: Boston, MA (1997).

Ruggles, R. (1998). "The State of the Notion: Knowledge Management in Practice," *California Management Review* 40(3).

Sengupta, K., and Ramesh, B. (1999, forthcoming). "Decision Support for Virtual Teams: Issues and Design Principles," *Accounting, Management, and Information Technology*.

Sengupta, K., and Zhao, J. L. (1998). "Improving the Communicational Effectiveness of Virtual Organizations through Workflow Automation", *International Journal of Electronic Commerce* 3, 49-69.

Shen, S. (1987). "Knowledge Management in Decision Support Systems," *Decision Support Systems* 3, 1-11.

Stoddard, D.B. and Jarvenpaa, S.L. (1995). "Business process redesign: tactics for managing radical change," *Journal of Management Information Systems* 12(1), 81-107.

Suchman, L.(1987). *Plans and Situated Actions: the Problem of Human-Machine Communication* Cambridge University Press.

Teece, D.J.(1996). "Firm Organization, Industrial Structure, and Technological Innovation," *Journal of Economic Behavior and Organization* 31(2), 193-224.

Teece, D.J. (1998). "Research Directions for Knowledge Management, *California Management Review 40(3),* 289-292.

Turban, E. and Aronson, J.(1998). *Decision Support Systems and Intelligent Systems* (Fifth Edition), Prentice-Hall: Upper Saddle River.

Weick, K.(1995). *Sensemaking in Organizations* Sage: Newbury Park, CA.

Zuboff, S.(1988). *In the Age of the Smart Machine* Basic Books.

Mark Nissen *is Assistant Professor of Information Systems and Acquisition Management at the Naval Postgraduate School. His research focuses on the investigation of knowledge systems for enabling and managing change in areas such as process innovation, electronic commerce and knowledge management. Recently he has been investigating knowledge systems to innovate processes in the acquisition domain and is currently involved with intelligent acquisition agents. Before his information systems doctoral work at the University of Southern California, he acquired over a dozen years of management experience in the aerospace and electronics industry and served as a Supply Officer in the Naval Reserve.*

Magdi Kamel *is an Associate Professor of Information Systems at the Naval Postgraduate School in Monterey. He received his Ph.D. in Information Systems from the Wharton School, University of Pennsylvania. His main research interest is in the design and implementation of database and knowledge base systems. Specifically, he is interested in data models and languages; data modeling; knowledge acquisition, representation, and implementation; interoperability and integration issues in heterogeneous database and knowledge base systems; and database and knowledge base connectivity to the Internet. He has consulted in these areas for several organizations and is the author of numerous published research papers on these topics. Dr. Kamel is a member of Association for Computing Machinery and IEEE Computer Society.*

Kishore Sengupta *is Associate Professor of Information Systems, Naval Postgraduate School, Monterey, California. In 1996-1997, he was a Visiting Scholar at the Hong Kong University of Science and Technology. Dr. Sengupta received his Ph.D. in Management Information and Decision Systems from Case Western Reserve University, Cleveland, OH. His research interests are in knowledge management for collaborative work, decision support for dynamic tasks, and electronic commerce. Dr. Sengupta has published in leading journals on management and information systems, including Management Science, MIS Quarterly, IEEE Transactions on Software Engineering, IEEE Transactions on Systems, Man, and Cybernetics, IEEE Transactions on Engineering Management, Accounting, Management and Information Technology, Omega, Decision Support Systems, and Concurrent Engineering Research and Applications.*