

DUAL INFORMATION SYSTEMS: SUPPORTING ORGANIZATIONAL WORKING AND LEARNING BY MAKING ORGANIZATIONAL MEMORY TRANSPARENT*

Timo K. Käkölä

Department of Computer Science and Information Systems

University of Jyväskylä, FIN-40351 Jyväskylä, Finland

Tel.: +358 50 523 4165, Fax: +358 14 603 068, E-Mail: timo.kakola@cc.jyu.fi

Kalle I. Koota

Orion Corporation, Orion-Pharma, R&D Information Systems

P.O. Box 425, FIN-20101 Turku, Finland

Tel.: +358 2 2727665, Fax: +358 2 2727510, E-Mail: kalle.koota@orion.fi

Abstract

The conceptual design of most computer-based information systems reflects a dualism of technology. During the development phase, part of the work-domain related knowledge is formalized and encoded in the software, making it difficult for users to reflect upon and use this knowledge. This design/use-dualism contributes to the deterioration of the interpretive flexibility of information systems. This paper outlines an information systems architecture called Dual Information Systems (DIS) that incorporates the concepts of an organizational memory information system (OMIS) in a broader framework. DIS help bridge the design/use dualism by providing organizations with a set of services that enable and reinforce both effective, institutionalized working *and* the questioning and (re)construction of computer-supported work routines. DIS have a four-layered conceptual structure: (1) people draw on the business layer to work and learn; (2) people use the breakdown layer to handle unexpected breakdowns; (3) self-organizing project teams use the project layer to create innovative work and IS (re)designs; and (4) the Knowledge Sharing Server acts as an OMIS by storing these redesigns and making them organizationally available to facilitate working and learning as well as subsequent redesign efforts. We outline the theoretical background, conceptual structure and generic services of DIS. We elaborate on the services and the conceptual design of the business and breakdown layers of DIS (bDIS). The services help people work effectively and develop competence needed to handle breakdowns and participate in the redesign project teams. The conceptual design extends the hyperknowledge framework of Chang et al. Finally, we demonstrate the conceptual design and services in a financial services organization with the help of the ReDIS prototype.

Keywords

Act Orientation, Dual Information Systems, Hyperknowledge Framework, Hyperknowledge Organization, Interpretive Flexibility, Knowledge-Creation Nets, Organizational Interfaces, Organizational Creation of Knowledge, Organizational Learning, Organizational Memory

* Timo Käkölä is responsible for the theoretical development of this paper. Both authors contributed equally to Section 4. Kalle Koota programmed the ReDIS prototype.

1 INTRODUCTION

Information systems researchers have paid considerable attention to developing systems to enhance managerial decision making and learning (e.g., group and executive support systems) and to automate office routines (e.g., workflow systems) [1]. While this research remains relevant, it distracts attention from the need for new knowledge about the development of systems that support *organizational* memory¹ and the creation and sharing of knowledge. Organizations' existing office systems remain important sources of new knowledge. It becomes crucial for researchers to determine how to redesign these systems to facilitate organizational creation and sharing of knowledge in the context of work and work redesign.

Redesign presumes changes in the conceptual design of office systems. To do redesign, one must first understand the role of computer-based information systems (CBIS) in organizations and why many systems are inadequate. We draw upon the work of Orlikowski [2], Nurminen [3], and Nonaka [4] in this paper to reach a conceptual understanding. Orlikowski [2] uses the term "the duality of technology" to imply that organizations can effectively employ information technology as a source of working and learning if agents² can use and modify the technology whenever it is necessary to redesign computer-supported work practices, and if the technology can be institutionalized as a legitimate component of the organizational working and learning environment.

Unfortunately, powerful CBIS are constrained because many organizations today suffer from *dualistic, institutionalized* CBIS that hide the constructed nature of CBIS from agents [5]. They (1) limit lateral communication, coordination and knowledge sharing; (2) provide little feedback to agents (especially in the lower echelons of an organization) on work arrangements and on the coordination and communication patterns that emerge from their use; (3) limit agents' ability to reflect and inquire within the social and technical contexts in which the agents are embedded, restraining them from creating, questioning, and modifying practical knowledge when problems emerge; and consequently (4) endanger the process of reinvention that any complex technological artifact should undergo when put to use [6]. For instance, changing the computer-supported work processes through user-driven work and IS design is almost impossible unless agents thoroughly understand the content and organization of work [7]. This design/use dualism of many IS is the problem to be addressed in this paper.

Organizations would benefit from an information systems architecture that bridges this dualism and taps the resources and capabilities of *all* agents to foster both effective, institutionalized working and the questioning and (re)design of computer-supported work. Orlikowski [2] provides neither a conceptual model nor practical guidelines to help organizations design such an

¹ Organizational memory can be defined as organizational knowledge with persistence [8]. Because the works this paper draws on to form the theoretical foundation of Dual Information Systems use the term "organizational knowledge," we will sometimes use that term instead of "organizational memory" to refer to knowledge with persistence.

² We use the word "agent" to refer to people whose work is computer supported. We want to avoid the term "user" since that term connotes that information systems can be used and studied independently of other types of work.

architecture. This paper continues the efforts of Käkölä [5] to do so. We will call the architecture Dual Information Systems (DIS).

Käkölä [5] complemented Orlikowski's work by drawing on a frame of reference called the act-oriented perspective [9]. This perspective helps transcend the design/use dualism by interpreting the functions and memory provided by computer software as an inseparable component of the work of knowledgeable people [3]. It also helps agents even on the shop floor level gain a comprehensive understanding of their work, its computer-supported parts, and its relationship to the business as a whole. In light of this perspective, a necessary but insufficient condition for a CBIS to qualify as a component of DIS is that it helps agents develop such an understanding.

However, well-developed *local* understanding of work is necessary but not sufficient for improving work. Moreover, the information systems architecture needed to align information systems with the act-oriented perspective would be expensive to build. For example, traditional integrated information systems hide their constructed nature to the extent that they often appear to agents as acting, knowledgeable subjects rather than media and outcomes of work [3]. The redesign of these systems to reflect the act-oriented perspective would unlikely be attractive to organizations if the benefits from the redesign were limited to understanding better "what is." To become fully applicable, DIS also need to help the agents on the frontline establish and join project teams, in which they collaborate with middle managers to (1) share knowledge of their computer-supported work processes and practices laterally within and between business units; (2) draw on their improved knowledge and skills to improve their own performance; and (3) create shared stocks of knowledge and crystallize this knowledge into work processes and practices that exceed the expectations of customers, realize the visions of top management, and create good jobs.

This paper outlines the conceptual design of DIS to enable and reinforce both effective, institutionalized working and the questioning and (re)construction of computer-supported work routines on individual and organizational levels. But no conceptual design can help transcend the design/use dualism without supporting organizational processes and structures. Therefore, this paper also aligns the conceptual design with the organizational design called "hyperknowledge organization" that, when enacted, supports the use of knowledge at work and the creation of new knowledge to reconstruct work.

The paper elaborates on the services of the business and breakdown layers of DIS that help people work effectively and develop competence needed to handle breakdowns and redesign work. We use the acronym bDIS to refer to the overall system formed by these two closely integrated layers. We extend the hyperknowledge framework [10] to develop the conceptual design of bDIS. To encourage commercial implementations of the conceptual design of DIS and especially its business and breakdown layers, the layers and their potential organizational benefits are demonstrated in the context of an imaginary financial services organization with the help of the artifact constructed, the ReDIS prototype.

Dual Information Systems thus denote the information systems architecture for hyperknowledge organizations. The architecture provides services defined in the conceptual design of DIS. These services help transcend the design/use dualism by facilitating shared knowledge creation and use (organizational memory) in the context of routine work and redesign work. The services (1) conceptually unite manual and computerized aspects of work, thus helping agents to understand their work holistically, including its computerized aspects; (2) let agents zoom in on

the details of their work practices and check shared databases for mistakes in order to deepen their knowledge and fix many breakdowns locally; (3) help agents draw on their improved knowledge of work designs to enter and interact in redesign project teams where they can share knowledge of computer-supported work practices so that best practices can be created and crystallized into work redesigns; and (4) store the created design knowledge in the organizational knowledge base so that it can later be used to enact, reflect on, and reconstruct work practices. In effect, DIS meet the definition of an Organizational Memory Information System (OMIS) as given by Stein and Zwass [11, p. 89]: an OMIS is “a system that functions to provide a means by which knowledge from the past is brought to bear on present activities, thus resulting in increased levels of effectiveness for the organization.”

This paper proceeds as follows. Section 2 outlines the theoretical background of DIS. Section 3 first outlines the conceptual design of DIS to provide a context for more detailed conceptual development of the business and breakdown layers of DIS. Next, the hyperknowledge framework is discussed and extended to develop the conceptual design of bDIS. Section 3 concludes by discussing the use of work and software process modeling in Dual Information Systems. Section 4 demonstrates the conceptual design of bDIS in a financial services organization and encourages commercial implementations with the help of ReDIS. The last section states conclusions and discusses issues for future research.

2 DUALITY OF TECHNOLOGY, ACT ORIENTATION, AND HYPERTEXT ORGANIZATION

To design Dual Information Systems, a comprehensive theoretical understanding of the role of CBIS in organizational working, the creation of knowledge, and learning is necessary. We recapitulate the work of Orlikowski [2], Nurminen [3], and Nonaka [4] in this section to provide the theoretical background of the conceptual design of DIS.

2.1 Duality of Technology, Interpretive Flexibility, and Time-Space Disjuncture

Orlikowski [2] introduces three concepts that are central to this paper: the duality of technology, its interpretive flexibility, and time-space disjuncture. The duality concept is derived from the duality of structure that forms a key concept in Giddens' [12],[13],[14] structuration theory. Giddens [13] distinguishes between system and structure in his analysis of social systems. He regards social systems as patterned social relations reproduced across time and space through the actions of *human agents*. Structure provides for the binding of the social relations into social systems. Social systems have structural properties that are drawn upon in social interaction. Roberts and Scapens [15] state:

Through being drawn on by people, structures shape and pattern (i.e., structure) interaction. However, the structures themselves are reproduced only through interaction. This is the ‘duality of structure’; it is in this way that structures can be seen to be both the medium and the outcome of interaction. (p. 446)

This recursive, continuous process is what Giddens calls *structuration*.

The duality of technology concept sees information technology as a structural property that is recursively enacted by human agency and institutionalized in structure. Designers produce a technology to provide resources and rules by creating and encoding work-domain related knowledge into it. Agents socially construct a technology by assigning it different meanings and using it flexibly in their work. But technologies usually become reified and institutionalized over time because agents cannot continuously reinterpret or physically modify them, if the agents are to accomplish their work efficiently.

The duality of technology recognizes that “technologies are products of their time and institutional context, and will reflect the knowledge, materials, interests, and conditions at a given locus in history” [2]. However, the dual nature of CBIS is masked by the time-space disjuncture arising from the various phases (e.g., design, implementation, and employment) of interaction between a technology and organizations. Within and between these phases, the actions constituting the technology are separated temporally and spatially from the actions constituted by the technology.

According to Orlikowski [2, p. 421], the time-space disjuncture “is collapsed . . . by understanding that technologies have different degrees of interpretive flexibility.” This emphasizes that “there is flexibility in how people design, interpret, and use technology, but that this flexibility is a function of the material components comprising the artifact, the institutional context in which a technology is developed and used, and the power, knowledge, and interests of human actors . . .” as well as time [2, p. 421]. Orlikowski [2, p. 421] summarizes the causality between the time-space disjuncture and the interpretive flexibility of technology as follows: “The greater the temporal and spatial distance between the construction of a technology and its application, the greater the likelihood that the technology will be interpreted and used with little flexibility.”

2.2 The Design/Use-Dualism of Information Systems: Implications and Alleviations

The interpretive flexibility of information systems is often poor [16-20]. This inflexibility results in part from agents with insufficient shared knowledge of: (1) the nature of social practices as a whole; (2) the articulation of these practices in time and space by the structural properties of organizations; (3) their own roles in the organization; and (4) the role of information systems as a structural property mediating work processes.

One important reason for agents’ lack of awareness is that the conceptual and material structures of computer software reflect the *design/use dualism* of technology; during the institutionalized use of an IS the constructed nature of the IS is masked by the software. Dualistic information systems separate symbolic information from the material and social systems the symbols represent, hide the processing rules and retention structures in the software and database schemas, and blur the role of people as the producers and consumers of information [3, 21].

The prevalent dualistic structure and poor interpretive flexibility of CBIS have costly implications. Agents are restricted to using functions expressed in the software [22, p. 390]. They also face considerable difficulties monitoring their actions since they cannot fully interpret and

validate the meaning of information produced by the dualistic, “black box” systems, and they cannot see and feel the outcomes of their computer-supported actions [20, pp. 79-96]. Because of their limited ability to control all aspects of work, including computerized tasks, the agents cannot necessarily be responsible for their work as a whole. Finally, the agents’ ability to intervene in and transform existing social practices is limited because the agents cannot easily criticize and challenge the interpretations, resources and norms embedded in the algorithms and databases of the CBIS [2, 23].

But agents can regain control of their jobs in computer-supported work environments. Orlikowski [2] states:

. . . knowledgeable and reflexive human agents are capable of altering the controlling influence of the technology. The extent to which individuals modify their use of technology, however, depends on whether they acknowledge its constructed nature. This is determined by the degree to which individuals can recognize the mediating role of technology, can conceive of an alternative beyond it, and are motivated to action. (p. 418)

Thus, agents’ ability to control their work depends on their domain and technology-related skills and knowledge as well as the shared stocks of knowledge that inform them about their own and other agents’ roles and normatively regulated and sanctioned behaviours [20, 23].

We summarize the relationships between the time-space disjuncture, the design/use-dualism, and the interpretive flexibility of CBIS as follows: The greater the temporal and spatial distance between the construction of a CBIS and its application and the greater the conceptual separation of the CBIS from agents and their work during design, the greater the likelihood that the CBIS will be dualistic (i.e., it will be institutionalized and its constructed nature will be hidden from agents) and that it will be interpreted and used with little flexibility.

The conceptual separation between dualistic and Dual Information Systems is necessary because it has clear implications for information systems design. For example, the redesign of a dualistic software package to incorporate tailorable functions and sophisticated support systems will likely increase the interpretive flexibility of information systems built on the software. But if the conceptual design of the software continues to separate computerized and non-computerized aspects of work, this redesign does not necessarily result in such services of Dual Information Systems that help agents theoretically and holistically understand their work.

The maximum benefits of DIS are unlikely to be realized unless the application of the conceptual design and services of DIS transcends the design/use dualism, and thus helps agents control and redesign their work as a whole, including its computer-supported aspects. For this purpose, DIS should bridge the time-space disjuncture vertically (between users and designers) by providing services that help users understand the constructed nature of software partly constituting and partly constituted by users’ work. In the following subsection, we outline a possible conceptual solution to build this bridge.

2.3 Act-oriented Perspective: Toward DIS on the Level of Human Agency

The time-space disjuncture between the design and use of CBIS and the dualistic nature of computer software imply that agents often cannot see the link between their work as a whole and

its computerized parts. Because Orlikowski's [2] work is descriptive, it does not provide a conceptual solution to uncover this missing link.

Käkölä [5] used the act-oriented perspective [3, 9] to provide a solution. This perspective states that CBIS cannot be separated from agents' work because no CBIS can serve as a conscious actor. Therefore, software modules should be designed so that they can be interpreted as computerized tasks that have one or more responsible human agents. Conceptually, each agent has his or her own information system receiving, memorizing, processing, and transmitting the information necessary for his or her work.

The act-oriented perspective bridges the time-space disjuncture *vertically* by seeing the knowledge encoded in software by designers in time-space context A as the acts of the responsible agents in time-space context B and *horizontally* by making explicit the coordinating role of CBIS, even when the computer-supported acts of agents are inter-connected through shared objects of work in integrated databases. Once this bridge has been built, agents can better exploit the interpretive flexibility of Dual Information Systems to work cooperatively and to share knowledge.

Yet, the perspective is not fully in line with the duality of technology. It operates on the level of human agency and emphasizes agents' need and ability to control their work and CBIS, but it does not address organizational memory as a way to create new computer-supported work practices. In this sense, act orientation favors the institutionalized character of information systems and organizations. Moreover, in the act-oriented perspective, all agents are seen as knowledge workers who need to be able to control the specialized knowledge embedded in their CBIS to the same extent as they control the manual aspects of their work. The assumption of the bounded nature of responsibility and control partly aligns the perspective with the traditional bureaucratic view. But this control over knowledge creates increased opportunities to learn skills and create new knowledge that transcend boundaries, and thus considerably expands the authority of workers in the lower echelons. Unless tied into strict bureaucratic rules, the perspective could yield an organization managed from the bottom-up in which entrepreneurial agents and small self-organizing groups would create and apply knowledge. Coordination of work and sharing of knowledge would be difficult because the key visions and norms sanctioning and justifying joint action would be missing or chaotic; no central authority would exist; and information would be dispersed and difficult to use. It is unclear how the act-oriented perspective alone could help organizations resolve or use this contradiction between increased opportunities to (re)construct work practices and the requirement of tightly bounded work and knowledge structures.

In summary, the paradox of act orientation is that it requires a bureaucratic structure to function well but simultaneously makes agents aware of the weaknesses of this structure by revealing how rigidly working and learning is governed by the organizational and technical rules and resources. We accept the perspective as a basis for Dual Information Systems but we are not satisfied with the functional bureaucracy or bottom-up driven groups as the best organizational forms for DIS. Dual Information Systems call for an organizational design that successfully combines the beneficial aspects of bureaucratic and group-based forms and eliminates their weaknesses. Nonaka [4] has developed an organizational design prototype called hypertext organization to help organizations realize such a combination. This design is described in the next section.

2.4 Hypertext Organization Design: Toward DIS on the Organizational Level

How can organizations create and rapidly shift between contexts of interaction where responsibility for the present and for the creation of future work practices are equivalently sanctioned and enacted? Nonaka [4] proposes ‘hypertext organization’ as an organizational design prototype that helps design such contexts. Hypertext organization is a *dual* organizational structure: it “coordinates the allocation of time, space, and resource within the organization” [4, p. 33] so that an organization can achieve high performance in routines *and* simultaneously ensure long-term survival by creating and applying new knowledge. DIS have the same dual purpose: facilitation of effective routine work *and* redesign work. Therefore, the maximum benefits of DIS would more likely be realized if the conceptual structure of DIS reflected the hypertext organization structure.

Hypertext organization is formed by the dynamic combination of hierarchically organized business units and self-organizing project teams that pursue the equivocal visions of the top management by drawing upon and accruing an organizational knowledge base. Nonaka [4] states:

The core feature of the hypertext organization is the ability to switch between the various ‘contexts’ of knowledge creation to accommodate changing requirements from situations both inside and outside the organization. . . . Each context has a distinctive way of organizing its knowledge creation activities. . . . Hypertext organization design first distinguishes the normal routine operation conducted by a hierarchical formal organization from the knowledge creating activities carried out by self-organizing teams. . . . By establishing the most appropriate organizational setting for the two activities, an organization can maximize the efficiency of its routine operation, which is determined by bureaucratic principles of division of labor and specialization, and also the effectiveness of its knowledge creation activities. (pp. 32-33)

Hypertext organizations are comprised of three layers; knowledge-base, business-system, and project-system (Figure 1). The ‘knowledge-base’ layer “embraces tacit knowledge, associated with organizational culture and procedures, as well as explicit knowledge in the form of documents, . . . computerized databases, etc.” [4, p. 33]. Normal work routines are enacted by a formal, bureaucratic organization in the ‘business-system’ layer (hereafter “business layer”). The ‘project-system’ layer (hereafter “project layer”) provides a field of interaction where loosely linked project teams create knowledge.

In a hypertext organization, knowledge is created through the circular movement of agents among the three layers. Members of project teams are selected from different functions and business units across the business layer. They interact with the knowledge-base layer at the bottom and make an ‘inventory’ of the knowledge acquired and created in the project layer. “After categorizing, documenting, and indexing the new knowledge, they come back to . . . business-system layer and engage in routine operation until they are called again for another project” [4, p. 33].

From the viewpoint of this paper, the hypertext organization as a theoretical organization design model is flawed: because the model is highly abstract, it pays little attention to technology. It views technology as a form of explicit knowledge stored in the knowledge-base layer. It views

practices in general (including those where technologies are designed and deployed) as forms of organizational tacit knowledge. Nonaka does not explicitly address interactions between technologies and work practices.

However, hypertext organization design is implicitly in agreement with the duality of technology because it views technologies and practices as shared stocks of knowledge and their development and use as a continuous creation and exploitation of knowledge. The act-oriented perspective can also be enacted effectively. The business layer relies on a bureaucratic division of work that allows clear assignment of responsibilities even for computerized pieces of knowledge in the knowledge-base layer. The project layer allows people to leverage on their improved knowledge of work as a whole, and thus reconstruct work when necessary. In the following section, we propose a conceptual design of DIS to make explicit the connection between DIS and hypertext organizations.

----- (Insert Figure 1: “Hypertext Organization – An Interactive Model of Hierarchy and Nonhierarchy [4, p. 34]” about here).-----

3 DUAL INFORMATION SYSTEMS AND COMPUTER-SUPPORTED WORK

3.1 The Conceptual Design of DIS

The primary purpose of the Dual Information Systems architecture is to enable and reinforce *both* effective, institutionalized working *and* the questioning and (re)construction of computer-supported work routines. Designing such an architecture is a complex undertaking. Thus, it is beyond the scope of this paper to examine in detail all the services that might be critical for the architecture to meet this purpose. In this subsection, we outline our proposal for the conceptual design of the DIS architecture to provide readers with an understanding of the context in which this paper will develop more detailed design guidelines for certain services of DIS.

In accordance with the hypertext organization design, the structure of the proposed conceptual design of DIS has three primary layers (Figure 2): (1) agents on the business layer continuously draw on the business layer of DIS to work, learn, and coordinate routines in the business units; (2) self-organizing project teams on the project layer use *the project layer* and *the knowledge sharing server of DIS* to produce innovative work and IS (re)designs that can be enacted on the business layer; (3) *the knowledge sharing server of DIS* is a repository of organizational memory in the form of explicit work and IS design knowledge in the knowledge-base layer. In addition, there is *the breakdown layer of DIS* that agents on the business layer can use to zoom in on the details of their work and to deepen their understanding of the computerized aspects of work in order to handle unexpected (coordination) breakdowns. We call hypertext organizations [4] that enable and are enabled by the DIS architecture hyperknowledge organizations.

From the point of view of effective work and work redesign, the importance of routine use of information systems cannot be over-stated. Information systems as a structural property of organizations remain constitutive of organizational realities that can be known, understood, and

acted upon [2, 19, 23, 24], For example, Tyre and Orlikowski presented and analyzed three case studies and found [19, p. 111] that “. . . routine use was . . . necessary for on-going adaptation; it provided the raw data that, if utilized, could lead to improvements in the technology or the way it was applied in the local context.” Furthermore, the creation of new knowledge to improve work in ambiguous circumstances is rarely possible without the objective information obtained by monitoring and measuring organizational routines, because this information can be used to refute or support and legitimate more subjective interpretations [25]. For these reasons, we focus on the business and breakdown layers of DIS (bDIS). Käkölä and Koota [26] discuss the project layer and the knowledge sharing server of DIS.

------(Insert Figure 2: “The conceptual design of a DIS architecture in a business unit of a hyperknowledge organization” about here.)-----

3.2 Business and Breakdown Layers of DIS

Käkölä [5] presented the Embedded Systems Approach (ESA) to instantiate the business and breakdown layers of DIS into Embedded Application Systems [9], which follow the act-oriented structure. They enable the effective enactment of routines in business units and allow agents to reflect and inquire into their work practices as a whole, including computerized tasks. In a continuous knowledge-creating spiral, ESA relies on and enables the project teams of the hyperknowledge organization. The teams develop new concepts of computer-supported work and share them among the business units. Work processes of each unit are externalized, visualized, and memorized by using organizational and software process modeling formalisms. Process models (subsection 3.5) serve as the building blocks of the organizational interface [27] through which the computerized tasks and knowledge become inseparable components of the agents’ working and learning environment. Breakdown management services let agents zoom in on the details of their work practices, check shared databases for mistakes, and fix many breakdowns locally. Additional learning services can easily be provided. We will elaborate in subsection 3.4 and section 4.

3.3 Hyperknowledge as a Basis of the Business and Breakdown Layers of Dual Information Systems

Dualistic technologies embed work-related knowledge in the form of linearly organized, fairly immutable concepts and procedures. These concepts and procedures are hidden in the software and tend to promote the institutionalization of work practices to those envisioned by designers in earlier time-space contexts. Consequently, the business and breakdown layers of DIS must transcend these linear structures, make organizational memory transparent, and let agents organize computerized information, concepts, and work procedures as flexibly as they organize their manual work practices and materials. In this way, agents can work effectively and construct multiple interpretations of work in a hyperknowledge organization.

Hyperknowledge [10] is a promising way to design the business and breakdown layers of DIS. It is strongly related to the concepts of hypertext and hypermedia [28, 29], and to the architecture

of hypertext systems [30]. However, the idea of hyperknowledge is wider than those of hypertext or hypermedia. Hyperknowledge is an ideal working and learning environment that holds work-related knowledge and, at the same time, defines the nature of hypertext or hypermedia. Such an environment partly eliminates the reifying effects of the time-space disjuncture: the designers still encode a part of work-related knowledge into the environment, but now this knowledge is transparent and partly nonlinear. Agents can navigate flexibly in the environment, widen their knowledge, actively externalize their tacit knowledge, and combine new knowledge into the environment.

Chang, Holsapple, and Whinston [10, 32, 33] originated the framework of the hyperknowledge environment. It recognizes that an agent:

cognitively possesses many diverse and interrelated pieces of knowledge (i.e., concepts). Some are descriptive, others are procedural in nature, yet others are concerned with reasoning, and so forth. The mind is able to deal with these concepts in a fluid and inclusive manner via controlled focusing of attention. In effect, the decision maker actively acquires (i.e., recalls, focuses on) desired pieces of knowledge by cognitively navigating among the universe of available concepts [33, p. 30].

The business and breakdown layers of DIS should be designed in the same manner so that they become natural extensions of people's working and learning environments.

According to the hyperknowledge framework, the configuration of a hyperknowledge environment consists of three components: the Language and the Presentation System, the Problem Processing System, and the Knowledge System. The Language and the Presentation System mediate messages to and from the hyperknowledge environment. The Problem Processing System contains the dynamic part of the system and may possess inferential abilities. The system consists of three different parts – the data manager, model manager and dialogue manager – with which it can handle all the user requests or responses to and from the various knowledge sources in the system. The Problem Processing System is capable of presenting the knowledge in different ways, as chosen through the user interface. The Knowledge System contains all the encoded, explicit knowledge, which can be descriptive, procedural, linguistic, reasoning, etc. It stores, in groups, concepts that are related to each other by definition and/or by association.

In an ideal hyperknowledge environment, the agent first receives an impression, possibly several impressions, of the same underlying concept. Each impression presents an image or images (visualization) on the user interface screen. The agent sees the concept through several windows and can thus internalize a holistic view of the concept. To avoid the problems presented by Conklin [34], the “lost in space” and “cognitive overhead” feelings, the hyperknowledge environment needs concept maps to guide both designers and agents [10].

The hyperknowledge framework is somewhat limited to the working, learning, and decision-making of individuals, yielding systems that primarily augment the intellect of individuals. Organizations may not become significantly more efficient or effective simply by giving everybody a definitive and associative access to organizational knowledge. Uncontrolled, unguided exploration and extension of available knowledge are unlikely to converge into meaningful concepts [4].

3.4 The Conceptual Design of the Business and Breakdown Layers of DIS

To remedy the individualistic orientation of the hyperknowledge framework, we develop the conceptual design of the business and breakdown layers of DIS in this section. The Embedded Systems Approach uses the *organizational role* concept [35, 36] to divide each business unit in the business layer of a hyperknowledge organization into a set of functional work groups. Each group is responsible for a clearly defined part of the work process(es) of a business unit. Each agent in a functional group enacts the same work role.

The efficiency and effectiveness of the approach depend on two factors. First, agents must be able and motivated to internalize specialized role-centric language and norms and to exploit the role-centric information resources to a maximum degree. Second, agents must be offered redundant information to help them construct shared stocks of knowledge about the interconnections between their computer-supported work routines and the organizational work and performance. Redundant information helps agents recognize their location in an organization, interact, develop a common direction, create new concepts, enter each others' spheres of operation, provide advice, fix coordination breakdowns, and even do each other's jobs if necessary [4].

The two factors described above promote both efficiency and effectiveness. When clear role-specific knowledge and resource requirements have been established, organizations can offer learning services and agents can use them to internalize quickly the skills necessary for role enactment. A shared language helps agents share tacit knowledge and use contradictions in their perspectives as a source of new knowledge to improve work practices [4, 20].

Clear boundaries between shared and role-centric stocks of knowledge are often difficult to determine. Moreover, the organizational creation and sharing of knowledge changes these boundaries over time: roles are merged or the job content and knowledge requirements are upgraded in other ways. The distinction is made for analytical and design purposes only.

The organizational interface of bDIS must offer agents a Language System with a balanced mix of shared and role-centric technology-enabled and constrained languages (Figure 3). To comply with the act-oriented perspective, the role-centric language defines institutionalized, prenegotiated responsibility for and control of the means and objects of work. For this purpose, bDIS must provide at least the following role-centric services: (1) *Work enactment and coordination* services help agents allocate resources and take care of their routines. (2) *Learning* services help agents develop a theoretical understanding of work, including its computerized aspects, needed to enact roles effectively. (3) *Performance monitoring* services help agents monitor both their work activity and its results. They also help agents benchmark their performance with agents within functional group(s) [37]. They promote internal competition and motivate agents to use the other services of bDIS effectively and to figure out ways to exceed performance expectations; all these are of great importance to accelerate the creation of knowledge [4].

No matter how well people do their work, routines may break down unexpectedly. Agents face two challenges in often chaotic breakdown situations. First, they must get their routines back on track as quickly as possible. Second, they must develop and test multiple hypotheses about what went wrong to ensure a lasting solution [20]. In this respect, breakdowns play a fundamental role in stimulating the creation of knowledge [4, 38, 39].

Breakdown management services help agents meet both challenges. When routines break down, they let agents shift their focus of attention from normal objects of work to the routinized patterns of interaction and the means of work, including the rules and resources afforded by software [5]. Hyperknowledge is especially useful here because the definitive and associative relationships between different chunks of knowledge in the Knowledge System help agents shift focus rapidly. For example, agents can perform audit trails to trace what happened earlier to artifacts (e.g., documents) for which they are responsible and use the learning services to understand what should have happened. If they hypothesize that the problem stems from mistakes of agents in other roles, they can use the redundant information services to better understand how and why such mistakes are possible.

------(Insert Figure 3: “The conceptual design of the business and breakdown layers of a DIS” about here.)-----

If the analysis reveals that the breakdown stems from coordination or other organizational problems spanning multiple functional work groups, agents can discuss the problems with middle managers. A project team can then be assembled to reconstruct computer-supported work practices and learning services to help eliminate in advance similar breakdowns. The project layer of DIS supports work redesign [26].

A shared language is developed by sharing redundant information according to the norms and rules negotiated and agreed upon in the project layer of a hyperknowledge organization. For this purpose, the business layer of DIS offers agents in functional work groups at least the following redundant information services: browsing, simulated enactment, and benchmarking of interconnected roles (i.e., the roles interacting with an agent directly or indirectly through Embedded Application Systems). In this way, agents can experiment with the jobs of their colleagues in interconnected functional work groups and understand bottlenecks in work performance. The norms and rules preserve the autonomy and privacy of agents by defining what information is shared and what is private. They can be reinforced by encoding some of them in the services of bDIS (e.g., through role-centric access rights to the Knowledge System of bDIS).

We divide the Knowledge System of bDIS into role-centric and shared components (Figure 3). The role-centric component is further divided into two subcomponents: tasks that people do and computerized tasks that Embedded Application Systems do for the people. Role-Connected Task Knowledge externalizes how people should do their routines (enactment knowledge), what are the objects of work (retention knowledge such as paper documents), what are the performance and skill requirements of successful enactment, and where work is physically done (ecological knowledge). These concepts are intertwined and enable the role-centric services of bDIS. For example, performance requirements must be tied explicitly with enactment knowledge so that people can use benchmarking to find and eliminate weaknesses in work performance. Role-Connected Computerized Task Knowledge presents a role-centric view of Embedded Application Systems. In accordance with the act-oriented perspective, the view defines the computerized aspects of work and relates them explicitly to the work of individuals in functional groups.

The shared Knowledge System is also divided into two parts: (1) Inter-Connected Role Knowledge views the work process of a business unit as a set of temporally and spatially

distributed interacting roles. It also includes generic knowledge of the objects of the process (e.g., products and services), external stakeholders (e.g., suppliers and customers), and the goals that connect these concepts to a whole. The staff knowledge component stores knowledge of performance, skill profiles, and availability of agents. It supports resource allocation and coordination of work. (2) Inter-Connected Computerized Role Knowledge presents an inter-connected role view of Embedded Application Systems. The view defines how the work of agents in different functional groups is coordinated and mediated by the systems.

The Problem Processing System dovetails with the Language System on the organizational interface side, and with the Knowledge System on the functionality side. The term “Problem Processing System” in the conceptual design of bDIS emphasizes that ‘routine work’ involves significant problem-solving and articulation work and must be supported by information systems to the same extent as ‘decision-making work’ [18, 40, 41].

3.5 Process Modeling and Dual Information Systems

How can organizational interfaces be developed to enable effective working, the creation of knowledge, and learning through smooth navigation in DIS? Most traditional structured systems analysis and design approaches are ill-suited to the development of organizational interfaces because (1) they focus almost exclusively on modeling data flows, data structures and other components of *technical* systems and (2) the models are difficult for agents to understand and use [42, 43]. These modeling languages reinforce the development of dualistic, reified information systems [5, 44, p. 12].

Fortunately, process modeling languages [45] remedy many of the weaknesses of the technically oriented modeling languages. Curtis, Kellner, and Over state [43, p. 75]: “Process modeling is distinguished from other types of modeling in computer science because many of the phenomena being modeled must be enacted by a human rather than a machine.” Käkölä [5] found the Role Interaction Net process modeling language [47] is reasonably effective in alerting people to the dual nature of information systems. Drawing on this research, we have developed the Knowledge Creation Net (KCN) language [48, 49].

The KCN language is based on organizational role [35, 36] and knowledge creation theories [4], Petri nets [50], and Role Interaction Nets [47]. In accordance with role theory, the language provides two primitives: roles and interactions. A KCN is composed of a set of concurrent roles. The behaviour of a role is described by its solitary actions and its interactions with other roles. Unlike the RIN language, the KCN language also describes how knowledge is created, shared, and converted in interactions between agents. This helps clarify the dual nature of information systems. People create and share tacit and explicit knowledge through the processes of socialization, externalization, combination, and internalization [4]. Computerized systems enable and constrain these processes by combining existing knowledge according to the rules and procedures encoded into these systems by people in earlier time-space contexts. This distinction between what people do and what computers ‘do’ can be made explicit when the Knowledge Creation Nets are employed in DIS. Figure 4 clarifies a few notations of the KCN language.

Techniques from Petri nets are used to give the language process description and enactment capability. Enactment capability means that the KCN models can be directly executed on the business layer of DIS if the technical implementation of the layer supports direct execution [47]. This is vital for DIS. First, contrary to static models, executable models provide organizations with clear incentives to maintain them. Second, executable models can be used both to measure the performance of work processes and to capture deviations between the specifications and the way work is really done. Consequently, the variances in performance and the underlying reasons for these variances can be quickly identified and fixed, and specifications can be updated to reflect improved work practices.

Models are always abstractions. Even process modeling can thus detach agents from their organizational and social setting and treat work itself as a sterile assemblage of tasks, roles, and interactions. For that reason, relying on process modeling alone in Dual Information Systems would be oversimplification. Additional learning services such as multimedia training sessions should be provided to help agents understand their work. However, their further investigation is beyond the scope of this paper.

------(Insert Figure 4: “Notations of the KCN language [48, 49]” about here.)-----

4 WORKING AND LEARNING WITH THE ReDIS PROTOTYPE

In this section, we present the ReDIS prototype to illustrate the business and breakdown layers of Dual Information Systems. Käkölä and Koota [26] describe the project layer and knowledge sharing server of ReDIS. The case organization and the work processes described in subsection 4.1 are fictitious, but the ideas stem partly from the IBM Credit Corporation case [51]. The case study gives ample opportunity to research the development and use of hyperknowledge-based DIS to support organizational coordination and the creation and sharing of knowledge. The ReDIS prototype helps business units process credit requests and issue credit. The ReDIS project was conducted in a laboratory environment. Personal computers and the Microsoft Access™ system [52] were selected as the development platform.

4.1 The Case Organization and its Credit Issuance Process

We make the following assumptions about the case organization and the role of ReDIS in the organization. (1) The Credit Corporation is a large organization and a subsidiary of another. (2) Issuing credit effectively is important for the organization as a whole because financing customers' purchases can be extremely profitable. (3) The Credit Corporation is organized as a hyperknowledge organization. The business units that issue credit within their regions form its business layer. (4) ReDIS systems support each business unit locally.

We focus on one business unit. Its credit issuance procedure is as follows (see Figure 5). After negotiating a sale, a sales agent in the Sales unit writes a standardized credit request and sends it electronically to the Credit unit for the region. The creditworthiness of the customer is checked.

Rejected requests are sent back to the sales agent. Approved, standard loan contracts are modified to meet customer requirements. An interest rate is determined. A quote is developed and sent back to the Sales unit.

Workers in different functional roles are responsible for their own work. If they cannot accomplish their tasks, they can seek help from “specialists,” who are typically employees with more experience and expertise. Relying on a specialist slows down the procedure but is sometimes unavoidable. If workers face a breakdown (e.g., an error in a credit request), they can enter into others’ spheres of operation and perform audit trails to see what has gone wrong. Spontaneous quality assurance in every step of the process is an integral part of the organizational culture of the Credit Corporation.

Different characteristics of a work process such as time, quality, and expense are measurable by analyzing various key figures. Each process step is measured by analyzing:

- Number of people involved in the step.
- Average processing time of credit requests.
- Average queuing time for the step.
- Percentage of requests that require a specialist’s help.

Agents use these measurements to benchmark their performance with agents in the same and inter-connected roles. Work redesign project teams use them and many other measurements to continuously improve the credit process and to periodically redesign it [26].

4.2 The Main Components of the Business and Breakdown Layers of ReDIS

Organizational interface. The organizational process models constructed using the Knowledge Creation Net (KCN) language serve as the organizational interfaces of the business units in ReDIS. The organizational interface of each unit has four basic functions. First, it provides agents with role-centric views of the Embedded Credit Application System (ECAS). These views let agents see both the manual and computerized parts of their work as a whole, and thus carry out their responsibilities. Second, agents can use the organizational interface to enact the role-centric functionality of ECAS. Third, it provides a full-fledged hyperknowledge environment with the services envisioned in the conceptual design of the business and breakdown layers (section 3.4); agents can navigate, run different types of queries, and internalize redundant, real-time information about the business unit, its work processes and performance, and ECAS. Fourth, it provides access to the project layer of ReDIS [26].

The performance monitoring services of ReDIS collect performance data from each process enactment and store them in the Knowledge System.

The Knowledge System. The procedural and descriptive chunks of knowledge visualized by the KCN model are stored in the Knowledge System of ReDIS. The Knowledge System is a component of organizational memory that holds all the work process-related explicit knowledge in a business unit. The system is implemented as a relational data base that holds relevant information about the business unit, the cast (agents associated with certain roles), the process, and process enactment (the number and characteristics of agents in different roles, and the interactions

that occur during the process). Naturally, all the relational tables accessed by ECAS are part of this larger knowledge system.

The Problem Processing System. An interactive working and learning environment is generated by the Problem Processing System that is driven by the Knowledge System's procedural knowledge in the form of AccessTM scripts and KCN models.

4.3 Using the Business and Breakdown Layers of ReDIS

We use working scenarios to illustrate how ReDIS fosters the enactment, coordination, and learning of computer-supported work processes. We assume that our imaginary persons, a credit checker and a clerk, have been trained for their jobs, but that they have not yet internalized the process of issuing credits. They do not work in other roles.

The credit checker, Sam, begins to work by entering ReDIS with his user id and password. The system now knows that he is working at the workstation, registers him as "present" in the Persons table of the staff knowledge system, and determines his role-connected rights and responsibilities. The KCN model of the credit issuance process appears on the screen. It serves as a concept map of the organizational interface to help agents construct a shared understanding of the process (Figure 5). The credit checker can see the different roles and tasks involved, and realize that he is an integral part of the unit with an important organizational function, not just another part of a mechanistic system.

The process fragments within the rounded rectangle visualize the inter-connected role view of the Embedded Credit Application System. In the earlier research of the first author, each computerized task performed by an Embedded Inventory Application System was visually represented in the role column of the agents responsible for the task. This design reflected the act-oriented perspective: the inseparability of agents' manual and computerized tasks. However, when this design was tested in a laboratory environment, users sometimes mixed up manual and computerized tasks [46]. Therefore, we have visualized the computerized tasks under the role ECAS. The database column facilitates the bridging of time-space disjuncture horizontally by visualizing the objects of work (here credit request documents) that mediate interactions between agents in different roles. Consequently, the structuring of work in time and space by ECAS is made explicit: agents can see when, where, how, and by whom these objects are stored and retrieved.

The number of requests in different stages of the work process is shown by the organizational interface in real time. This up-to-date reflection of work brings the KCN model to life and gives agents a sense of control over their work. Twenty-two requests are now waiting for the credit check.

------(Insert Figure 5: "The organizational interface of the business unit" about here)-----

The organizational interface is designed to provide agents with more detailed knowledge of the business unit and its processes to help coordinate and control work effectively and to handle breakdowns. Sam can use the KCN-based interface to zoom in on the details of the work

processes. In Figure 6, he has opened the Process Description form to view the interactions and ECAS-modules of the Check Credit task. The KCN model shows that credit checkers use ECAS to retrieve queuing credit requests from the data base, and then approve or reject the requests, possibly with the help of specialists. ECAS automatically provides customer information, stores approved requests in the data base and sends rejected requests back to sales agents.

Learning services related to each interaction are available by clicking the “Instructions” button. The credit checker can also click the KCN-icons of the ECAS modules to see the role-centric computerized tasks and the resources and rules designers or colleagues have encoded in these tasks. He can tailor certain rules and parameters according to the norms the work redesign project teams have negotiated using the project layer of DIS. Naturally, the norms are such that tailoring cannot jeopardize system integrity. For instance, as a default ECAS brings the requests from the request queue according to the FIFO-rule, but Sam prefers to handle them in descending order with respect to the requested amount. Therefore, he has changed the ordering parameter of ECAS from “request send time” to “requested amount”. These services help bridge the time-space disjuncture vertically between designers and workers.

-----*(Insert Figure 6: “The organizational interface of the credit checkers. The role-centric benchmarking service has been activated” about here)*-----

The credit checker can run predefined queries, write his own queries, and associatively navigate in the Knowledge System to retrieve information about different roles, interactions, and people. It is also possible to do role-centric benchmarking. Sam clicks the “Compare with Others” button of the Process Description form (Figure 6) and enters the three-month time frame as a parameter to compare his performance over the last three months to the average of the credit checkers in his business unit.

-----*(Insert Figure 7: “Credit check using the Embedded Credit Application System” about here)*-----

To start checking new credit requests, the agent clicks with a mouse the “Check Credit” process element of the organizational interface (Figure 5). The respective module of ECAS is executed. It opens the Credit form of ECAS and retrieves from the Credit table the next request in the queue (Figure 7). It also opens the Customer form to provide detailed customer information. A new instance is added to the Interaction table that relates the worker, the process step, and the credit request, and also tells when the interaction started and ended. Finally, the credit checker is registered as “engaged” in the Persons table.

In the following scenario, we assume that Sam faces a tricky request and needs to consult a “specialist.” He clicks the concept “specialist” in the organizational interface to open the Cast form (Figure 8). Information about specialists is automatically filtered from the staff knowledge system. He can then check out which workers are specialists, which are present, and, if possible, not busy, and where the suitable workers are located. He can contact the most suitable person, possibly using e-mail (the Contact button).

The credit checker records the outcome of the check using the Credit form (Figure 7). He also fills the check box to indicate that a specialist took part in the interaction. ECAS updates the Credit table and, in the case of approval, forwards the request to wait for modification. Finally, the performance monitoring service of ReDIS stores the use of specialist, the finishing time, and the outcome of the process in the Interaction table.

----- (Insert Figure 8: “The cast form of the organizational interface” about here) -----

The breakdown management services form the breakdown layer of ReDIS. They help agents during breakdowns quickly shift focus from routines to various details, including ECAS and the databases in which part of the work-related knowledge of agents has been encoded. The audit trail and other breakdown management services offer support during breakdowns that may emerge during the credit issuance process. After all, all situations cannot be solved simply by consulting a specialist. The audit trail service helps agents track down the life cycles of credit requests. The hyperknowledge characteristics of ReDIS play an important role when more knowledge is needed about previously unimportant aspects of work.

We illustrate the use of breakdown management services in the following scenario: A clerk, Amanda, is writing a quote letter for a significant sum of money. She knows that the customer has a poor credit history and begins to wonder whether the credit request should have been accepted in the first place. Maybe an error has taken place earlier in the process. She decides to investigate before completing the letter. But how to approach the problem? First, Amanda needs to understand how credit requests are checked in the credit issuance process. Second, and more importantly, she needs to know who is responsible for checking this particular credit request.

The KCN models of ReDIS help agents in different roles create a shared explicit language of their work routines, making the sharing of tacit knowledge easier. Fixing breakdowns is facilitated. Amanda has learned (i.e., in KCN-terms, internalized) from the organizational interface (Figure 5) that credit checkers are responsible for checking incoming credit requests. She activates the audit trail service (Figure 7) to trace the life cycle of the credit, and zooms in on the details of the credit-check process (Figure 9). She stepwise executes the KCN model of the credit-check process (Figure 6) using the simulated enactment services of ReDIS. The original credit request and customer information are available during the simulation. The customer information confirms her doubts about the creditworthiness of the company. She sees from the KCN model how ECAS automatically forwards requests according to the decisions of credit checkers. She realizes that the request would have automatically gone to a modifier rather than being returned to the sales agent, if the credit checker had *accidentally checked* the request approved. Amanda contacts the responsible agent to discuss it.

In summary, the services of the business and breakdown layers of ReDIS make organizational memory transparent and transcend the design/use-dualism of traditional CBIS by helping people (1) develop a theoretical understanding of their office work, including its computerized aspects; (2) work effectively; (3) maximally exploit knowledge gained from the routine use of ReDIS to monitor and continuously improve their performance; (4) develop and test multiple hypotheses of possible causes of breakdowns to get office routines back on track fast; and (5) participate in and contribute to work redesign projects on the project layer of the Credit Corporation.

------(Insert Figure 9: “The audit trail form” about here)-----

5 CONCLUSIONS AND FUTURE RESEARCH

Insufficient knowledge of work practices and the role of CBIS in enabling and conditioning these practices, especially in the lower echelons of organizations, reduces the interpretive flexibility of CBIS. Agents' insufficient knowledge results, in part, from the design/use-dualism of technology: during the institutionalized use of CBIS their constructed nature is partly hidden from people. CBIS with high interpretive flexibility transcend this dualism by helping people master their computer-supported work as a whole, participate in project teams that redesign work practices, and maintain constructive, internal competition that motivates the creation of knowledge in the context of work and work redesign. This paper describes Dual Information Systems, a conceptualization of the information systems architecture for hyperknowledge organizations that reduces the time-space disjuncture between the development and use of information systems *vertically* by interpreting all the computerized tasks and chunks of knowledge as the work and knowledge of agents, and *horizontally* by offering agents redundant information about their inter-connected work processes. It also describes a part of the ReDIS prototype to crystallize the conceptual design and encourage commercial implementations.

This paper has identified several design principles for the next generation of CBIS in office environments. First, to bridge the design/use-dualism of CBIS and thus reach and maintain high interpretive flexibility of information systems, it is fruitful to focus design on three conceptual layers of DIS, the project, business, and breakdown layers, as well as the knowledge sharing server. Second, the design of DIS must ensure that agents in the business layer of a hyperknowledge organization can easily enact routines according to partly prescribed work flow procedures, develop a theoretical understanding of their computer-supported work, and monitor their work performance using quantitative and qualitative benchmarks. Third, the design must offer breakdown management services to help agents in the business layer deepen their theoretical knowledge during breakdowns, find out what is wrong, and recover. Fourth, the design must help agents quickly return to their routines or, if recovery is not possible due to the severity of breakdowns, enter the project layer to redesign work. Finally, the design must accelerate the knowledge-creating spiral by helping organizations store and share work redesigns through knowledge sharing servers. The last three principles, especially, facilitate the redesign of CBIS as a coherent DIS architecture that constitutes an Organizational Memory Information System.

Process modeling languages and hyperknowledge environments are the cornerstones of the business and breakdown layers of DIS. Work routines, including their computerized parts, can be designed using process models, encoded in hyperknowledge environments, and visualized for enactment and reflection through organizational interfaces. Agents can flexibly navigate in the hyperknowledge environment, deepen their knowledge of the specific details of work, and understand the interconnections between their work and the business process as a whole.

We have drawn on Nonaka's hypertext organization model because a dual organizational design and a dual information system design mutually reinforce each other, increasing the interpretive flexibility of information systems. In traditional bureaucracies, (1) the creation of knowledge is fairly limited to the upper echelons of organizations [4]; (2) rigid, narrowly defined jobs make it difficult for nonmanagement workers to exploit the knowledge-creating potential of hyperknowledge-based DIS [20]; and (3) the performance monitoring services of DIS are likely to subject workers to unilateral managerial control. Managerial performance monitoring in hyperknowledge organizations should focus on results rather than on precise measurement of work steps to ensure the autonomy of functional work groups in the business [53]. Moreover, the extent of managerial control should be negotiated *constructively* by managers and functional work groups in the project layer, and subjected to renegotiation if managers or work groups cannot carry out their responsibilities or meet performance objectives. Käkölä [54] presents a more detailed analysis of the implications of interactions between managerial control patterns and coordination technologies for organizational working and learning.

Future research must establish more elaborate guidelines for the design of hyperknowledge organizations in which *all* agents can exploit the interpretive flexibility of DIS to work efficiently in the business layer, to redesign work effectively in self-organizing project teams, and to accumulate design knowledge in the knowledge-base layer. The conceptual design of DIS must also be elaborated with the detailed coverage of the services that, at least, should be offered to the project and knowledge-base layers of hyperknowledge organizations.

The next step is to assess and validate the utility of the services of business and breakdown layers of DIS by testing ReDIS first in the laboratory and later in the field. Although this research is still in progress, earlier research of the first author indicates that a system like ReDIS can be useful for working and learning. First, process model-based organizational interfaces help agents work and understand their work as a whole [46]. Second, useful hyperknowledge-based working and learning environments can be built [55, 56]. Consequently, we are confident that Dual Information Systems in general, and their business and breakdown layers in particular, offer substantial benefits to organizations that implement them to increase the interpretive flexibility of information systems in office work.

6 ACKNOWLEDGEMENT

We are grateful to Inger Eriksson and Markku Nurminen, whose innovative visions about Embedded Application Systems inspired our research, and to Omar El Sawy, Kalle Lyytinen, Lorne Olfman and the anonymous reviewers for their constructive comments on earlier versions of this paper.

7 REFERENCES

- [1] Swanson, E. B. and Ramiller, N.C. (1993). Information Systems Research Thematics: Submissions to a New Journal, 1987-1992. *Information Systems Research*, 4(4), pp. 299-330.

- [2] Orlikowski, W. (1992). The Duality of Technology: Rethinking the Concept of Technology in Organizations. *Organization Science*, 3(3), pp. 398-427.
- [3] Nurminen, M. I. (1988). *People or Computers: Three Ways of Looking at Information Systems*. Studentlitteratur, Lund, Sweden & Chartwell-Bratt.
- [4] Nonaka, I. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organization Science*, 5(1), pp. 14-37.
- [5] Käkölä, T. (1995a). Increasing the Interpretive Flexibility of Information Systems through Embedded Application Systems. *Accounting, Management, & Information Technologies*, 5(1), pp. 79-102.
- [6] Ciborra, C.U. and Lanzara G.F. (1994). Formative Contexts and Information Technology. *Accounting, Management & Information Technologies*, 4(2), pp. 61-86.
- [7] Hellman, R. (1989). User Support: Revealing Structure Instead of Surface. *Behaviour & Information Technology*, 8(6), pp. 417-435.
- [8] Ackerman, M.S., (1994). Augmenting the Organizational Memory: A Field Study of Answer Garden. *Proceedings of CSCW '94*, pp. 243-252.
- [9] Eriksson, I. and Nurminen, M. I. (1991). Doing by Learning: Embedded Application Systems. *Journal of Organizational Computing*, 1(4), pp. 323-339.
- [10] Chang, A-M., Holsapple, C. W., and Whinston, A. B. (1994). The Hyperknowledge Framework for Decision Support Systems. *Information Processing and Management*, 30(4), pp. 473-498.
- [11] Stein, E.W. and Zwass, V. (1995). Actualizing Organizational Memory with Information Systems. *Information Systems Research*, 6(2), pp. 85-117.
- [12] Giddens, A. (1979). *Central Problems in Social Theory: Action, Structure and Contradiction in Social Analysis*. Berkeley, CA: University of California Press.
- [13] Giddens, A. (1984). *The Constitution of Society*. Cambridge, UK: The Polity Press.
- [14] Giddens, A. (1991). Structuration theory: Past, Present and Future. In C. G. A. Bryant and D. Jary, (Eds.), *Giddens' Theory of Structuration, A Critical Approach*. London: Routledge and Kegan, pp. 201-221.
- [15] Roberts, J., and Scapens, R. (1985). Accounting Systems and Systems of Accountability: Understanding Accounting Practices in their Organizational Context. *Accounting, Organizations, and Society*, 10, 4, pp. 443-456.
- [16] Bødker, S. and Grønbæk, K. (1991). Design in Action: From Prototyping by Demonstration to Cooperative Prototyping, in *Design at Work* (eds. J. Greenbaum and M. Kyng), Lawrence Erlbaum Associates, pp. 197-218.
- [17] Eriksson, I., Hellman, R., and Nurminen, M. I. (1988). A Method for Supporting Users' Comprehensive Learning. *Education & Computing*, 4(4), pp. 251-264.
- [18] Gasser, L. (1986). The Integration of Computing and Routine Work. *ACM Transactions of Office Information Systems*, 4(3), pp. 205-225.
- [19] Tyre, M.J. and Orlikowski, W.J. (1994). Windows of Opportunity: Temporal Patterns of Technological Adaptation in Organizations. *Organization Science*, 5(1), pp. 98-118.
- [20] Zuboff, S. (1988). *In the Age of the Smart Machine - The Future of Work and Power*. Oxford: Heinemann Professional Publishing Ltd.

- [21] Boland, R.J. Jr. (1991). Information Systems Use as a Hermeneutic Process, in *Information Systems Research: Contemporary Approaches and Emergent Traditions* (eds. H.-E. Nissen, H.K. Klein and R. Hirschheim), Elsevier Science, pp. 439-458.
- [22] Kogut, B. and Zander, U. (1992). Knowledge of the Firm, Combinative Capabilities, and the Replication of Technology. *Organization Science*, 3(3), August, pp. 383-397.
- [23] Lyytinen, K.J. and Ngwenyama, O. K. (1991). What Does Computer Support for Cooperative Work Mean? A Structural Analysis of Computer Supported Cooperative Work. *Accounting, Management & Information Technologies*, 2(1), pp. 19-37.
- [24] Orlikowski, W. and Robey, D. (1991). Information Technology and the Structuring of Organizations. *Information Systems Research*, 2(2), pp. 143-169.
- [25] Daft, R.L., Lengel, R.H., and Trevino, L.K. (1987). Message Equivocality, Media Selection, and Manager Performance: Implications for Information Systems. *MIS Quarterly*, 11(3).
- [26] Käkölä, T. and Koota, K.I. (1997). Redesigning Computer-Supported Work with Dual Information Systems: the Work Process Benchmarking Service. Forthcoming in *Journal of Management Information Systems*.
- [27] Malone, T.W. (1985). Designing Organizational Interfaces. *Proceedings of CHI '85 Human Factors in Computing Systems*, pp. 66-71. ACM, New York.
- [28] Bush, V. (1945). As We May Think. *The Atlantic Monthly*, 176(1), pp. 101-108.
- [29] Engelbart, D. (1963). A Conceptual Framework for The Augmentation of Man's Intellect, in *Vistas in Information Handling* (eds. P. W. Howerton and D. C. Weeks), Cleaver-Hume Press, London, 1, pp. 1-29.
- [30] Nelson, T. (1987). *Literary Machines*. The Distributors, Indiana.
- [31] Nielsen, J. (1990). *Hypertext and Hypermedia*. Academic Press Inc, San Diego.
- [32] Chang, A-M., Holsapple, C. W., and Whinston, A. B. (1989). A Decision Support System Theory. Working paper. University of Arizona, Tucson, USA.
- [33] Chang, A-M., Holsapple, C. W., and Whinston, A. B. (1993). Model Management Issues and Directions. *Decision Support Systems*, 9(1), pp. 19 - 37.
- [34] Conklin, J. (1987). Hypertext: An Introduction and Survey. *IEEE Computer*, 20(9), pp. 17 - 41.
- [35] Roos, L. L. Jr. and Starke, F. A. (1981). Organizational Roles, in *Handbook of Organizational Design, 1: Adapting organizations to their environments* (eds. P. C. Nyström and W.H. Starbuck), pp. 290-308. Oxford University Press, Oxford.
- [36] Stryker, S. and Statham, A. (1985). Symbolic Interaction and Role Theory, in *Handbook of Social Psychology*, 1 (eds. G. Lindzey and E. Aronson), Random House, New York.
- [37] Spendolini, M.J. (1992). *The Benchmarking Book*. American Management Association, New York.
- [38] Heidegger, M. (1977). *The Question Concerning Technology*. Harper & Row, New York.
- [39] Winograd, T. and Flores, F. (1986). *Understanding Computers and Cognition: A New Foundation for Design*. Ablex Publishing Corporation, Norwood, NJ.
- [40] Gerson, E.H. and Star, S.L. (1986). Analyzing Due Process in the Workplace. *ACM Transactions of Office Information Systems*, 4(3), pp. 257-270.
- [41] Wynn, E. (1991). Taking Practice Seriously, in *Design at Work* (eds. J. Greenbaum and M. Kyng), Lawrence Erlbaum Associates, pp. 45-64.

- [42] Bansler, J.P., and Bødker, K. A. (1993). Reappraisal of Structured Analysis: Design in an Organizational Context. *ACM Transactions on Information Systems*, 11(2), pp. 165-192
- [43] Curtis, B., Kellner, M., and Over, J. (1992). Process Modeling. *Communications of the ACM*, 35(9), pp. 75-90.
- [44] El Sawy, O.A. and Khorshid, H.S. (1994). A Design Theory of Virtual Workflows. Research Paper IOM 94-13, School of Business Administration, University of Southern California.
- [45] Armenise, P., Bandinelli, S., Ghezzi, C., and Morzenti, A. (1993). A Survey and Assessment of Software Process Representation Formalisms. *International Journal of Software Engineering and Knowledge Engineering*, 3, 3, pp. 410-426.
- [46] Käkölä, T. (1996). Evaluation of an Embedded Application System for Supporting Organizational Working and Learning. Manuscript submitted for publication.
- [47] Rein, G. (1992). Organization Design Viewed as a Group Process Using Coordination Technology. Ph.D. dissertation. MCC Technical Report CT-039-92, Austin, Texas.
- [48] Koota, K. (1995). Tietojärjestelmien käyttäminen CSCW-prosesseissa organisaation oppimisen ja tiedon luonnin tukena. Pro Gradu-tutkielma. Turun Yliopisto, Tietojenkäsittelyoppi.
- [49] Koota, K.I. and Käkölä, T. (1996). The Knowledge Creation Net-process modeling language. Manuscript.
- [50] Peterson, J.L. (1981). Petri Net Theory and the Modeling of Systems. Prentice Hall, Englewood Cliffs, NJ.
- [51] Hammer, M. and Champy, J. (1993). Reengineering the Corporation: A Manifesto for Business Revolution. HarperCollins Publishers, New York.
- [52] Jennings, R. (1993). Using Access™ for Windows™. Que Corporation, Carmel, IN.
- [53] Hammer, M. (1990). Reengineer Work: Don't Automate, Obliterate. *Harvard Business Review*, 68, pp. 104-112.
- [54] Käkölä, T. (1995b). Designing and Deploying Coordination Technologies for Fostering Organizational Working and Learning: From Vision to Reality? *Scandinavian Journal of Information Systems*, 7(2), pp. 45-74.
- [55] Vanharanta, H., Käkölä, T., and Back, B. (1995). Validity and Utility of a Hyperknowledge-Based Financial Benchmarking System. *Proceedings of the 28th Annual Hawaii International Conference on System Sciences*, Vol. 3, pp. 221-230. IEEE.
- [56] Vanharanta, H., Käkölä, T., and Kangas, K. (1995). Usability of a Hyperknowledge-Based Executive Support System for Financial Benchmarking. *Proceedings of the 28th Annual Hawaii International Conference on System Sciences*, Vol. 3, pp. 130-139. IEEE.