

Integrating Artificial Intelligence into Human Resource Knowledge Systems for Intelligent Management

Dr Debi Prasad Das

Associate Professor, KIIT School Of Management, KIIT University Bhubaneswar

*Corresponding author Dr Debi Prasad Das.

DOI: [https://doi.org/10.63001/tbs.2026.v21.i03.S.I\(3\).pp1780-1810](https://doi.org/10.63001/tbs.2026.v21.i03.S.I(3).pp1780-1810)

KEYWORDS

Decision Support Systems, Data Warehousing, Knowledge Management, Knowledge Warehouse, Artificial Intelligence, Human Resource Management, Organizational Knowledge, Information Infrastructure, Decision Making, Knowledge Sharing

Received on:

25-08-2026

Accepted on:

13-09-2026

Published on:

23-09-2026

Abstract

Decision support systems (DSS) are increasingly vital to the daily operations of modern organizations. Data warehousing provides the infrastructure to extract, cleanse, and store large volumes of data, enabling knowledge workers to make fact-based decisions. However, much of a firm's intellectual capital resides in the minds of employees rather than in digital systems. To address this gap, a new generation of knowledge-enabled systems is required, capable of capturing, organizing, leveraging, and disseminating both data and tacit knowledge. This paper proposes an extension of the traditional data warehouse model through a knowledge warehouse (KW) architecture. The KW framework facilitates the coding, retrieval, and sharing of organizational knowledge, thereby enhancing decision quality across the enterprise. This approach signals a new direction for DSS, emphasizing knowledge improvement as a central purpose. Future DSS effectiveness will be measured not only by data processing capacity but also by how well systems promote knowledge, strengthen decision makers' mental models, and improve organizational decision making.

Introduction

The complexities of decisions in the information age compel every manager to utilize information analysis tools for supporting business decisions. Over the last three decades, the organizational role of information technology has evolved from efficiently processing large amounts of batch transactions to providing information in support of decision-making activities. This paradigm shift is reflected in the fact that in the 1970s most IS organizations

changed their name from “data processing” to “management information systems” [7]. In addition, the variability, interdependency and uncertainty of factors affecting decision-making process are complex. Decision support systems (DSS) are interactive, computer-based systems intended to provide support to the decision makers engaged in solving various semi- to ill-structured problems involving multiple attributes, objectives and goals. Decision support systems are becoming increasingly more critical in the daily operation of organizations. With the

evolution of enterprise network computing, client/ server architecture, and a set of significant new information processing concepts, it is now possible for organizations to provide the key people in the firm with access to needed information and the means to utilize that information in a decision support context. Since the mid-1980s data warehouses have been developed and deployed as an integral part of a modern decision support environment. A data warehouse provides an infrastructure that enables businesses to extract, cleanse, and store vast amounts of corporate data from operational systems for efficient and accurate responses to user queries [26]. A data warehouse empowers knowledge workers with information that allows them to make decisions based on a solid foundation of fact [12]. However, only a fraction of the required knowledge exists on computers; the vast majority of a firm's intellectual assets exist as knowledge in the minds of its employees [48]. Hence, a data warehouse does not necessarily provide adequate support for knowledge intensive queries in an organization. What is needed is a new generation of knowledge-enabled systems that provides the infrastructure required to capture, enhance, store, organize, leverage, analyze, and disseminate not only data and information but also knowledge. The existing enterprise-wide data warehouses can be extended to create a knowledge warehouse (KW). This

warehouse can be used as a clearinghouse of knowledge to be used throughout the organization by the employees to support their knowledge intensive decision-making activities. The KW can also evolve over time by enhancing the knowledge it contains.

Just as in a data warehouse environment where data mining techniques can be used to discover untapped patterns of data that enable the creation of new information, by extension then, use of technologies such as data warehousing, data mining and other artificial intelligence (AI) technologies can enhance the knowledge creation, storage, dissemination and management processes [2]. However, for an effective knowledge warehouse to become a reality, different types of knowledge (i.e., both tacit and explicit knowledge) and different forms of knowledge (e.g., text streams, binary large objects, production rules, mathematical models, and what-if cases) need to be captured, codified, and cataloged. In addition, this codified knowledge must contain knowledge about itself (meta-knowledge) and must be analyzed to create new knowledge.

The purpose of this paper is to describe the processes required for developing a knowledge warehouse and to propose, as an extension to the data warehouse model, a knowledge warehouse architecture that can facilitate the capturing, coding, retrieval and

sharing of knowledge. The KW is used to enhance the generation of new knowledge throughout the organization. The primary goal of a KW is to provide the knowledge worker with an intelligent analysis platform that enhances all phases of the knowledge management process. Just as the emergence of data warehouses a decade ago signaled a new direction for the DSS, we argue that the knowledge warehouse proposed here suggests a new and evolving direction for DSS in the next decade. This new direction is based on an expanded purpose of DSS. That is, the purpose of DSS in knowledge improvement; i.e., enhanced learning. This expanded purpose of DSS also suggests that the effectiveness of each DSS will, in the future, be measured based on how well it promotes and enhances knowledge, how well it improves the mental model(s) and understanding of the decision maker(s) and thereby how well it improves his/her decision making.

The remainder of this paper is organized into the following sections. In Section 2, we provide some knowledge management background. In Section 3, we discuss how DSS, artificial intelligence (AI) and information technology (IT) can enhance knowledge management. We then present the foundations for the goals and requirements for a KW in Section 4. In Section 5, we present the proposed KW architecture. In Section 6, we discuss guidelines

as how such a ware-house could be implemented. In Section 7, we provide a roadmap for future DSS research based on our proposed architecture. Finally, in Section 8, we provide our summary and conclusions.

Knowledge management

Knowledge management is the practice of adding actionable value to information by capturing tacit knowledge and converting it to explicit knowledge; by filtering, storing, retrieving and disseminating explicit knowledge; and by creating and testing new knowledge. In this context, tacit knowledge includes the beliefs, perspectives, and mental models so ingrained in a person's mind that they are taken for granted [48]; it consists of subjective expertise, insights and intuitions that a person develops from having been immersed in an activity or profession for an extended period of time. On the other hand, explicit knowledge is knowledge that can be expressed formally using a system of language, symbols, rules, objects, or equations, and can thus be communicated to others; it consists of quantifiable data, written procedures, universal principles, mathematical models, etc. [10,48,72].

New knowledge is created through the synergistic relationship and interplay between tacit and explicit knowledge [48], specifically, through a four-step process of socialization, articulation, integration, and understanding/internalization (Fig. 1). Socialization

is the process of sharing with others the experiences, technical skills, mental models, and other forms of tacit knowledge. For example, apprentices learn a craft not through language, but by working with their masters; i.e., observing, imitating and practicing under the master's tutelage. On-the-job-training provides this mode of sharing tacit knowledge in the business world. OJT is complemented with film clips of the expert performing the task, virtual reality representations, and kinematic analysis (from the

field of robotics).

Articulation is the process of converting tacit know-ledge to explicit knowledge. In the decision-making process, articulation may include, but is not limited to, one or more of the following: (1) specifying the pur-pose of the decision; e.g., to understand how the num-ber and locations of warehouses influence supply costs in a new marketing area, (2) articulating parameters, objective functions, relationships, etc., in a DSS math-

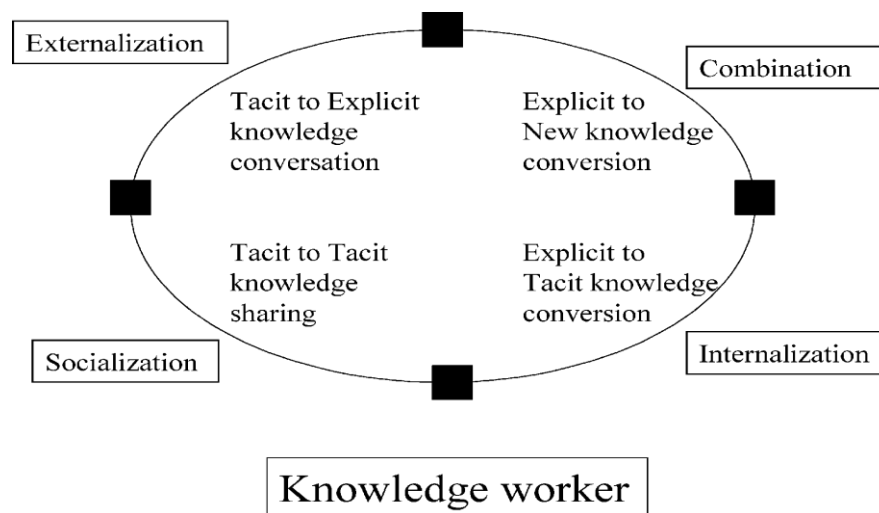


Fig. 1. The knowledge spiral.

emational model (i.e., building a model), (3) articulating 'what-if' model cases that reflect existing and potential decision making situations, and (4) evaluating the de-cision alternatives, given the uncertainty in the deci-sion making environment. In other situations (e.g., those requiring the analysis of complicated physical movements), articulation may take the form of

kine-matic analysis; i.e., attaching sensors to various key appendages and then digitizing and recording the movements of interest. Articulation may also include knowledge extraction in expert systems, determination of causal maps, brainstorming, etc.

Integration is the process of combining several types of explicit knowledge into new

patterns and new re-lations. The Gestalt theory of learning literature states that “all problems with which we may be confronted, and also the solutions of such problems, are matters of relations; not only does our understanding of the prob-lem demand our awareness of certain relations, we cannot solve the problem without discovering certain new relations” [51]. One potentially productive inte-gration of explicit knowledge is the analysis of multi-ple, related ‘what-if’ cases of a mathematical model to find new relationships, or meta-models, that determine the key factors of the model and show how these key factors interact to influence the decision [62].

Understanding is the process of testing and validat-ing the new relationships in the proper context, thereby converting them into new tacit knowledge. Perkins’s [51] theory of understanding, from the theory of learn-ing literature, suggests that understanding involves the knowledge of three things: the purpose of the analysis (i.e., what the decision maker wants to understand), a set of relations or models of the process/system to be understood, and arguments about why the relations/ models serve the purpose. Internalization is the process of using the new patterns and relations, together with the arguments of why they fit the purpose, to update and/or extend the decision maker’s own tacit knowl-edge base, thus creating a spiral of

learning and knowl-edge that begins and ends with the individual [23,48].

For a more comprehensive review of tacit to explicit knowledge conversion, please refer to Refs. [48,11].

DSS, IT, and AI support of knowledge management

DSS, IT, and AI can all be used to enhance knowl-edge management and its knowledge conversion pro-cesses: i.e., tacit to tacit knowledge sharing, tacit to explicit knowledge conversion, explicit knowledge leveraging, and explicit to tacit knowledge conversion. These process enhancements are discussed individu-ally below (Fig. 2).

3.1. Sharing tacit knowledge

One of the primary potential applications of infor-mation technology to sharing tacit knowledge is the use of digitized filming of the physical demonstration of a process. Once stored, this digitized film clip can be made available on the internet for anytime, anyplace viewing. The film clip can also include slow motion segments of the physical process where applicable, complete with verbal explanations included within the clip to enhance the understanding of the process being demonstrated.

A potential application of artificial intelligence to tacit to tacit knowledge sharing is the use of kinematic analysis of the physical process. Kinematics includes the use of reflective

dots and/or sensors attached to the various appendages and joints of the demonstrator to enhance the determination of quick or subtle move-ments or actions during the demonstrated process; e.g., to detect twisting or turning of the fingers while a master chef kneads bread dough.

Once the process is recorded, kinematic analysis software is used to further analyze the relative motion of the appendages and joints; thus, kinematics provides a natural conversion of tacit knowledge to explicit knowledge.

3.2. Converting tacit knowledge to explicit knowledge

In tacit to explicit knowledge conversion, the literature of knowledge acquisition in expert systems (ES) provides both guidance and techniques [35]. Knowledge acquisition involves employing various

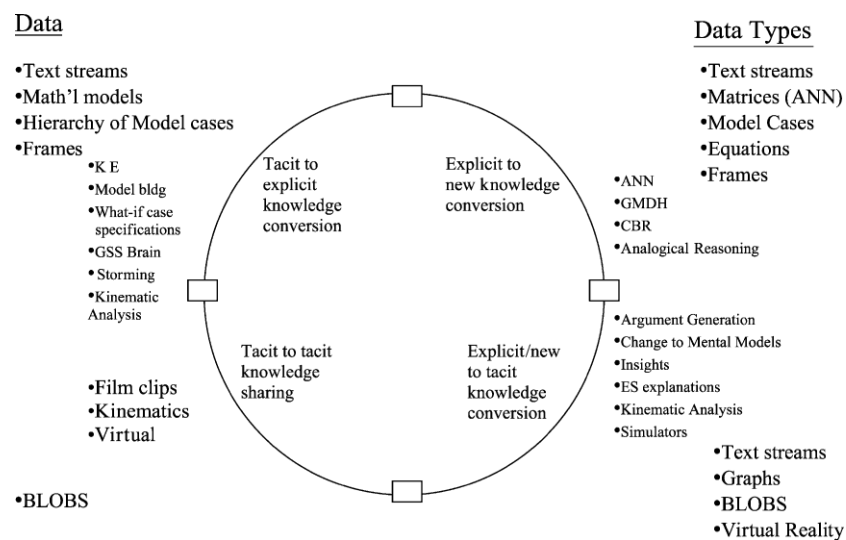


Fig. 2. Technologies and data/knowledge types in knowledge management.

techniques to elicit information (verbal and/or quanti-tative) from the knowledge worker, interpreting this information (more or less skillfully) in order to infer the underlying knowledge and reasoning processes, and using this interpretation to guide the construction of some model or language that describes (more or less accurately) the knowledge worker's performance [28].

DSS can also enhance the tacit to explicit knowl-edge conversion through the specification of mathe-matical models. Specifically, in the model building process (e.g., in linear programming models) the knowledge worker is asked to explicitly specify the goal or objective of the model, the decision variables, and perhaps the

relative importance of the decision variables (in the case of a goal programming model). The knowledge worker also explicitly specifies the model constraints in terms of the decision variables, and estimates both the numerical coefficients of the decision variables in each constraint and in the objec-tive function, as well

as the right hand side constraint values. The explicit knowledge reflected in these model components (decision variables, coefficients, con-straints and objective functions) reflects the tacit knowledge built up over the years of being immersed in the decision making environment. The resulting models may be stored in the form of a set of explicit mathematical inequalities [16], as annotated graphs of arcs and nodes in network flow models [29,64], as a set of arc descriptions [34] or as a condensed canonical model formulation with links to relational tables for instantiation [63].

DSS can also enhance the tacit to explicit knowledge conversion by eliciting one or more what-if cases (i.e., model instances) representing situations that the knowledge worker wants to explore. As the knowledge worker changes one or more model coefficients or right hand side values (e.g., in a linear programming model) to explore its effect on the modeled solution, s/he is estimating ranges of those parameters/values that reflect the actual and/or potential decision making environment represented by the model. That is, the knowledge worker is converting the tacit knowledge of various historical situations and/or decisions into explicit knowledge that can be shared with other workers and leveraged to enhance decision making. These multiple, related model instances can be stored, along with their associated solutions, as tuples in a relational

database, as objects in an object-oriented database, or as sparse matrices.

Another source of tacit to explicit knowledge conversion occurs in the brainstorming of GSS. GSS brain-storming sessions solicit the participants' ideas and concerns about a stated problem. The ideas are then anonymously relayed (without evaluative comments) to the other participants for their enhancements and modifications, generating a continual stream of related and tangential ideas directed toward solving the stated problem. At some point of time, the session leader directs the participants to stop generating new ideas and start evaluating, again anonymously, a specific idea. The evaluations are given in the form of short lists of things the participant likes about the idea and also short lists of concerns that may hamper implementation. The group then addresses the concerns, evolving toward a valid and implementable solution to the stated problem. The ideas, likes and dislikes of GSS brainstorming sessions are stored as text streams for sharing, processing and future use.

3.3. Knowledge leveraging: converting explicit knowledge to new knowledge

Once a knowledge worker's tacit knowledge is converted to explicit knowledge and stored in an appropriate (computer readable) form, it can be leveraged by making it available to others when and where they need it. In

addition, analyzing explicit knowledge to produce new knowledge can further leverage it.

For example, explicit knowledge generated from GSS brainstorming sessions and stored as text streams can be analyzed by text mining software, a form of AI-based data mining, to provide key words, related concepts, clusters of similar ideas, etc. The traditional approach to text mining is based on searching the document and counting the number of occurrences of a given word in the document. AI-based search methods use an inductive learning algorithm to determine the key words and extract the appropriate statistical information from the textual documents [27,42]. An alternative text mining approach is information extraction, which finds specific information in a textual document according to a predefined set of rules and guidelines which are specific to a given topic area [69]. Commercially available text mining software packages include CRYSTAL [58], RAPIER [9], and AutopSlog [53].

On the other hand, explicit knowledge stored in the form of instances of a mathematical model (what-if cases) can be leveraged via deductive and/or inductive model analysis systems. Here, deductive model analysis systems (DMAS) apply paradigm-or model-specific knowledge to a single instance of the model, addressing such questions as “Why is this the solution?,” “Why do the solutions to two model

instances differ so much?,” or, in the case of linear programming models, “Why is this instance infeasible?” Deductive model analysis systems exist for each of the three major modeling paradigms: linear programming, simulation and spreadsheet models [20,21,37,38,41].

On the other hand, inductive model analysis systems (IMAS) operate on a set of many related model instances that represent historical situations familiar to the knowledge worker and/or several (if not many) what-if cases. The primary goal of IMAS is to help the knowledge worker develop insight(s) into the business environment represented by the model [56]. IMAS are distinguished from deductive analysis systems by both the required input and the type of processing logic employed; i.e., IMAS apply inductive analysis technologies (e.g., statistical analysis, the group method of data handling, genetic algorithms) to extract new knowledge in the form of key factor identification, simplified meta-model generation, etc. [52,54 – 56, 62,65,66].

Another form of explicit knowledge leveraging is found in case-based reasoning (CBR). CBR is characterized by the knowledge worker making his or her inferences and decisions based directly on previous cases recalled from memory [40]. That is, the knowledge worker tries to avoid, or reduce, the potential for failure

by recalling previous similar failures and avoiding the associated pitfalls or changing key factors in those previous failures. S/he can also speed the decision-making process by not having to generate and evaluate all alternatives from scratch. Finally, the attributes of past cases can be generalized to improve decision making in the future [22]. CBR requires case storage capabilities (perhaps in the form of frames), a filtering of cases for relevancy of key factors, a sophisticated recall capability based on key factors, and a case-based inference capability based on those parts of the previous case which are appropriate for the current decision.

3.4. Learning new knowledge: converting explicit knowledge to implicit knowledge

DSS/IT/AI can also provide valuable aids in internalizing explicit and new knowledge; i.e., in helping the knowledge worker to learn. One mode of internalizing explicit and/or new knowledge is through the modification of the internal mental model that a knowledge worker uses to serve as a performance guide in specified situations. Such mental model modifications may occur in the building of a DSS model. For example, a knowledge worker might modify his or her mental model based on the discovery of new relationships between key factors during model development, the development of counterexamples of assumed relationships, and/or the acknowledgement of fallacies in deductive logic uncovered during modeling.

Another source of mental model modification may be the adjustment of the relative importance of various components of the mental model. DMAS can be helpful here; e.g., sensitivity analysis offered in some types of mathematical modeling (i.e., linear programming models) can be used to help the knowledge worker understand and alter the relative importance of key parameters and how incremental changes in one parameter can affect the solutions [19,20].

A third source of mental model modification may come from the inductive analysis of multiple, related solved model instances. For example, if several model instances are specified in which two or more uncertain parameters are varied over appropriate ranges, an analysis of the multiple solved instances may provide new knowledge concerning not only the relative importance of key factors, but also how the key parameters interact, perhaps in a nonlinear fashion, to affect the model solution [56,71].

Another aid in internalizing explicit knowledge is provided by expert systems. Here the explanation capability of ES provides understandable and amplifying rationale(s) for a recommended course of action.

Still another way that DSS can help the knowledge worker internalize explicit knowledge

is to enhance his understanding of the knowledge. Understanding, according to Perkins's [51] theory of learning, consists of knowing three things: (1) the purpose of the analysis, or what the knowledge worker wants to understand, (2) a design, or hypothesized (mathematical) model, of the process/system to be understood, and (3) arguments about why the design serves the purpose. These arguments can be of three different types. Evaluative arguments focus on the accuracy, sufficiency, necessity and consistency of a proposed model and its components. Simple explanatory arguments focus on explaining or defining the elements of the model and/or state what each element contributes. And, finally, deep explanatory arguments seek to explain a design or model in terms of basic underlying principles; e.g., the underlying formulae and interconnections between the balance sheet, income statement and funds flows statement in a business financial problem. The advantages of deep explanatory arguments include their power of abstraction, generalization, and insight generation, resulting from the application of basic principles and relations applicable to the current analysis. The basic disadvantage of deep explanatory arguments is the difficulty of defining, storing and retrieving relevant basic principles, relating these basic principles to the model, and successfully communicating the relationships to

the knowledge worker [29,31,38]. Thus, this type of analysis requires not only the storage of multiple, related model cases, but also the storage, retrieval and processing of the purpose and underlying principles potentially applicable to the specific decision making environment, stored as text streams and referenced through key words and context.

4. Goals and requirements for knowledge warehousing

The goal of KW is to provide the decision maker with an intelligent analysis platform that enhances all phases of the knowledge management process. Several comments can be made to further amplify and explain the KW goal.

First, this goal assumes that the user of the KW is the decision maker. That is, we assume that the user is not an expert in the various technologies used to enhance knowledge management, but rather is an expert in the decision making field.

Second, an intelligent analysis platform is defined as a PC-based platform that makes available to the decision maker an array of analytical tools, each of which utilizes various technologies to aid the socialization, articulation, integration, and understanding/ internalization of knowledge management. The purpose of including artificial intelligence is to amplify the cognitive capabilities of the decision maker in

converting tacit knowledge into explicit knowledge, integrating this explicit knowledge by analyzing it to detect new patterns and relations, and understanding the new knowledge by providing analogs and explanations.

Third, AI technologies are often able to find important facts, patterns, relations and/or other types of new knowledge that would not have been found using standard analysis techniques such as regression analysis. The new knowledge gained can then be used to aid decision makers in determining organizational action [57]. One applicable AI technology is data mining, a process that can be divided into two distinct categories—verification-driven and discovery-driven. In verification-driven data mining, a prior hypothesis is formed about the nature of relationships among data. The result of the mining process is then used to reach a conclusion regarding the validity of this hypothesis. Discovery-driven data mining starts without any pre-conceived notion regarding the nature of relationships among data. It is the task of the data mining system to find significant patterns in the data. Two sub-categories of discovery-driven data mining are supervised learning (classification) and unsupervised learning (clustering) [15]. Supervised learning is equivalent to learning with a teacher and involves building a model for the specific purpose of optimally predicting some target field in the

historical database (the value of which can be used to gauge whether the right or wrong prediction was made). In contrast, unsupervised learning does not have any well-defined goal or target to predict (and, thus, no particular supervision over what is a right or wrong answer). Techniques such as clustering and detection of association rules fall into the category of unsupervised learning [7].

The knowledge warehousing goal suggests three functional requirements for KW: (1) an ability to efficiently generate, store, retrieve and, in general, manage explicit knowledge in various forms, (2) an ability to store, execute and manage the analysis tasks and their supporting technologies with minimal interaction and cognitive requirements from the decision maker, and (3) an ability to update the KW via a feedback loop of validated analysis output. Each of these three functional requirements is discussed individually below.

Knowledge storage and retrieval

The KW must provide the same services for knowledge that a data warehouse provides for data. This requirement is complicated in the KW by the several different forms of knowledge feeds. That is, the primary source of data in data warehouses is transaction data (easily stored in a relational database), but the primary sources of knowledge in the knowledge warehouse include

text streams from GSS and ES, film clips (stored as binary large objects or BLOBs), mathematical models and their instances (stored as equations, matrices, arc/node descriptions, etc), and analysis results (stored as equations, weight matrices, text streams, etc). Further, for knowledge stored in the form of models and solved model instances, the KW is required to efficiently store, retrieve and manipulate many solved model instances, with each instance tied (logically) to its associated model and/or tied (logically) to a related instance; i.e., two related instances normally exhibit a high degree of commonality in parameter values and can thus be stored and retrieved more efficiently if logically related (e.g., through inheritance).

Analysis task management

The analysis of knowledge is not a simple process. Specifically, an analysis task frequently utilizes various inductive and deductive AI technologies; e.g., neural networks, group method of data handling (GMDH) [6], statistics, inductive production rule generation, genetic algorithms, case-based reasoning. Each task has its own requirements with respect to

- (1) input data (e.g., the number and domain coverage of stored data or knowledge),
- (2) execution parameters required by the analysis technologies (e.g., step-size and node architecture for neural networks, the complexity factor and

number of layers for GMDH), and

- (3) output format (e.g., weight matrix, polynomial equations, production rules, quality measures). Further, some analysis technologies are limited to specific knowledge paradigms, whereas others are equally applicable to all paradigms; e.g., the explanation task implemented in ROME/ERGO [41] is limited to spreadsheet models, whereas the meta-model generation implemented in INSIGHT is applicable to all mathematical models [56].

KW must efficiently support the storage, initiation, execution and management of knowledge analysis tasks and the associated implementation technologies. Specifically, the analysis tasks and the associated technologies must not only be stored in KW, but also be logically tied to the appropriate knowledge paradigm, if required. Further, to minimize the cognitive requirements of the decision maker during analysis task execution, the required runtime interaction (e.g., appropriate step size in neural network models, complexity factors in GMDH, etc.) must be stored in the knowledge warehouse and retrieved as appropriate during task execution.

4.3. Feedback and storage of new knowledge

In the operation of a data warehouse, data in the warehouse is updated only periodically (say weekly or monthly) with new data from the

transaction processing system. However, in the operation of a KW, the data and knowledge stored in the warehouse can be updated constantly from either of two different feed-back loops: one loop associated with on-line knowledge extraction

(e.g., a GSS brainstorming session), and the other loop from a real-time storage request of the decision maker/user based on the results of an analysis task s/he has validated and approved. The KW must support both feedback loops.

Knowledge warehouse architecture

These goals and requirements of a KW can be implemented via an extension of the data warehouse architecture. The proposed extension, shown in Fig. 3, consists of six major components: (1) the data/knowledge acquisition module, (2) the two feedback loops, (3) the extraction, transformation and loading module, (4) a knowledge warehouse (storage) module, (5) the analysis workbench, and (6) a communication manager/user interface module. Each of these components is described below.

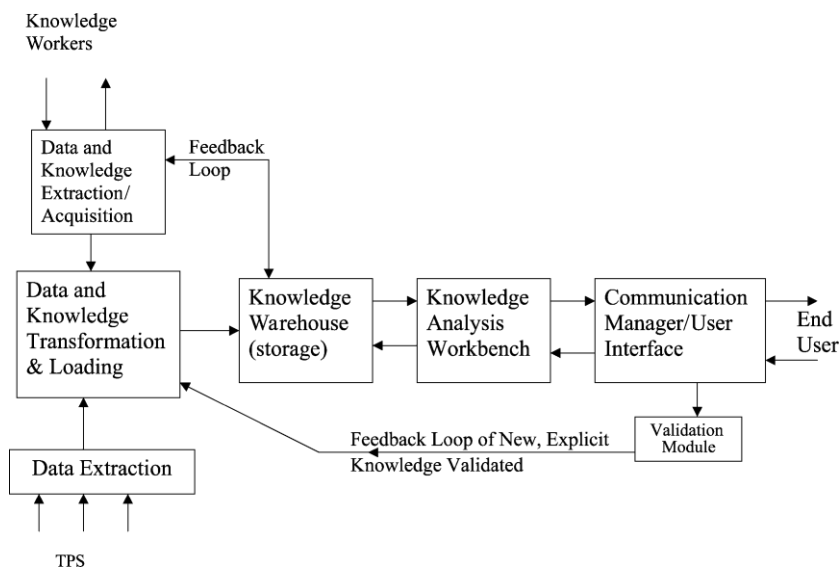


Fig. 3. Knowledge warehouse architecture.

Knowledge acquisition module

The knowledge acquisition module is primarily responsible for the tacit to explicit knowledge con-version; i.e., directly acquiring tacit knowledge from the decision maker/user. This acquisition module includes a specialized

user interface to aid in one or more of the following processes: (1) idea generation in a GSS brainstorming environment, (2) mathematical model specification in a model-based environment, (3) what-if case specification in a model-based environment, (4) production rule elicitation in an expert system-based

environment, (5) purpose and fundamental knowledge elicitation in any analysis process [12,62] kinematic analysis in a physical process demonstration, etc.

Feedback loops

Note that there is one a feedback loop between the knowledge acquisition module and the KW storage module (via the knowledge loading module). This feedback loop provides the capability of not only storing the explicit knowledge elicited from the decision maker(s), but also of immediately broadcasting knowledge from one user to other users (in a GSS brainstorming session), displaying up-to-date lists of specified what-if cases (in a model-based DSS), or displaying current rule bases (in ES-based systems). The other feedback loop that exists between the extraction, transformation and loading module and the communication manager module provides for the storage of new validated explicit knowledge that has been generated in the system.

5.3. Knowledge extraction, transformation and load-ing module

The knowledge extraction, transformation and load-ing module is similar to that in the data warehouse in that it is responsible for extracting, reformatting, clean-sing and loading data from external databases into the KW storage area (see Ref. [18]).

5.4. Knowledge warehouse storage module

One of the primary components of the KW architecture is an object-oriented knowledge base management system (KBMS) that integrates the knowledge base, model base, and analysis tasks. A KBMS is a system that manages the integration of a wide variety of knowledge objects into a functioning whole. These knowledge objects include numerical data, text streams, validated models, meta-models, movie clips, animation sequences, as well as the software used for manipulating them. The KBMS is implemented in an object-oriented environment. The KBMS must not only manage data, but all of the objects, object models, process models, case models, object interaction models and dynamic models used to process the knowledge and to interpret it to produce the knowledge base.

Object-specific knowledge is stored as part of the appropriate object. The specific form of the knowledge storage mechanism may include frames, semantic nets, rules, etc. Stores of knowledge include, but are not limited to, meta-data, meta-models and instances of meta-models. For example, a model's purpose is stored as part of the associated model, whereas the basic underlying principles may be stored with a more general model class.

Messages sent to the objects are generic in

form, independent of the method's technology. If additional information is required to execute a specified method, a message is sent to other appropriate object(s).

The object-oriented database technology provides several advantages for this application. One advantage is that existing knowledge is integrated with (1) its own meta-knowledge, (2) examples or instances of the knowledge, and (3) methods, including the analysis tasks. This enhances storage efficiency; e.g., if the knowledge is in the form of a model and its instances, related instances may differ from a base case by only one or two parameter values and the solution vector, and all common parameter values can be inherited from the base case or other parent instance for storage efficiency. A second advantage is that some analysis tasks (e.g., the linear programming sensitivity analysis task in ANALYZE) can be logically tied to a specific class of models, whereas other analysis tasks can be tied to a superclass of all models and be independent of the specific modeling paradigms. A third advantage is that method over-loading allows a single user-specified command to call several different implementations of a given task and apply the appropriate technology to different forms of knowledge; this reduces the cognitive burden on the decision maker by providing him/her with independent execution calls (i.e., messages) for

all analysis tasks. It also provides a primary prerequisite for effective management of technology; i.e., over-loading, in conjunction with encapsulation, makes the changing of implementation technologies transparent to the user.

Knowledge analysis workbench

The analysis workbench handles all interaction with the analysis tasks, including task control, argument generation, and management of technology. The task controller handles all requests for data and run-time interactions (e.g., complexity factors in GMDH algorithms, step sizes in neural networks) required by the analysis technologies. That is, the task controller acts as an AI-based surrogate decision maker for task interactions, shielding the real decision maker from the requirements of knowing the technologies, their nuances, interactions, etc.

The argument generation sub-module evaluates the outputs of the various analysis tasks, especially the causation task, filtering out implausible or inconsistent results based on relative measures of accuracy, simplicity, conceptual validity, sufficiency, necessity, and consistency. It then generates simple and deep explanatory arguments that (hopefully) enhance the decision-maker's understanding of the modeled environment. In generating these arguments, the argument generation module

interfaces with the knowledge base, the instance base and model base, applying deductive knowledge, analogical reasoning, and other technologies, as appropriate.

The management of technology module manages the repository of analysis technologies. Specifically, it provides for the encapsulation of new analysis algorithms into object model classes, integration of legacy data mining applications, incorporation of new analytical models and meta-models into the object model repository, etc.

5.6. Communication manager

This module, which handles all analysis communication between KBMS and the user interface, includes six functional sub-modules: a knowledge engineer, what-if interface, query processor, results presentation manager, on-line help, and user interface. The knowledge engineer sub-module is an expert system-based sub-system responsible for interacting with the decision maker to develop the purpose of the analysis and the basic underlying principles of the modeled environment. Both types of knowledge are used in the development of arguments. This knowledge may be stored in the knowledge base in the form of frames, rules, semantic nets, etc.

The what-if interface is designed to efficiently and effectively help the decision maker specify one or more what-if cases to be

investigated. It includes an analogical component that is used to suggest pertinent instances by varying one or more parameter values. It also includes one or more interactive graphical displays, or summaries, of instances already available, so that the decision maker can see at a glance what has already been tried and what instance(s) might lead to additional insights. The what-if interface also includes a capability to suggest potentially valuable cases based on the planning analysis task.

The query processor provides the interface between the decision maker and the analysis task. It translates natural language, QBE or SQL-like queries specified by the decision maker into machine executable queries. The result representation manager selects the most appropriate presentation view for each analysis result; e.g., graphics [50], natural language production rules, polynomials, decision trees, etc. The selection is based on a combination of the analysis task output and the decision maker's preference which, in turn, is based on an adaptable machine learning algorithm which analyzes previous uses of models and analysis tasks by the current decision maker [14,44].

The help sub-module provides the user with information concerning the model (e.g., assumptions, parameter ranges, units of measurement, internal model structure), instances (differences from base case, key decision variable

values), pertinent knowledge (e.g., meta-models, meta-data, basic principles, analysis purpose), and analysis tasks (e.g., applicable technology, technology description, explanatory traces of results, technical parameters used, advantages and limitations of technologies).

6. Development and implementation of the knowledge warehouse architecture

Development and implementation KW architecture outlined earlier may involve considerable amount of organizational time and effort and may cross the boundaries of many business units and departments. The usual time frame is measured in months and not days and the amount of money involved usually represents millions not thousands of dollars. As with the development and implementation of DSS projects [1 – 4,45,59,60,68], a large-scale KW project may require large investment of time and money that puts a tremendous pressure on those involved. As a result, KW projects may not always be successful. Many of the factors that affect the successful development and implementation of DSS projects can also be important in determining success for KW. Among these factors are support from the top executives [24,43, 45]; users involvement and participation [5,30 – 32,61,67]; well-defined business objectives or goals for the DSS [45,61,68]; resources adequacy issues; organization and political issues within the company [17,43,70];

technological issues [45,61,68]; process management issues [47,49]; goals, plans and communication issues [25]; values and ethics [33] and other external issues.

All of these factors play an important role in successful development and implementation of knowledge warehouses. However, since knowledge warehouses focus on the harnessing of intellectual capital within an organization and making it available to all who need them, additional factors should be considered as well. These factors are due to the additional tasks that knowledge warehouses should perform. They are the following.

(1) Creation of a knowledge management infrastructure. The task involves workstations, networks, databases, search engines, and publishing tools.

(2) Building a knowledge culture by active promotion of the knowledge agenda, including the development and diffusion of knowledge management models, frameworks, and language. This requires the creation of mechanisms for the development and maintenance of knowledge bases in different functions and departments.

(3) Facilitation of knowledge-oriented connections, coordination and communication throughout, and also without, the organization.

Successful development and

implementation of KW architecture requires these generic activities.

1. Designing and implementing techniques to identify and record both knowledge and ignorance (e.g., taking inventory and auditing) and then designing processes to share, use and protect such knowledge and to remedy ignorance by learning or knowledge creation.

2. Designing and orchestrating contexts, environments and activities to discover and release what is not formally or explicitly known (e.g., socializing and experiencing) and possibly coaching and encouraging people to be effective in these processes.

3. Articulating and communicating the purpose and the nature of knowledge management and connecting it to other strategic and operational initiative and activities of the organization.

Although it is beyond the scope of the paper to develop a prototype for the proposed KW architecture, we provide a list of vendors and their products that can be employed to implement the modules of the KW architecture. There are plethora of products available commercially and the list we provide here is not exhaustive by nature. In addition, our goal was not to identify the best vendor or the best product that is available to aid the KW architecture, but rather to review and present some product offerings that

support the various processes of the proposed architecture. To identify the vendors/products, we reviewed several sources such as, KM World, Knowledge Management, KM World Buyer's Guide, Directory of Data Warehousing solution providers, and by searching the web on Knowledge Management and Data Warehousing related product offerings. In reviewing the product offerings, we found that although many products cover each component of the KW architecture individually, there were very few that provided an integrated solution covering all aspects of the KW. (See Exhibit A for a partial list of products that claim to offer integrated solution for the KW architecture.)

There are many other products offered by vendors that may not provide a comprehensive solution but support the implementation of the processes under each component of the KW architecture that we have proposed.

The KW should efficiently generate, store, retrieve and, in general, manage explicit knowledge in various forms to provide the decision maker with an intelligent analysis platform that enhances all phases of knowledge. The knowledge based systems module in our proposed KW architecture helps accomplish that and tools for this module should support processes such as mental model extraction, knowledge engineering and integration for the extraction and storage of various types of

organizational knowledge. (Some examples of vendors and their products available to support the KBMS module of KW architecture can be found in Exhibit A.)

Secondly, the KW should be able to store, execute and manage the analysis tasks and its supporting technologies. Processes such as sensitivity analysis, mining of the knowledge warehouse, machine learning and pattern recognition fall under this component. In addition, capabilities such as hypothesis testing for meta-models should also be available. (See Exhibit A for examples of vendors and their products available to support the Knowledge Analysis Workbench module of KW architecture.)

Finally, the KW should provide computer-assisted support to generate natural language arguments concerning both the comparable validity of the models, meta-models and relations produced by analysis tasks, and how this new knowledge relates to the decision maker's purpose. (See Exhibit A for a sample of vendors and their products that can support communication manager module of our KW architecture.)

Roadmap for future DSS research

In general, DSS has made significant research contributions in knowledge extraction/acquisition with knowledge engineers

of expert systems and the mathematical models of management scientists. DSS has also made significant contributions in the warehousing of data/knowledge and in the communication of results to end users; i.e., databases and user interfaces are principle components of all DSSs.

However, the knowledge spiral proposed by Nonaka and Takeuchi [48] along with the knowledge warehouse proposed herein suggest a different direction for DSS in the next decade. This new direction is based on an expanded purpose of DSS; specifically, the purpose of DSS should be to enhance all four aspects of the knowledge spiral (tacit to tacit knowledge sharing, tacit to explicit knowledge conversion, new knowledge generation, and explicit to tacit knowledge internalization). That is, the purpose of DSS is knowledge enhancement.

In this vein, one research area of DSS becomes the development of a set of theoretical foundations upon which to build future development and applications, one for each quadrant of the knowledge spiral. For instance, cognitive mapping might provide a productive foundation for the tacit to explicit knowledge conversion as a source of study for extracting and analyzing the decision maker's mental models of a given decision making environment. Similarly, the Gestalt theory of insight [39] combined with Newell and Simon's [46] theory of goal directed search might provide a solid foundation for new

knowledge generation, based on the dichotomy of inductive/deductive problem solving and reflecting the hemispheric specificity of the human brain. Cognitive dissonance and Perkin's [51] theory of understanding might provide a valid foundation for explicit to tacit knowledge internalization, the former suggesting that no learning takes place until the decision maker perceives a significant difference between his/her mental model and the real world, and the latter suggesting ways to resolve such perceived differences. And finally, the theory of communication might provide a general foundation for tacit to tacit knowledge sharing.

This expanded purpose of DSS as knowledge enhancement also suggests that the effectiveness of each DSS will, in the future, be measured based on how well it promotes and enhances knowledge, how well it improves the mental model(s) and understanding of the decision maker(s) and thereby how well it improves his/her decision making. One research thrust along these lines, especially applicable to model-based DSS, might include extracting (via case analysis) the novice decision maker's mental model in some decision making environment, comparing it to the (expert's) mathematical model, generating arguments concerning why the mathematical model is superior to the decision maker's mental model, feeding these arguments back to the decision

maker in an attempt to change and improve his/her mental model, and then re-testing the decision maker to determine whether his/her revised mental model produces better decisions. Such a research thrust, based on the lens model [8,36], would provide a missing link in the current research [13].

This expanded purpose of DSS also points out that one of the four quadrants of the knowledge spiral, specifically the tacit to tacit knowledge sharing, has been largely ignored in the DSS literature. This suggests potentially viable research directives in both kinematics for learning physical actions concerned with decision making and a kinematics equivalent for learning-by-doing or on-the-job-training in mental model formulation. Research questions that might be addressed include: How decision making is learned (i.e., by observation, by case analysis, etc.), How many observations or cases are required to learn complex tasks, How transferable such knowledge is, etc.

In addition to the expanded purpose of DSS, the knowledge warehouse architecture proposed in this article shows two major areas in which DSS could/ should foster future research and development. One such area is in providing the motivation, tools, techniques and demonstrated benefits associated with the development and use of the knowledge analysis work-bench. In the DSS literature, especially the

management science aspects of it, the focus of research has historically been on model specification and model solution. In the future, it seems that the analysis of solutions is the more important aspect of modeling, along with providing the decision maker with an understanding of the analysis results. Several DSS researchers have developed some theory in this vein, but the area still needs further refinement. For example, in model-based DSS, we need to identify a 'minimal spanning set' of analysis tasks that leads to successful model analysis, and to validate these tasks through experimentation.

Another research area could explore and evaluate technologies that are potentially applicable to analysis and understanding. Initial evaluation could match the input, processing, output, and feedback characteristics of various technologies against the corresponding requirements of the prime analysis tasks mentioned above. The results would provide a research agenda for the application of the technologies to the analysis tasks, along with empirical testing of their effectiveness.

A third research area would utilize artificial intelligence techniques to develop deep explanatory arguments based on basic principles and organizational goals to show why one suggested decision is 'better' than comparable alternatives in a given decision making

environment. Such deep explanations could improve the decision maker's confidence in the DSS, as well as enhance his/her insight into the decision making environment and foster better decisions in the future. It should be noted, however, that this requirement assumes the existence of a knowledge warehouse containing the basic business principles and the organizational goals, as well as an indexing scheme and search algorithms to extract appropriate principles and goals for specific arguments.

A second area in the knowledge warehouse architecture that could benefit from future DSS research is in the validation process of knowledge prior to being fed back into the knowledge warehouse. Such questions that should be addressed include: (1) How much filtering of potential new knowledge should be allowed, (2) Who should be responsible for this filtering (CKO, leaders in GSS/GDSS, etc.), (3) What the filtering criteria should be, and (4) What are the tradeoff of artificial intelligence vs. human intelligence in this filtering process. The answers to these questions could significantly impact the implementation and eventual overall quality of the knowledge warehouse and the decisions it supports.

8. Summary and conclusions

In this paper, we have proposed a

knowledge warehouse (KW) architecture as an extension to the Data Warehouse (DW) model. The KW architecture will not only facilitate the capturing and coding of knowledge but will also enhance the retrieval and sharing of knowledge across the organization. Essentially, the KW will provide the same service for knowledge that a DW provides for data. The primary goal of the KW is to provide the decision maker with an intelligent analysis platform that enhances all phases of knowledge.

The development and implementation of the KW architecture proposed here is a large, multifaceted project, with much work remaining. Specifically, there are three major aspects of associated research. The first addresses the analysis tasks themselves; specifically, (1) defining/refining the analysis tasks that most likely enhance insightful understanding, (2) developing a task-vs.-technology table that matches the various inductive analysis technologies with the appropriate analysis task, and (3) evaluating the results of these technologies when applied to model analysis. The second area of research addresses the empirical testing of the insight generation capability of KW and its analysis tasks in both a controlled and real-world environment. A third area of research addresses the computer-assisted generation of arguments, especially deep explanatory arguments, and empirically testing their ability to

enhance user understanding.

In order to accomplish these goals, the KW should efficiently generate, store, retrieve and, in general, manage explicit knowledge in various forms. Secondly, the KW should be able to store, execute and manage the analysis tasks and its supporting technologies. Finally, the KW should provide computer-assisted support to generate natural language arguments concerning both the comparable validity of the models, meta-models and relations produced by analysis tasks, and how this new knowledge relates to the decision maker's purpose.

The proposed KW architecture consists of an object-oriented knowledge base management system module (OO-KBMS), a knowledge analysis workbench, and a communication manager. The OO-KBMS module integrates a wide variety of knowledge objects and analysis tasks. The knowledge analysis workbench handles the interaction with the analysis tasks, including task control, argument generation, and encapsulation of new analysis algorithms into object models. The communication manager handles all analysis communication between the OO-KBMS and the user interface. The communication manager accomplishes this effectively through the use of five functional sub-modules: a knowledge engineer, what-if interface, query processor, results presentation manager, and on-line help.

The KW will also include a feedback loop to enhance its own knowledge base with the passage of time, as the tested and approved results of knowledge analysis is fed back into the KW as an additional source of knowledge. The primary role of the feedback loop is to provide the capability of both storing the explicit knowledge elicited from the decision maker(s), and also immediately making it available for other users in the system.

Exhibit A

Products that claim to offer integrated solution for the KW architecture are Raven, FrontOffice and Knowledge Warehouse 5.0 from SAP.

- Raven is the code name for the package released in mid 2000 by Lotus Development which has three main components — expertise profiling/locating, a collaborative portal and content tracking and analysis piece (Velker 1999).

- FrontOffice Technologies' flagship product FrontOffice's capabilities include: an enterprise document management based on Microsoft Exchange; integrated searching of enterprise documents, e-mail, intranet, and the internet; and document access from the FrontOffice Workplace, Microsoft Exchange, Windows 95 Explorer, and custom applications.

- SAP's Knowledge Warehouse 5.0

offers five different functionalities — Web check-in, authoring and editing, Document Modeling Workbench, Performance Assessment Workbench, Integration with the Document Management System (DMS), Connection to business workflows.

The following are some examples of vendors and their products available to support the KBMS module of the proposed KW architecture.

(a) Tower Technology (www.towertechnology.com) delivers high volume, production imaging, case management and integrated document management (IDM) solutions. The flagship product Tower IDM is an integrated document management solution that is tightly integrated with Lotus Notes and MS exchange. Tower IDM provides one common enterprise for full function production imaging, case management, COLD/ERM and document management in both client/server and internet browser environments.

(b) IBM software Solution's (www.software.ibm.com/data) KnowledgeX enables companies to make informed decisions by improving the creation, dissemination and use of acquired organizational knowledge. IBM's KnowledgeX facilitates conversion of information into knowledge by revealing hidden relationships from disparate information

sources.

(c) FileNET (www.filenet.com) delivers content management software solution for corporate and gov-ernment organizations. FileNET's Panagon products help customers to better manage their digital content and business processes in order to use information more effectively. FileNET's internet and client/server solution provide standard-based workflows, document imaging, electronic document managing and report ma-nagement (Computer Output to Laser Disk (COLD)) software for managing information and enhancing productivity.

(d) The Unisys (www.unisys.com) Universal Repo-sitory (UREP), a highly scaleable enterprise system, helps integrate different services (such as Asset Man-agement, Component-based Development, Corporate Meta Data Management, and Tool Interoperability) of the enterprise.

The following are some examples of vendors and their products available to support the knowledge analysis workbench module.

(a) The VantagePoint (www.thevantagepoint.com): provides competitive technical intelligence professio-nals and technology managers with new, powerful, and unique capabilities to help extract knowledge from text databases thus enhancing the following

five analysis tasks:

- scanning (identification of new technologies, developments in existing technologies, and new uses of technologies),

- profiling (discovery of the key people and organizations),

- mapping and decomposition (identification of key dependency relationships among technolo-gies (other technologies, scientific phenomena, manufacturing capabilities, etc.)),

- trending (establishing how a technology has

emerged, its applications, and what factors (technical and non-technical) appear to govern its development),

- forecasting (projecting how a technology could

evolve, how it might diffuse into application, and the potential impacts of these events).

(b) VxInsight (<http://www.cs.sandia.gov/fdkjohns/JIIS/Vx—Overview.html>): developed by San- dia National Laboratories, VxInsight provides a visual mechanism for browsing, exploring and retrieving information from a database. The graphical display conveys information about the relationship between objects in several ways and

on multiple scales. In this way, individual objects are always observed within a larger context.

(c) KnowledgeMiner

(<http://www.knowledgemi-ner.net/>): is a new data-mining tool that enables anyone to use its unique form of modeling to quickly visualize new possibilities. It uses principles of Artificial Intelligence and the tool is designed to extract hidden knowledge from data easily. It was built on the cybernetic principles of self-organization: Learning a completely unknown relationship between output and input of any given system in an evolutionary way from a very simple organization to an optimally complex one.

(d) Dataware Technologies' Knowledge Management Suite 3.0 (www.dataware.com) with its text mining capabilities helps users discover hidden relationships between concepts that are buried in large knowledge sources. It accomplishes this by generating a list of related concepts thus increasing the amount of information users can process and at the same time minimizing the possibility of overlooking key information. It provides a single point access to internal and external data sources. It also helps identify and contact co-workers with expertise on specific topics.

(e) Autonomy (www.autonomy.com) develops software that automates large volumes of unstructured content. It is able to automate

these tasks because of the software's ability to analyze a document, extract ideas, and determine which ideas are most important. This is the result of proprietary pattern matching technology. The software can also profile users by analyzing the ideas in the document they read or produce. Autonomy's Portal-in-a-box features the ability to automatically create and maintain easy to navigate portal with well-organized information from hundreds of sources.

The following are some examples of vendors and their products available to support the communication manager module.

(a) 80-20software (www.80-20.com):

The document management extensions for Microsoft exchange delivers ubiquitous, seamless and inexpensive document management to the enterprise. MS exchange 5.0/5.5, office 97, outlook 97, windows 95/NT4.0 and internet explorer should be available to every user in the organization with the power to share information.

(b) Lotus Development's

(www.lotus.com) Domino.Doc has transformed document management from a niche application for small groups of specialists to a broader, flexible infrastructure solution, scalable to every user across the organization. The fact that it is fully customizable enables an organization to manage documents throughout their life cycle, share info across the network via web browsers,

notes other ap-plications. It also leverages the scalability, flexibility and security of the Lotus Domino server and thus functions as a key component of knowledge manage-ment through the enterprise.

(c) Knowledge Track's (www.knowledgetrack. com) corporate portal solution is the Knowledge Center v3.0. Often corporations use departments as pilots for implementation of corporate portals. Al-though a solution may be successful in the depart-mental level, the challenge is to take that solution and spread it in the enterprise. Typical problems are lack of scalability and sluggish performance. The knowl-edge Center offers a central location for employees to unlock and organize corporate information relevant to their job functions and thus help companies to com-pete more effectively. The enterprise can share infor-mation with the entire supply-chain, collaborate around information, and easily view and search for information.

References

1. Ahn, T., and G. Grudnitski. "Conceptual Perspectives on Key Factors in DSS Development: A Systems Approach." *Journal of Management Information Systems*, vol. 2, no. 1, 1985, pp. 18–32.
2. Alavi, M., and E. Joachimsthaler. "Revisiting DSS Implementation Research: A Meta-Analysis of the

Literature and Suggestions for Researchers." *MIS Quarterly*, vol. 16, no. 1, 1992, pp. 95–116.

3. Alter, Steven. "Transforming DSS Jargon into Principles for DSS Success." *Decision Support and Executive Information System*, edited by Peter Gray, Prentice-Hall, 1994, pp. 2–26.
4. Alter, Steven. "What Do You Need to Know to Develop Your Own DSS?" *Decision Support and Executive Information System*, edited by Peter Gray, Prentice-Hall, 1994, pp. 58–65.
5. Barki, Henri, and Jon Hartwick. "Measuring User Participation, User Involvement, and User Attitude." *MIS Quarterly*, vol. 18, no. 1, 1994, pp. 59–82.
6. Barron, Andrew R. "Predicted Square Error: A Criterion for Automatic Model Selection." *Self-Organizing Methods in Modeling: GMDH Type Algorithms*, edited by S. J. Farlow, Marcel Dekker, 1984, pp. 86–104.
7. Berson, Alex, and Stephen Smith. *Data Warehouse, Data Mining, and OLAP*. McGraw-Hill, 1997.
8. Brunswik, Egon. *The Conceptual Foundation of Psychology*. University of Chicago Press, 1952.
9. Califf, Mary Elaine. *Relational Learning*

- Techniques for Natural Language Information Extraction. 1997. University of Texas at Austin, PhD proposal.
10. Choo, Chun Wei. *The Knowing Organization: How Organizations Use Information to Construct Meaning, Create Knowledge, and Make Decisions*. Oxford UP, 1998.
11. Davenport, Thomas H., and Laurence Prusak. *Working Knowledge*. Harvard Business School Press, 1998.
12. Devlin, Barry. *Data Warehouse: From Architecture to Implementation*. Addison-Wesley Longman, 1997.
13. Dhaliwal, Jaswinder S., and Izak Benbasat. "The Use and Effects of Knowledge-Based System Explanation: Theoretical Foundations and a Framework for Empirical Evaluation." *Information Systems Research*, vol. 7, no. 3, 1996, pp. 342–362.
14. Dolk, Daniel R., and Benn Konsynski. "Knowledge Representation for Model Management Systems." *IEEE Transactions on Software Engineering*, vol. 10, no. 6, 1984, pp. 619–628.
15. Fayyad, Usama M., et al., editors. *Advances in Knowledge Discovery and Data Mining*. AAAI Press, 1996.
16. Fourer, Robert. "Modeling Languages versus Matrix Generators." *ACM Transactions on Mathematical Software*, vol. 9, no. 2, 1983, pp. 143–183.
17. Gaskin, James T. *Corporate Politics and the Internet: Connection Without Controversy*. Prentice-Hall, 1997.
18. Gray, Paul, and Hugh J. Watson. *Decision Support in the Data Warehouse*. Prentice-Hall, 1998.
19. Greenberg, Harvey J. *A Computer-Assisted Analysis System for Mathematical Programming Models and Solutions: A User's Guide for ANALYZE*. Kluwer Academic Publishers, 1993.
20. Greenberg, Harvey J. "Enhancements of ANALYZE: A Computer-Assisted Analysis System for Linear Programming." *ACM Transactions on Mathematical Software*, vol. 19, no. 2, 1993, pp. 223–256.
21. Greenberg, Harvey J. "Syntax-Directed Report Writing in Linear Programming Using ANALYZE." *European Journal of Operational Research*, vol. 72, 1994, pp. 300–311.
22. Hammond, Kristian F. "Case-Based Planning." *Proceedings of the Workshop on Case-Based Reasoning*, 10–13 May 1988, pp. 17–20.
23. Harris, Karen. *Chief Knowledge Officer:*

- Managing Intellectual Assets. Gartner Group Document KA-03-8437, 1998.
24. Hogue, John T., and Hugh J. Watson. "Management's Role in the Approval and Administration of DSS." *MIS Quarterly*, vol. 7, no. 2, 1983, pp. 15–26.
 25. Horner, B. R., and Izak Benbasat. "Measuring the Linkage between Business and Information Technology Objectives." *MIS Quarterly*, vol. 20, no. 1, 1996, pp. 55–81.
 26. Inman, W. H. *Building the Data Warehouse*. Wiley, 1996.
 27. Jang, Dong-Hyun, and Sung Hyon Myaeng. "Development of a Document Summarization System for Effective Information Services." *RIAO 97: Computer-Assisted Information Searching on Internet*, 1997, pp. 101–111.
 28. Johnson, L. "The Need for Competence Models in the Design of Expert Systems." *International Journal in Systems Research and Information Science*, vol. 1, 1985, pp. 23–36.
 29. Jones, C. V. "An Introduction to Graph Based Modeling Systems: Part I. Overview." *ORSA Journal on Computing*, vol. 2, no. 2, 1990, pp. 136–151.
 30. Joshi, K. "The Measurement of Fairness or Equity Perceptions of Management Information Systems Users." *MIS Quarterly*, vol. 13, no. 3, 1989, pp. 343–358.
 31. Joshi, K. "A Model of Users' Perspective on Change: The Case of Information Systems Technology Implementation." *MIS Quarterly*, vol. 15, no. 2, 1991, pp. 228–241.
 32. Joshi, K., and T. Lauer. "Impact of Information Technology on Users' Work Environment: A Case of Computer Aided Design (CAD) System Implementation." *Information & Management*, vol. 34, 1998, pp. 349–360.
 33. Kallman, Ernest A., and John P. Grillo. *Ethical Decision Making and Information Technology*. 2nd ed., McGraw-Hill, 1996.
 34. Kennington, John L., and R. V. Helgason. *Algorithms for Network Programming*. Wiley, 1980.
 35. Kidd, Alison L., editor. *Knowledge Acquisition for Expert Systems: A Practical Handbook*. Plenum, 1987.
 36. Kim, C. N., and Raymond McLeod Jr. "Expert, Linear Models and Nonlinear Models of Expert Decision Making in Bankruptcy Prediction: A Lens Model Analysis." *Journal of Management Information Systems*, vol. 16, no. 1, 1999, pp. 189–206.

37. Kimbrough, Steven O., et al. "The Coast Guard's KSS Project." *Interfaces*, vol. 20, no. 6, 1990, pp. 5–16.
38. Kimbrough, Steven O., J. R. Oliver, and C. W. Pritchett. "On Post-Evaluation Analysis: Candle-Lighting and Surrogate Models." *Interfaces*, vol. 23, no. 3, 1993, pp. 17–28.
39. Kohler, Wolfgang. *The Task of Gestalt Psychology*. Princeton UP, 1969.
40. Kolodner, Janet L. "Extended Problem Solver Capabilities through Case-Based Inference." *Proceedings of a Workshop on Case-Based Reasoning*, 10–13 May 1987, pp. 21–30.
41. Kosy, D. W., and B. P. Wise. "Self-Explanatory Financial Planning Models." *Proceedings of the National Conference on Artificial Intelligence*, Aug. 1984, pp. 176–181.
42. Kupiec, Julian, Jan Pedersen, and Francine Chen. "A Trainable Document Summarizer." *SIGIR-95: Proceedings of the 8th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, edited by Edward A. Fox, Peter Ingwersen, and Raya Fidel, ACM Press, 1995, pp. 68–73.
43. Lederer, Albert L., and Vijay Sethi. "Root Causes of Strategic Information Systems Planning Implementation Problems." *Journal of Management Information Systems*, vol. 9, no. 1, 1992, pp. 25–46.
44. Liang, Ting-Peng. "Model Management for Group Decision Support." *MIS Quarterly*, Dec. 1988, pp. 667–680.
45. Meredith, Jack R. "The Implementation of Computer Based Systems." *Journal of Operational Management*, Oct. 1981, pp. 11–21.
46. Newell, Allen, and Herbert A. Simon. *Human Problem Solving*. Prentice-Hall, 1972.
47. Newman, Michael, and Rajiv Sabherwal. "Determinants of Commitment to Information Systems Development: A Longitudinal Investigation." *MIS Quarterly*, vol. 20, Mar. 1996, p. 1.
48. Nonaka, Ikujiro, and Hirotaka Takeuchi. *The Knowledge-Creating Company*. Oxford UP, 1995.
49. Palvia, Prashant, and N. L. Chervany. "An Experimental Investigation of Factors Influencing Predicted Success in DSS Implementation." *Information & Management*, vol. 29, 1995, pp. 43–53.
50. Parsaye, Kamran, et al. "Intelligent Databases." *AI Expert*, Mar. 1990, pp. 38–47.

51. Perkins, David N. Knowledge as Design. Lawrence Erlbaum Associates, 1986.
52. Piela, Paul C., et al. "An Object-Oriented Computer Environment for Modeling and Analysis, Part 1—The Modeling Language." *Computers & Chemical Engineering*, vol. 15, no. 1, 1991, pp. 53–72.
53. Riloff, Ellen. "Automatically Generating Extraction Patterns from Untagged Text." *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, 1996, pp. 1044–1049.
54. Saltelli, Andrea, and T. Homma. "Sensitivity Analysis for Model Output." *Computational Statistics & Data Analysis*, vol. 13, 1992, pp. 73–94.
55. Saltelli, Andrea, and J. Marivoet. "Non-Parametric Statistics in Sensitivity Analysis for Model Output: A Comparison of Selected Techniques." *Reliability Engineering & System Safety*, vol. 28, 1990, pp. 229–253.
56. Sharda, Ramesh, and D. M. Steiger. "Inductive Model Analysis Systems: Enhancing Model Analysis in Decision Support Systems." *Information Systems Research*, vol. 7, no. 3, 1996, pp. 328–341.
57. Skyrme, David J., and Debra M. Amidon. *Creating the Knowledge-Based Business. Business Intelligence*, 1997.
58. Soderland, Stephen, and Wendy Lehnert. "Wrap-Up: A Trainable Discourse Module for Information Extraction." *Journal of Artificial Intelligence Research*, vol. 2, 1995, pp. 131–158.
59. Sprague, Ralph H. "DSS in Context." *Decision Support Systems: Putting Theory into Practice*, edited by Ralph H. Sprague and Hugh J. Watson, 2nd ed., Prentice-Hall, 1989.
60. Sprague, Ralph H., and Eric D. Carlson. *Building Effective Decision Support Systems*. Prentice-Hall, 1982.
61. Sprague, Ralph H., and Hugh J. Watson. *Decision Support for Management*. Prentice-Hall, 1996.
62. Steiger, David M. "Enhancing User Understanding in a Decision Support System: A Theoretical Basis and Framework." *Journal of Management Information Systems*, vol. 15, no. 2, 1998, pp. 199–221.
63. Steiger, David M., and Ramesh Sharda. "LP Modeling Languages for Personal Computers: A Comparison." *Annals of Operations Research*, vol. 43, 1993, pp. 195–216.
64. Steiger, David M., Ramesh Sharda, and B. LeClaire. "Graphical Interfaces for

- Network Modeling: A Model Management System Perspective.” *ORSA Journal on Computing*, vol. 5, no. 3, 1993, pp. 275–291.
65. Stephanopoulos, George, G. Henning, and H. Leone. “MODEL.LA—A Modeling Language for Process Engineering—I. The Formal Framework.” *Computers & Chemical Engineering*, vol. 14, no. 8, 1990, pp. 813–846.
 66. Stephanopoulos, George, G. Henning, and H. Leone. “MODEL.LA—A Modeling Language for Process Engineering: II. Multifaceted Modeling of Processing Systems.” *Computers & Chemical Engineering*, vol. 14, no. 8, 1990, pp. 847–869.
 67. Tait, Peter, and Iris Vessey. “The Effect of User Involvement on System Success.” *MIS Quarterly*, vol. 12, no. 1, 1988, pp. 91–108.
 68. Turban, Efraim, and Jay E. Aronson. *Decision Support Systems and Intelligent Systems*. 6th ed., Prentice-Hall, 2000.
 69. Turney, Peter. *Extraction of Keyphrases from Text: Evaluation of Four Algorithms*. Technical Report, National Research Council of Canada, 1997.
 70. Tuttle, Brad. “Investigating the Impacts of Managerial Turnover/Succession on Software Project Performance.” *Journal of Management Information Systems*, vol. 9, no. 2, 1992, pp. 127–144.
 71. Wagner, Harvey M. “Global Sensitivity Analysis.” *Operations Research*, vol. 43, no. 6, 1995, pp. 948–969.
 72. Wah, Louie. “Behind the Buzz.” *Management Review*, Apr. 1999, pp. 16–26.