A Framework for Understanding Large Scale Digital Storage Systems

J.F. Berry
US Department of Defense
Ft. George G. Meade, MD 20755

Abstract

The digital revolution is now underway. The use of binary zeros and ones to store data is increasing at a steady rate. They may represent text, images, pictures, sounds, maps, books, music, instructions, programs, or just about anything else which can be represented digitally. As the sizes of the digital data holdings have continued to grow, so too has the need to provide meaningful access to this data. There are a number of efforts now underway to provide such access. In most cases the efforts have been domain specific and progress in one area has been hard to replicate in a different domain. Part of this difficulty has been the lack of a general set of concepts and vocabulary that are sufficiently broad enough to bridge the gaps.

This paper presents a general taxonomy of knowledge that is independent of subject matter domain. It begins with knowledge as the most general class and then proceeds to subdivide knowledge into its constituent parts: factual knowledge, procedural knowledge, and judgmental knowledge. Definitions of each type of knowledge are given along with examples sufficient to understand each subclass. A vocabulary is introduced that provides a means to discuss the topic in a manner independent of a specific problem domain. Understanding of the differences between different types or classes of knowledge is necessary if a person or an organization is to begin to build systems that acquire, organize, store, and retrieve various types of knowledge.

The paper concludes with a discussion of some tools that are currently available to assist in the building and maintaining of a knowledge resource.

Introduction

Technology that supports the acquisition and storage of vast quantities of digital data is now being adopted by many organizations. Within the government sector, including NASA/EOS/DIS and various defense agencies, organizations are actively planning for storage systems that can handle multiple petabytes of data [1],[2]. This quantity of data will be obtained from new generations of digital-based sensors. A data explosion also is underway in the health community as more and more imaging systems are added to their inventories. Geophysical organizations are moving toward the use of three-dimensional views of a physical world that once was seen only in two dimensions. Visualization of physical phenomena is already showing real promise and its use is expected to grow rapidly as newer and better tools are developed. Weather models that deal with large numbers of variables have been in use for some time now. In this field the emphasis is shifting toward producing simulations that offer significant speedups over real time so the results can be used to anticipate actual physical events before they occur. Entertainment and filmmaking are two more industries that are accelerating their efforts to move from film to digital data. Digital image creation, editing, and processing are already widely used in the field of special effects where computers are routinely used to create the required scenes. This industry is also rapidly moving toward the use of digital imaging as the production media of choice. All of these trends are expected to accelerate throughout this decade. Improved hardware, software, and development techniques will contribute to this acceleration. A recent issue of Advanced Imaging [3] listed four pages of electronic imaging and imaging products for use in the industrial sector. In this market sector as well, the digital revolution is well underway.

As these trends accelerate, various industries are beginning to show concern. The key questions of how to acquire and store digital data are becoming less important. They are being replaced by questions of how do we assure the effective and efficient use of data as it becomes available, often in near real time. The issues of location, relevance, and usefulness are becoming increasingly important. Within petabyte digital data stores, the relatively simple question of where something is stored has assumed new importance. Techniques that worked for locating items in megabyte or even gigabyte stores are proving that they do not scale.
The binary zeros and ones that are now being stored in vast digital libraries can represent almost anything. They may be text, images, pictures, sounds, maps, books, music, instructions, or programs. In fact, today’s storage systems can contain almost anything that can be represented digitally. One example of an organization that has started to convert its holdings to digital representations is the US Library of Congress [4]. A broad range of users is now requesting access to this data.

Historically, the computing industry has focused on providing hardware technology based on physical constructs such as bytes, pages, tracks, and sectors. These physical components are accessed through logical collections that consist of fields, records, and files. These items are used to provide the means by which the storage and retrieval of data are handled. The IEEE Working Group on Mass Storage uses a bit file construct as its logical view [5]. As recognition has grown concerning the need for higher level views, other working groups have been formed to examine metadata that many define as “data about data” [6]. Thus, our attention has started to focus on what will be needed to move to a higher level of abstraction than that provided by bitfiles.

**Knowledge taxonomy**

A number of people have been working on the development of classification schemes for digital data stores. These schemes provide useful logical abstractions for the zeros and ones that are being stored in our digital data stores [7],[8],[9],[10]. This paper presents a digital data classification scheme that is a bit more general than those in the cited references. The classification scheme is called a knowledge taxonomy. It has been put forth in the hopes that others may find it as useful in their work as I have found it to be in mine. The classification scheme specifically identifies knowledge components that are not found in other models. To the degree that such components are missing, these other models will be limited in their scope and, ultimately, their utility. If the taxonomy presented in this paper is to be widely accepted, it must be meaningful and useful. To be meaningful, it must provide a theory that includes a vocabulary and a framework. Both are needed if one is to understand the subject matter in question—digital data. To be useful, a taxonomy must provide a model general enough to be applicable across the entire range of subject matter that it addresses. It also must be descriptive in a way that is independent of a specific piece of subject matter while still providing a uniform way of uniquely identifying that subject matter. Finally, and most importantly, it must provide a model that can be used as a guideline for building systems that support the areas of interest. Specifically, the model must support the processes of acquiring, handling, storing, using, sharing, and preserving virtually any form of digital data. Early work on the model was begun in 1976 and has continued to the present, [11],[12],[13]. Since that time much progress has been made. This work has been used by the author and others as the basis for two successful implementations that are currently being used in my agency, [14],[1],[15]. Brief descriptions of these implementations are given at the conclusion of this paper. One paper in particular contains great detail on the data modeling effort used in the first implementation [14].

From this point onward in the paper, I refer to large collections of digital data as knowledge resources. My interest has been focused on large collections of knowledge of multiple types as those have been the ones that have been of the most interest to my organization. Searches of the literature will reveal techniques for handling single types of digital data in relatively small quantities within narrow subject domains. The IEEE Mass Storage Model represents a significant improvement over single domain approaches [5]. Its specification provides a means for handling very large quantities of a single type of data known as a bitfile. As such it is applicable to all knowledge domains. However, because the bitfile construct represents a low level of abstraction, it is limiting. Considering the systems that I have helped implement over the years, there is great utility in having a level of abstraction beyond the bitfile.

There are a number of advantages to be gained when one views the digital data problem as a knowledge problem. First, most people I have talked to seem to agree that knowledge, however they define it, represents a very high level of abstraction. The concepts that underlie the word knowledge are sufficiently broad to permit it to serve as the highest level of abstraction of the taxonomy. Specifically, the word knowledge, as it is used in this paper, has three parts: (1) what is known (information), (2) what is known about how to do something (techniques), and (3) the reasons why actions are taken or not taken (justification). Within the knowledge taxonomy, these three areas are called Factual Knowledge (What), Procedural Knowledge (How) and Judgmental Knowledge (Why). Each of the three areas, “What,” “How,” and “Why,” can be subdivided into two parts. Factual knowledge (information) consists of data (facts) and metadata (relationships). Procedural knowledge (techniques) consists of algorithms (rules) and heuristics (plans). Judgmental knowledge (justification) consists of constraints (limits) and goals (targets). The Knowledge Taxonomy is diagramed in Figure 1.

This taxonomy is most relevant to knowledge that is acquired, stored, and used by organizations whose function is to perform knowledge processing. It also is useful for describing knowledge of interest across organizations. Indeed, an ability to share knowledge has been an important characteristic of our systems implementation. The need to
share more knowledge in the future is likely to become more and more important as a global economy emerges.

Let us now proceed to define the items shown in Figure 1. **Factual knowledge** deals with facts or data about those portions of the real world that are of interest to an organization. **Procedural knowledge** deals with the operations or procedures that are applied to other knowledge, including data, to organize it better, to understand it, or to transform it into new knowledge. **Judgmental knowledge** deals with the morals, values and ethics, the rules and regulations, and the goals and objectives to which the people in an organization subscribe and which influence and guide their behaviors. **Judgmental knowledge** is often used to determine which facts or data are relevant, and which procedural knowledge is to be used for knowledge transformation. In many cases a distinction as to which type of knowledge is being used in a given instance is often not clear because:

1. A fair amount of the knowledge in an organization is hidden (being contained mainly within human heads and not available to others in digital forms).
2. There are hierarchies of knowledge contained within each category (factual, procedural, and judgmental).
3. Each type of knowledge may employ other types as input (for example, a statistical routine may need some numerical data as input. Judgmental knowledge is often used to determine which statistical routines yield valid results) and may produce other types of outputs.
4. Judgmental knowledge, since it involves norms, morals, values, ethics, and the interpretation of rules, regulations, and laws, can be somewhat nebulous, especially across occupational areas.
5. People are not used to sorting out and distinguishing various types of knowledge. The example given in (3) above for the selection and use of a statistical routine involved all three types of knowledge.
6. Knowledge has different contexts which tend to make it appear different to different people.

This latter point suggests that a classification of knowledge also must provide for subject matter context or else it will be lacking a vital dimension. This is a point that we missed in our original work. Thus, the expansion of the knowledge taxonomy to include context and subject matter will also be dealt with in this paper. One of the primary values in recognizing knowledge context is the conceptual separation of subject matter according to the needs and purposes of the individuals making use of the knowledge. Examples will be provided to make it clear that an ability to change knowledge context is one of the secrets to the creation of high value knowledge resources. In addition, an understanding of knowledge context is critical if one is to build computerized knowledge resource systems whose intent is to support the acquisition, storage, use, creation, and retention of various types of knowledge. As we will see later in this paper, different types of approaches and tools are needed to handle different types of knowledge. What works for one type of knowledge may be unsatisfactory for another type. In earlier papers, this was referred to as "knowledge independence" [11],[12],[13].

**Factual knowledge**

Factual knowledge comprises the “what” component of an organization’s aggregate knowledge. It deals with specific facts (“data”) and the relationships (“metadata”) between those facts and other objects that individuals in an organization choose to acquire and store. Data, therefore, reflects perceptions of observable reality. Metadata is something quite different. It provides a frame of reference that can be used to interpret the facts. The presence of both components is necessary if factual knowledge is to be widely used by multiple individuals across an organization or across multiple organizations. Without metadata, data is likely to be useful only to the persons who collected the data in the first place as they are the only ones who understand the frame of reference that goes
with the data. Examples will serve to make these points clear.

Data (facts)

"Data" is specifically defined to be recorded digital symbols of all types. As context is added to the knowledge taxonomy, we see that different contexts provide the means to view the same data differently. Within this knowledge taxonomy, the definition of data is intended to include all forms and representations of digital data such as observations, measurements, text, pictures, charts, graphs, spreadsheets, numerical points, digitized sound, forms, and so on. Today, all of this data is available in digital forms. Within many communities, the bulk of digital storage is used to hold collections of stored values or observations. Let us take a very simple example to illustrate this point. Consider a data bank that contains instances of line voltages that are associated with various pieces of electrical equipment. The recording of voltages is likely to be of interest to a company that produces a wide variety of electrical equipment. The stored values or instances of specific line voltage are the data of interest to the example company. However, the data values, of themselves, are not very useful because they generally are meaningful only to the persons responsible for causing the data to be stored in the first place. Access to a data bank that consists of large numbers of entries of the numbers "120," "240," "6," "12," "10," "3.5," "9.3," and so on, may not be of much use apart from the organization that caused this data to be created in the first place. (The numbers selected for this example are likely to mean something to most of the readers of this paper as they represent familiar voltage instances to most of us. As such, the context needed to make the voltage numbers meaningful is available within our own knowledge stores.)

Storing more and more data values whose meaning is known only to a selected group of people is an unsatisfactory situation for a number of reasons. Increasingly, it is recognized that there are almost always links between stored data values and something that has been observed or that is predicted to happen in the real world. Merely recording the data value "120," for example, is of little value unless it is related to something that falls within someone's perception of the real world. The database community, among others, has developed the concepts, entity and attribute, to handle these cases. Raw values such as the number "120" are very ambiguous until they are associated with another object that is often called an attribute. In the case being discussed, the attribute of "volts" provides a context for the data value "120." An attribute is defined to be some property common to all members of an object in a specific class. Taken together, all of the objects within a specific class are called entities. An entity is defined to be a person, place, thing, or event that has existed, does exist, or might exist in the portion of the real world that is of interest to an organization. To identify a given entity from others within its class and also from others that are not within its class, it is necessary to include enough attributes to distinguish it from other objects. Entity objects can exist apart from other entity objects. Within ordinary English language usage, the role of attributes is provided by adjectives and the role of entities is provided by nouns. In most cases, common language usage can be used to distinguish between the two types of objects, entities and attributes.

The pairing of the data, "volts/120," is called an attribute/value pair. Such pairing is a first step toward reducing the ambiguity of values found in a digital data base. Ambiguity still exists however. The pairing does not indicate which entity objects in the real world can have the attribute (property) of "volts" or which can have the value of "120." However, the creation of an attribute/value pair of "volts/120" has already served to limit the context of the value "120" to the measurement of voltage. Attribute/value pairs are always associated with entity objects and cannot exist in their own right. In fact, they can be associated with any entity object that has the attribute of voltage. In common English usage this is known as "noun/adjitive" pairing.

The concepts of entity, attribute, and value provide the basic building blocks for the data piece of factual knowledge. They provide the minimum essential ingredients that are needed for this portion of the taxonomy. Happily, the technology needed to support the management of these items is mature and readily available. Commercial database management systems have been providing such capabilities for years now. However, facts comprise only one piece of factual knowledge. The other piece of factual knowledge is metadata. Metadata is defined as the representation of relationships between entities found in the factual data. In this knowledge taxonomy, metadata is added to the entities, attributes, and values for the purpose of capturing additional knowledge in a manner that can be shared across multiple users.

Metadata (relationships)

Let us now explore the concepts that underlie the term, relationship. Relationship is most similar to the "verb" construct in everyday English. In fact, a relationship between entities is often described by the use of a verb. Relationship also can and really should have attributes that further refine and describe a specific relationship. These relationship attributes have much in common with the "adverbs" found in English and relationship attributes are consistent with English adverbs. The use of verb concepts to understand relationship provides an easy way to sort out whether
something is a relationship or an entity. Relationships represent the verb part of the activity and the entity is the object that participates in the action. For example, consider two entities, "computer" and "room." They can be connected together with a relationship called "located in." The entity "computer" can be related to another entity called "manufacturer" by a relationship called "manufactured by." The "computer" entity can be related to another entity "user" by the relationship "assigned to." Each of the above relationships can have attributes which help define the "what," "when," "where," and "how" details of the relationship. Relationships can be "static" or "dynamic." "Static" relationships have persistence and can be predefined. "Dynamic" relationships are determined by the presence or absence of some relationship attribute, most often time. Dynamic relationships are often the most interesting and are often the subject of much analysis. "Time" or the temporal dimension of metadata provides a means to form a complex set of relationships that have a past, present, or future connotation. Temporal concepts, especially as used by humans, appear to be very specific, but often are not. For example, a query to examine all of the events that occurred within the last 20 minutes is quite specific. However, discussing with users the reasons for such queries often reveals that they might be interested in the occurrence of an event that occurred at 19 minutes and 59 seconds even though it did not fall within the "20 minute" criteria. The concepts of "close," "near," and "far" are relevant in the temporal dimension.

A second concept that has been useful in understanding metadata as it is defined in this taxonomy is set theory. This theory adds several useful constructs to metadata, including the ability to organize entities into sets. Within factual knowledge, sets can be grouped into supersets and subsets using standard set theory definitions. A single entity can participate in multiple sets at the same time and sets can be recursive. One of the most common examples of this type of set is a "parts explosion" where entities such as "nuts," "bolts," and "wires" often participate in many sets. Sets also can be disjoint with no common attributes between the sets. Disjoint sets often provide the basis for distinct databases in many organizations. Some of the most interesting relationships are those that are found among objects that share common attributes. For example, the entities called "man" and "woman" can form a relationship called "marriage." This relationship can develop into another relationship called "parenting" connecting both of them with the entity "children." One or the other may be connected to an entity called "employer" through a relationship called "working for." The real world where most factual data exists is often complex. This leads to relationships that could easily be termed as "messy." However, it is the ability to define complex relationships that make entity/attribute/value pairs useful outside of their original knowledge context.

An ability to model complexity is a necessity if one is to develop factual knowledge bases that are useful to a broad range of users.

The terms entity, attribute, and relation are now in common use within the relational database community. With the conventional data processing world, the total collection of attributes and values that are associated with a single object (entity) are known as a "record" and each attribute is known as a "field." Collections of "records" are called "files." Unfortunately, the data processing meanings for "records" and "fields" are not restricted to entities and attributes since "records" and "fields" also may consist of arbitrary mixtures of different entities, attributes, and values. "Files" likewise can consist of arbitrary mixtures of records of various types. Allowing such mixtures severely restricts the use of the "record," "field," and "file" concepts. The database community has made a lot of money by providing fixes for these limitations. In later paragraphs, I will comment on the limits placed on factual knowledge by current database technology. Today's relational database systems provide minimal support for the definition of dynamic relationships. Most of them also provide only the most basic support for the definition of complex set types. Finally, all of today's systems do a poor job of handling temporal data.

Procedural knowledge

Procedural knowledge is concerned with the "how" component of an organization's aggregate knowledge. It includes all of their operations, procedures, and problem solving techniques. In effect, procedural knowledge is what all of the people within the organization know how to do.

A continuum has turned out to be the most useful way to represent procedural knowledge. Many people, including Herbert Simon have defined this type of knowledge using a continuum. The extremes of Simon's continuum were labeled "programmed" and "nonprogrammed" [16]. Within this knowledge taxonomy, the terms algorithmic and heuristic are used in the same sense as Simon's "programmed" and "nonprogrammed" knowledge.

Algorithmic procedural knowledge is used for repetitive and routine problems that can be solved by detailed step-by-step procedures. Examples include such things as statistical routines, measurement routines, instructions for using conversion tables, and some scientific formulas. A standard computer program written in a language such as FORTRAN or C is one of the most common forms of algorithmic procedural knowledge available in digital form. Heuristic procedural knowledge, on the other hand, includes the procedures used to solve unstructured problems. Techniques for inferencing, heuristic search strategies, tree-pruning approaches, and reasoning are examples of heuris-
provide examples of heuristic procedural knowledge. Most computing systems that use some form of what has been called "artificial intelligence" provide examples of heuristic procedural knowledge. Generally, it is not too difficult to distinguish between algorithmic and heuristic procedural knowledge, but, as with all continua, items that fall near the middle frequently have characteristics of both classifications, and sharp distinctions tend to become blurred. The two categories of procedural knowledge are explained in greater detail in the following sections. A procedural knowledge continuum is shown in Figure 2.

**Algorithmic procedural knowledge**

Algorithmic procedural knowledge is used whenever problems are stereotyped, routine, and amenable to being solved by the repetitive application of some step-by-step rules or procedures. Use of this knowledge implies a complete and detailed understanding of the problem and its solution. Algorithmic knowledge frequently can be expressed in a computer program where the programmer is able to specify carefully the passing of control from one piece of program code to another piece. Every instruction to the computer must be understood and carefully placed in the proper sequence. Within various problem domains, many people have been very successful in computerizing algorithmic procedural knowledge. A large number of people around the world now spend vast amounts of time in converting such knowledge into complex programs. An example of a federal agency that employs extensive amounts of algorithmic procedural knowledge is the Social Security Administration who maintains social security records and prepares the monthly checks that go to millions of recipients.

**Heuristic procedural knowledge**

Not all problems can be solved algorithmically. Often heuristics or "rules of thumb" are developed based upon experience with a given problem set. These heuristics can be used to determine which algorithms are useful within a given problem set and also are employed to help develop new algorithms to replace the heuristics. Many computer programs fall into this category. For example, a spreadsheet program is algorithmic since the processing of every command is predetermined. However, its use is very much heuristic when it is used to address a problem that is being expressed as a spreadsheet. As the needs within a specific problem domain become known, it becomes possible to build tools such as spreadsheets that can be used in a heuristic fashion to achieve solutions to problems that are very complex.

Heuristic procedural knowledge is used to solve problems or to react to situations when it is not possible to specify a particular algorithmic solution. Heuristic knowledge is generally specific to skill areas such as economics, medicine, mathematics, and so on, but it can also involve general commonsense reasoning as well. Heuristic procedural knowledge often can make use of algorithmic procedural knowledge that was developed in another knowledge domain. For example, mathematical and statistical formulas have utility across a broad range of areas. Specific heuristic procedural knowledge is used to guide solutions or reactions and is frequently expressed in either goal-driven or data-driven terms. A transportation planner seeking to obtain an optimal routing (the goal) for the dispatching of multiple types of cargo across several types of transportation systems can be seen as being "goal-driven." An economist running various statistical tests (algorithmic knowledge) on census data operates with a data-driven set of heuristics. The results of each test might suggest certain paths to pursue and close other paths from further consideration. In this case, the results of the execution of the algorithmic knowledge are used in a heuristic fashion to determine the next step or steps to be taken.

**Inferencing.** One common form of heuristic procedural knowledge is inferencing. There are three types of inferencing that are generally recognized: deductive, inductive, and abductive [17]. Deductive inference refers to the process of reasoning that whatever is true of all instances or members of a class must be true of one instance or member. The premise of the deductive argument is said to provide definite evidence for its conclusion. This point is, at the same time, its strength and its weakness. If a deduction does not have a major premise that is without exception true, then attempts to use this form of logic do not have the expected inevitability.

Inductive inference is a form of reasoning where a body of facts or observations is used to discover rules or generalizations that, more or less, explain the observed phenomena. Stated another way, induction means to generalize from a number of cases that which is true and to infer that the same is true for the whole class. This type of inferencing
is directly related to reasoning by analogy and it has much
in common with statistical probability theory. "Data min-
ing" is another example of the use of inductive inferencing
within a digital data store. A specific example of inductive infe-
rence would be for a doctor to examine the records of
all of his polio patients, to observe that all had high fevers
in the early stages of the disease, and then to generalize a
rule which stated that one of the probable characteristics of
polio is the occurrence of a high fever in the patient. If, un-
fortunately, the doctors were also to observe that all of his or
her patients were homeowners, then inductive inferencing
might also lead him or her to conclude that the condition
of being a homeowner was also characteristic of having
polio. What this is points up is the fact that the premise of
an inductive inference may or may not provide definitive
evidence for the conclusion.

Abductive inference is a kind of reasoning where a
hypothesis is formed which, if true, would explain some
collection of observed facts. For example, an observation
that Jane Smith has spots when taken with a rule that all peo-
ple who have measles also have spots might lead a doctor
to hypothesize, via abductive inference, that perhaps Jane
Smith has measles. The problem with arriving at a definite
conclusion at this point is that fact that other types of dis-
case might also produce spots. Therefore, a necessary step
in this form of inference is to attempt to identify additional
evidence that could be used to support the hypothesis.
A high fever and a coated tongue also might be character-
istics that are often present in people who have measles
(and many other diseases as well). Positive or negative in-
dications of symptoms, if taken in combination through a
process of abductive inference, may lend strong support to
a given hypothesis.

Example of use of heuristic knowledge

Let us take a more detailed example to illustrate a sim-
ple, yet complex form of heuristic procedural knowledge.
The problem is to assemble a jigsaw puzzle that has 1,000
pieces. A brute force algorithmic solution would involve
1,000 factorial trials and the sheer size of trials means
that such an approach is not very practical. Therefore,
most people do not attempt to solve the problem in this
manner. Instead they might begin to analyze the problem
to identify any characteristics that might be exploitable
for solution. The results of such exploration might re-
veal that it is possible to construct stable partial solutions
to the problem (parts of the puzzle that can be assem-
bled and then used later to build bigger parts). In addi-
tion, there seem to be two distinct types of data that are
available during the problem-solving activity; the shape
of the contours and the designs printed on the surfaces.
Therefore, the problem-solver might hypothesize that it
would be possible to place all of the pieces of a par-
ticular color in one or more piles and to place all of
the potential edge pieces in other piles. This observation
and hypotheses are used by the problem-solver to for-
mulate a problem-solving strategy that consists of for-
mulating the problem in a way in which stable subcon-
structions can be formed and used to build more com-
plete constructions. (Simply stated, this means to choose
those pieces of the puzzle that appear to be the easi-
est to put together and do them first.) Another part of
the strategy is to characterize the rules of construction
along two or more dimensions (color and shape) that are
not perfectly correlated but are each sufficient (generally)
to determine the correct solution. As a person proceeds
toward the solution of the puzzle, different facts about
the puzzle become evident at various points in the solu-
tion process. The circumstances may be such that the
data are incomplete, subjective, or erroneous (a situa-
tion that is quite prevalent in many real-life problems).
For example, a dog may have chewed off the corners
of some of the pieces or may have destroyed some of
the images. Thus, at any point in time, a person's data
concerning the problem or its solution may be incom-
plete. Nevertheless, the objective is to continue toward
the solution by using the subconstructions as they are
built. These subconstructions can be used as feedback to
help the puzzle solver understand the whole puzzle be-
fore it is constructed. This feedback helps speed the solu-
tion along. Thus, while it may not be possible to de-
fine a suitable or feasible algorithmic solution to solve
a 1,000 piece puzzle, it is relatively easy to define a
set of heuristics that use feedback to achieve the solu-
tion.

As illustrated in these examples, heuristic procedural
knowledge is extremely flexible and is capable of dealing
with a broad variety of problems and situations. The uses
of computer assisted heuristics are likely to grow over the
next five years.

Judgmental knowledge

Judgmental knowledge includes the “why” component of
knowledge. It deals with the constraints and goals used to
determine which factual and procedural knowledge is re-

tent to a given situation or problem. The exercise of judg-
mental knowledge in both the business and governmental
sector also includes consideration of moral and legal issues
that impact the situation. There appear to be two different
types of judgmental knowledge: (1) Constraints (which in-
clude values, laws, rules, and regulations) and (2) Goals
(which include objectives).
Constraints (rules of behavior)

Constraints include two different kinds of knowledge. The first is related to the values that are imparted by the society as a whole and also by the specific groups of which a person is a member. In particular, specific professions have codes of conduct to which their members are expected to adhere. Values generally consist of the written and unwritten guidelines to moral and ethical conduct expected of a person or of a group of persons. The Federal Employees’ Code of Ethics, the Code of Ethics for the Association for Computing Machinery (ACM), and the Ten Commandments found in the Bible are examples of guidelines for behavior that have been written down. Rules of social behavior or survival instincts are examples of values that are, perhaps, less formalized. Values do not carry the force of law and cannot be used as a legal defense to justify their use. Another form of judgmental knowledge that is very important within the scientific community is its rules for creating instances of what is considered valid knowledge for a specific subject area. This is often referred to as the paradigm for that area. Paradigms are a form of judgmental knowledge quite durable and resistant to change.

Regulations, rules, and laws are a form of judgmental knowledge similar to values in that they are written down as statutes or rules of conduct. They are different in that they carry the force of law and judicial action. The Privacy Act and the Freedom of Information Act are two well known examples of this type of judgmental knowledge. The Civil Service Regulations are examples of government-wide rules meant to guide and control employee conduct within the Federal government.

Goals and objectives

Goals are a form of judgmental knowledge closely related to the reasons for the creation and maintenance of specific knowledge in the first place. They are often expressed in broad terms. For example, one of the statistical agencies of the Department of Agriculture has the goal (mission) of developing and providing economic information to members of Congress, USDA policy makers, other government agencies, state and local officials, foreign government leaders, farmers, farm organizations, marketing firms, and farm supply companies.

Broad goals are often broken down into specific goals intended to further refine the definition of the mission in question. The specific goals of the agency mentioned above include developing economic information on:

- US food and fiber production
- farm and rural adjustments
- agricultural use of natural resources
- consumer interests relating to agriculture
- foreign trade of agriculture products
- USDA international technical assistance

Goals such as those earlier may be further refined at increasing levels of detail to clarify the definition of the mission. If properly done, the goals and subgoals should form a tree that can be traced from top to bottom. Otherwise, it is difficult for anyone engaged in knowledge processing for the mission to understand why he or she is engaging in specific activities.

Objectives relate to the specific actions or tasks to be completed in connection with particular goals. Essentially, objectives should serve as a means to measure the progress toward some goal or set of goals. When used this way, the objectives may be expressed digitally as a project management system.

Some examples of a specific objective designed to satisfy a goal are the following: General goal—Make economic information more accessible to more users. Specific objective—Put “Supply and Demand Estimates” into a system that can be accessed through the internet. This would make them available to a broad range of potential users. General goal—Provide more timely economic information. Specific objective—Change the update cycle of the data contained within “Agricultural Statistics” from a yearly basis to a quarterly basis.

Goals and objectives can be externally or internally generated. Examples of external goals for federal agencies are the legislation passed by the Congress. Laws often define the areas that the legislators expect to be addressed by their legislation and they also may specify performance criteria that are expected to be met. Such external definitions and criteria can be directly translated into internal organization goals and objectives.

Use of judgmental knowledge

The area of judgmental knowledge is the poorest defined and least understood aspect of a knowledge resource. Nevertheless, it often is one of the most important pieces of knowledge and is needed to make sense of the other pieces of knowledge. Judgmental knowledge provides the underlying basis for other forms of knowledge. Its exercise provides the justification for the collection, use, sharing, and retention of other forms of knowledge. Judgmental knowledge (laws, rules, regulations, laws, ethics, code of conduct, and so on) should be used to determine whether the knowledge activities are legal or valid. Judgmental knowledge should be made available to all users of a knowledge resource so that the resource can be used for the purposes intended and not be misused for purposes that are likely to cause damage or harm. Personal information collected on individuals
comes most readily to mind as an example of the type of knowledge that has great potential for use and abuse.

**Knowledge abstraction**

There is one final concept to be introduced to further expand this knowledge taxonomy. It is knowledge abstraction. The processes of knowledge abstraction occupy most of the time and energy of individuals who work with knowledge resources and whose job it is to produce new knowledge. Almost no one is interested in understanding or explaining all of the complexities of the phenomena that exist in the real world. Instead, each of us tends to focus on only those portions that are of interest to us. Most of what is known in science, for example, has been constructed in just this fashion. There was a great deal of knowledge on physical and chemical behavior before there were molecular chemistry theories and they, in turn, preceded atomic theory.

**Abstraction process**

The process of abstraction is an important one to a knowledge resource since it is these abstractions that help to reduce the amount of understanding which humans need to possess to use the knowledge contained within the resource. In the example from the preceding paragraph, it is possible to use molecular chemistry theory without being an expert in atomic theory. The necessary abstractions only need to provide an approximate, simplified characterization of atomic theory that is at a lower level of knowledge than molecular chemistry that is, itself, lower in a knowledge hierarchy than theories of chemical behavior. The characteristics of hierarchies of abstractions, and hierarchies of knowledge itself, are highly useful concepts that should be exploited within knowledge resources. (One should keep in mind that hierarchies are artificial constructs that are merely useful simplifications of real-world complexities that rarely can be completely described by one or more hierarchies.)

**Abstraction validity**

It is important to note that the conditions under which abstractions are valid must be preserved as part of the knowledge about that abstraction. For example, one knowledge abstraction is the minimum caloric requirements that are needed to sustain healthy life conditions. Another piece of knowledge abstraction might be facts on the numbers of people within a given population that are likely to experience allergies to milk products. The first abstraction is used by the Food and Nutrition Service Agency, which is charged with establishing the base levels of foods needed to sustain a food stamp recipient. These base levels are translated into monetary terms to determine how much money must be allocated to achieve these goals. These abstractions (minimum caloric requirements and food costs) are not likely to be valid for that portion of the population that cannot use milk products in their diet. Therefore, other knowledge must be used to establish a diet that can provide the ingredients that normally would be supplied by milk products (which are relatively cheap by comparison).

**Time and the temporal dimension of a knowledge resource**

Time is a concept that applies to the entire taxonomy of knowledge. The contents of a knowledge resource are associated with some point in time, either the past, the present, or the future. Most data base systems that might be used to manage factual data do not deal well with the time dimension. Those that do so use crude approaches that do not capture the full richness of this concept. Intervals are an important time construct with many variations possible, such as second, minute, hour, day, month, year, and so on. These dimensions may aggregate in one dimension but not in another. Also, the combination of facts relevant to one dimension of time may not be appropriate to another dimension of time. For example, producing yearly data from accumulations of hourly observations may not yield the same results as producing yearly data from monthly observations even if the same measurement processes were used. Much more attention needs to be paid to time aside from considering it to be simply an attribute of some entity or relationship. In fact, time may be one of the most important contexts of all.

**Time dependencies**

Time dependencies apply to procedural knowledge as much as they apply to factual knowledge. Algorithmic knowledge can change over time. For example when the federal government converted its workers from the Civil Service Retirement System (CSRS) to the Federal Employees Retirement System (FERS), all of the formulas used to calculate benefits due under CSRS did not apply to persons who are under the FERS system. This points out that it often is critical to keep track of the time associated with procedural knowledge and with the factual knowledge that is used with that procedural knowledge to keep them synchronized.

*Judgmental* knowledge requires an explicit time context as well. Laws, rules, and regulations are subject to change with the passage of time. What is permitted or legal today may not be acceptable tomorrow or it may not have been acceptable in the past. Concepts of ethics, values, and morals also change over time. Paradigms that govern the accepted way of conducting research also are subject
to change over time. Hindsight analysis of past decisions, actions, or events requires an understanding of the environment at the time the decision was made. The controversy over the "Enola Gay" exhibit is a case in point. Future analysis requires the extrapolation of several alternative future environments as the future cannot be predicted with absolute certainty. Reactive decisions require understandings of the situation at this instant in time and hence require access to up-to-date knowledge.

Knowledge context

To make the knowledge taxonomy relevant to a specific situation or organization, it is necessary to consider the environment of that situation or organization. By environment I mean the context or subject matter of the knowledge. There appear to be three general contexts to knowledge. They are: mission, tools and support, and direction. Each of these three contexts can be further divided into the knowledge skills needed and the area where the skills are to be applied. This is shown in Figure 3.

A great deal more can be said about knowledge contexts and the reader is encouraged to refer to the author’s previous works on this subject [11],[12],[13]. These documents contain a substantial discussion of this area. It is worth noting that the same piece of knowledge can apply to multiple mission, tools and support, and direction contexts. This means a knowledge resource must be prepared to support multiple contexts if it is to be successful. Few systems do this today and fewer still acknowledge the need to provide such support. For example, it is not clear how valuable a statistical routine coded in FORTRAN would be to an economist who was not able to program. This points up the fact that knowledge must often be transformed from one form (for example, a formula in a reference book) into another form (a computer program) if it is to be used with digitally based knowledge resources. This is true even if the reference book is available in digital form in a knowledge resource. Mere availability is no guarantee of usefulness. Knowledge must also be available in a form that is suitable for its use.

Technology for developing a knowledge resource

Over the years much technology has been developed that can be of value in developing the various pieces of a knowledge resource. Many of the tools identified in the following discussion are now in use in various organizations. Indeed, the speed with which knowledge resources can be developed is likely to be directly correlated to the quality and quantity of tools that become available. Furthermore, it is likely that many tools will be domain specific.

Factual knowledge tools

It is now possible to buy commercial off-the-shelf database management systems that offer substantial capabilities for managing factual knowledge. These systems are a great improvement over the concepts of files, records, and fields, as they provide greater levels of abstraction that are not oriented toward the storage domain. In particular, relational database management systems have proven useful for handling certain types of metadata.

Relational database management systems. In the mid-1980s my own organization first started to use a relational database management system to organize and store our metadata. We took a problem that had been automated using a file, record, and field approach during the 1970s. This approach had proved to be difficult to implement and very costly to modify and maintain. The first step undertaken in the project was to do extensive data modeling of the problem to understand it better. What emerged was a complex set of entities, attributes, and relationships that described the problem and its data. The factual knowledge of the problem was implemented using one of the first relational database management systems available at that time. This implementation has since been ported to a newer relational database system and is is in use. The data modeling effort that we used to define the factual knowledge was described in an unclassified paper published in the Database/87 conference proceedings [14]. This paper contains a good overview of data modeling and can serve as an introductory reference to this topic. The paper contains sufficient detail for others to replicate the approach. Almost all of the issues related to the development of a factual knowledge resource are explored in the paper. It was during this implementation that we demonstrated the value of the relational data model in automating factual knowledge. We also experienced limitations of the relational data model. We found out, for example, that existing relational database management systems did not contain the semantics needed to create and maintain complex data models. An example of a common complex data model is the "parts explosion" that can contain parts, subassemblies, and assemblies. This situation is common in the real world. Current systems have not overcome this limitation. Relational databases also handle only static relationships, not dynamic relationships that can be created on the fly as a result of knowledge processing. However, some of the vendors have introduced the concept of a "temporary table" to provide a limited means for handling such cases. In particular, current relational database management systems do not do a very good job of handling the temporal dimensions of our problems. SQL is especially deficient for expressing the temporal queries.
Current use of relational database management systems. Using the experiences gained in our first implementation, we decided to continue to use relational databases to automate the storage, management, and retrieval of a much broader set of factual knowledge. The second major implementation has been referenced and described in several papers [1],[15]. While some of the mass storage system development mentioned in these papers was not very successful, the metadata portion of the activity was delivered early in the project. It represents a significant success story. It currently provides the technology used to find and retrieve much of the data stored in my organization's data repositories. There are over a hundred entity types and as many relationships. Our users can query this metadata database using entity and relationship attributes and quickly find the retrieval "handles" or "path names" which point to the raw data files that contain the data of interest. There are currently a number of Sun-based database servers dedicated to this function. The factual knowledge management being done in this system is far more sophisticated than that offered by the hierarchical storage managers currently being sold in the market place. Most of those systems do not provide a means to deal explicitly with metadata.

Object oriented database management systems. Object-oriented database technology shows promise in the handling of factual knowledge. Since my organization has no specific experience in implementing this type of system for knowledge resources, nothing further will be said about this area.

Procedural knowledge tools. Many tools have evolved for the handling of procedural knowledge. Within limited areas we have been very successful in automating procedural knowledge. What is lacking at this time is an ability to tie this type of knowledge back to our factual knowledge. In addition, it has been difficult to tie various types of procedural knowledge together. Like other organizations we have been slow to appreciate how one type of knowledge needs references to other types of knowledge if both types are to realize their true value.

Programming languages. Multiple programming languages have emerged that permit the conversion of algorithmic procedural knowledge into a digital form that can be shared with others and also be easily used (compared to a printed form of such knowledge). My own organization has devoted substantial resources toward building and maintaining program libraries that contain much of our accumulated algorithmic procedural knowledge. Other portions of our algorithmic knowledge are found in the large numbers of technical reports produced to document the approaches that we use. Where programs are concerned, they require their users to devote a lot of time toward learning the tool (programming language) as opposed to spending that time improving their mission skills. In general the closer a given algorithmic tool is to the mission context of its user, the easier it is to use by a person with the skills in a specific knowledge context.

Heuristic tools. Tools that support heuristic knowledge have also improved. In selected knowledge domains, heuristics are being used on a daily basis to achieve solutions that would have been impossible just a few years ago. Visualization is one of the best examples of the promise of heuristics as a problem-solving approach. Neural networks are another example of heuristic tools that have been successfully used by my organization to solve some difficult problems.
Judgmental knowledge tools

Finally, judgmental knowledge plays an important role in the acquisition, creation, storage, and use of knowledge. To the extent possible, this type of knowledge needs to be explicitly defined so it can become part of the digital knowledge resource. For now, recording this type of knowledge as text may be the best that can be done. Much research still remains to be done to establish how we can best deal with this type of knowledge. Some crude examples already exist. They are found in project management systems where goals and objectives can be specified and a schedule for accomplishing the goals and objectives is defined. In these tools, the temporal dimension plays a key part.

Summary

This paper has attempted to define a general taxonomy of knowledge. This taxonomy is comprehensive enough to serve as a basis for understanding all of the types of knowledge that can be recorded in a large scale digital storage system. Within the limits of this paper it was not possible to fully flesh out all of the concepts. However, this author believes that sufficient detail has been given to permit others to assess whether this approach has utility in their own work. The concepts and thoughts that have been presented here have served as guidelines for my own approach to developing digital storage systems. In particular it was the basis for the selection of a relational database management system to manage the metadata that controls much of the storage now in use within my own agency. An understanding that different types of knowledge require different approaches is essential if we are to build systems in the future that will help us in our quest for knowledge.

References

[1] G. Cotter, "High Performance Computing at the National Security Agency," Frontiers of Supercomputing II, Univ. of California, 1994, pp. 387–392.
[2] V. Peterson, "The High Performance Computing Initiative: A Way to Meet NASA’s Supercomputing Requirements for Aerospace," Frontiers of Supercomputing II, Univ. of California, 1994, pp. 393–398.
[3] Advanced Imaging, PTN Publishing Co., Jan. 1995, pp. 63–66.
[4] "Digital Library of Congress Leading the Superhighway Revolution . . .," Advanced Imaging, PTN Publishing Co., Jan 1995, pp. 10.
[5] "Mass Storage Reference Model," IEEE Open Storage Systems Interconnection Standard P1244, Storage Systems Working Group, IEEE Mass Storage Systems and Technology Committee, 1995.
[6] Workshop on Metadata for Scientific and Technical Data Management, IEEE Computer Society Technical Committee on Mass Storage Systems Specialists Workshop, National Archives II, Washington, DC, May 1994.
[7] D. Nebert, "Announcement of Electronic Technical Work Group to Incorporate Spatial Data Elements and Operators into ANSI 239.50 Standard," Workshop on Metadata for Scientific and Technical Data Management, IEEE Computer Society Technical Committee on Mass Storage Systems Specialists Workshop, National Archives II, Washington, DC, May 1994.
[8] A. Mcnab and T. Karl, "NCDC Metadata—Some General Characteristics," Workshop on Metadata for Scientific and Technical Data Management, IEEE Computer Society Technical Committee on Mass Storage Systems Specialists Workshop, National Archives II, Washington, DC, May 1994.
[9] J.S. Hughes and A.M. Farny, "The Management of Metadata Within the Planetary Data System," Workshop on Metadata for Scientific and Technical Data Management, IEEE Computer Society Technical Committee on Mass Storage Systems Specialists Workshop, National Archives II, Washington, DC, May 1994.
[10] P. Drew and D. McInnis, "A Heterogeneous Geographic Information Architecture for Hong Kong Infrastructure Systems," Workshop on Metadata for Scientific and Technical Data Management, IEEE Computer Society Technical Committee on Mass Storage Systems Specialists Workshop, National Archives II, Washington, DC, May 1994.
[11] J. Berry and C.M. Cook, "Managing Knowledge as a Corporate Resource," NTIS No. AD-A029 897, July 1976.
[12] J. Berry and C.M. Cook, "Viewing Knowledge as a Resource Within Departments of the Federal Government," NTIS No. PB 271 925, Sept. 1977.
[13] J. Berry and C.M. Cook, "Managing Knowledge as a Corporate Resource," Information Management in Public Administration, F.W. Horton and D.A. Marchand, eds., Information Resources Press, Arlington, VA, 1982.
[14] J. Berry and S. Roth, "Use of the Extended Relational Data Model," Proc. of the DATABASE’87 Conf., San Diego, CA, June 1987.
[15] M. Shields, "Mass Storage at NSA," 3rd NASA GSFC Conf. on Mass Storage Systems & Technologies, Oct. 1993, pp. 195–200.
[16] H. Simon, The New Science of Management Decision, Harper & Brothers, New York, 1960.
[17] H.E. Pople, "On Mechanization of Abductive Logic," Proc. 3rd Int’l Conf. Artificial Intelligence, 1973.