Dynamic Classification Ontologies[1]

Jonghyun Kahng and Dennis McLeod
Computer Science Department, University of Southern California, Los Angeles, California

Abstract

A cooperative federated database system (CFDBS) is an information sharing environment in which units of information to be shared are substantially structured, and participants are actively involved in information sharing activities. In this chapter, we focus on the problem of building a common ontology for the purpose of information sharing in the CFDBS context. We introduce the concept and mechanism of the dynamic classificational ontology (DCO), which is a collection of concepts and inter-relationships to describe and classify information units exported by participating information providers; a DCO contains top-level knowledge about exported information units, along with knowledge for classification. By contrast with fixed hierarchical classifications, the DCO builds domain-specific, dynamically changing classification schemes. Information providers contribute to the DCO when information units are exported, and the current knowledge in the DCO is in turn utilized to assist information sharing activities. We will show that, at the cost of information providers' cooperative efforts, this approach supports effective information sharing in the CFDBS environment.

5.2.1 Introduction

With the rapid growth of computer communication networks over the last decade, a vast amount of information of diverse structure and modality has become available on the networks. We consider this environment from the viewpoint of a *Cooperative Federated Database System* (CFDBS; see Heimbigner and McLeod, 1985); here units of information to be shared are substantially structured and participants are actively involved in information sharing activities.

A CFDBS consists of a number of autonomous *Information Frameworks* (IFs) and *Information Repositories* (IRs), as well as one or more *Dynamic Classificational Ontologies* (DCOs) (Fig. 1). IRs are major sources of information, while IFs are principally portals to IRs and other IFs (because an IR is a special kind of IF, the IF in the following will stand for both IFs and IRs unless otherwise mentioned). A DCO is a common ontology (a collection of concepts and their relationships to describe information units) which serves the basis of mutual "understanding" among participating IFs. Information is shared via mediators provided to support *import* (folding remote information into local environments), *export* (registering information to share), *discovery* (searching for relevant information), and *browsing* (navigating through information sources). Participants in the CFDBS communicate through an agreed-upon common data model and language.

Information sharing in the CFDBS faces a number of challenging problems due to the large volume of information and the rich structure of information units. Our approach is to dynamically build a common ontology, which is used by participants to describe and interpret

[1] This research has been funded in part by NIMH under grant no. 5P01MD/DA52194–02, and in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center with additional support from the Annenberg Center for Communication at the University of Southern California and the California Trade and Commerce Agency.

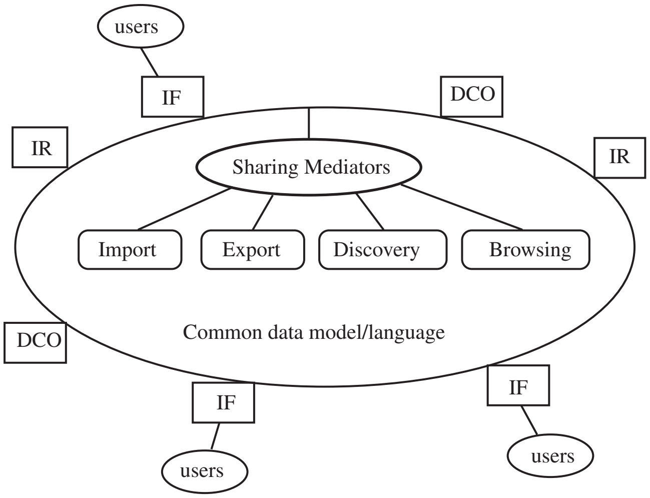


Figure 1 Top level view of a cooperative federated database system (CFDBS).

information that they share. An emphasis of our approach is based on the observation that it is extremely difficult to reach a total agreement on an ontology if the number of participants is large, and that the ontology should be allowed to change dynamically as the CFDBS evolves.

In this chapter, we present the concept and mechanism of the DCO, which addresses problems of the common ontology, and we illustrate how the DCO facilitates information-sharing activities. In order to reduce the size of the DCO, it contains a small amount of high-level meta-knowledge on exported information. Specifically, it contains a collection of concepts and their relationships that is to be used for classification of exported information. We rely on classification because it is an effective scheme for organizing a large amount of information. Further, some relationships in the DCO are not pre-determined but are computed based on exported information. Such relationships typically involve inter-related concepts which are useful for describing or classifying other concepts (for example, relationships among a set of subjects). An advantage of this approach is that participants are not required to agree on those relationships in advance. Another advantage is that those relationships are allowed to change as the usage of involved concepts changes.

The remainder of this chapter is organized as follows. Section 5.2.2 describes the spectrum of heterogeneity present in the CFDBS environment. Section 5.2.3 examines issues associated with the common ontology for resolution of semantic heterogeneity, and reviews related research. Section 5.2.4 discusses the role of classification as an information organization scheme and discusses the representation of classification. Section 5.2.5 describes the DCO in detail, while Section 5.2.6 shows how the knowledge in the DCO is utilized in the mediation of information-sharing activities, in particular, export and discovery. Section 5.2.7 concludes this chapter.

5.2.2 Heterogeneity

A key aspect of the CFDBS is heterogeneity of information at two levels of abstraction:

1. *Data model heterogeneity:* Information systems may use different collections of structures, constraints, and operations (i.e., different data models) to describe and manipulate data. For example, an information system may use a DBMS that supports object-based data modeling and an OSQL; another may store data in a collection of HTML documents and access them through http; and yet another may use a UNIX file system with various file management tools.
2. *Semantic heterogeneity:* Information systems may agree on a data model, but they may have independent specifications of data. This exhibits a wide spectrum of heterogeneity because most data models offer many different ways to describe the same or similar information.

Data Model Heterogeneity

Apparently, there are two alternatives to resolve data model heterogeneity. The first is to translate between every distinct pair of data models; the second is to adopt a common data model and translate between each data model and the common one. In the CFDBS, in which the number of distinct data models is expected to be large, the second alternative is more cost effective and scalable. This follows from a simple calculation of the number of necessary translations: $O(n^2)$ vs. $O(n)$, where n is the number of distinct data models. The price to pay for this alternative is that participating information systems should agree on a common data model; however, the cost of adopting a common data model can be well justified by its benefits. In fact, nearly all proposed systems for database interoperation assume some common data model (Arens *et al.*, 1993; Garcia-Molina *et al.*, 1995; Hammer and McLeod, 1993; Levy *et al.*, 1996; Mena *et al.*, 1996; Sciore *et al.*, 1992).

There is a tradeoff in choosing a common data model. Simple data models reduce the degree of potential semantic heterogeneity and the maintenance cost, but they limit the capability of information sharing; the opposite applies to semantically rich data models. A good example can be drawn from the recently exploding World Wide Web (WWW). Although it provides a great opportunity for people around the world to initiate information sharing, it comes with some intrinsic drawbacks. First of all, its data model is too simple to effectively describe diverse information. The simplicity, of course, represents both sides of a coin. It is the simplicity in part that has made the WWW so rapidly accepted in the Internet community. The simplicity would be acceptable as long as the information to be shared remains simple. This, however, is not the case, because people are now becoming more and more ambitious about sharing diverse information, both structured and unstructured, using WWW.

In the CFDBS, it is essential to adopt a data model that is more expressive than simple hypertext or flat files/tables, for example, because the CFDBS is intended to share information with diverse structures. An advantage of semantically rich models is that no information is lost when translation is done from less expressive models. It is also essential to have operations (query and manipulation languages) that are expressive enough to meet various demands and yet primitive enough to understand easily. An object-based data model might be a good choice because it is easy to understand (compared to richer models such as those of the KL-ONE family, which are more popular in the AI community), it allows effective data modeling in various application domains, and translation from other popular models is reasonably feasible. However, the query language for object-based data models needs improvement. Currently, variations of OSQL are the most prevalent languages for object-based models, but they fail to take advantage of object-oriented concepts such as inheritance. This is mainly because OSQL has its origin in SQL for relational models. Another drawback of OSQL is that users need quite a bit of training before effectively using it. A language that allows users to navigate through databases comfortably without formulating complicated queries would make information sharing in the CFDBS much more effective.

As object-based models, relational models, and some extensions of them have been widely adopted as a common data model in federated database environments, taxonomies of semantic heterogeneity allowed in such models have been extensively studied for the last decade (Kent, 1989; Kim *et al.*, 1993; McLeod, 1992; Sheth and Kashyap, 1992). For the purpose of illustration, two university databases described in an object-based data model are shown in Fig. 2. We summarize incompatibilities between objects resulting from the semantic heterogeneity:

1. *Category:* Two objects from different information sources are under compatible categories if they represent the same or similar real-world entities. Specifically, they may have equivalence, subconcept/superconcept, or partially overlapping relationships. For example, Employees in A and People in B are equivalent because they both represent employees of the universities; Persons in A is a superconcept of People in B because the former represents a more general category of human beings than the latter; Students in A and People in B may be partially overlapping because some students may be employees, as well. On the other hand, Courses in A and People in B are under incompatible categories.

2. *Structure:* Two objects of a compatible category may have different structures. For example, Employees in A and People in B have quite different structures: People has an attribute "birthday," but Employees does not; the attribute "phone-nos" of Employees is equivalent to a combination of the attributes "work-phone" and "home-phone" of People. Another common example of structural incompatibility is that an attribute in one database is an object in another.

3. *Unit:* Two objects under a compatible category with a compatible structure may use different units. Salary in A and B gives an example of this incompatibility, given that the former is measured in dollars whereas the latter is in francs. Quality grades are another frequently encountered example. A grade may be measured on the scale of A, B, C, etc., or it may be measured on the scale of 1 to 10, for instance.

Other incompatibilities orthogonal to the above ones include:

4. *Terminology:* Synonyms and homonyms cause terminological incompatibilities. The attributes "SSN" of Employees and "ID" of People are an example of synonyms.

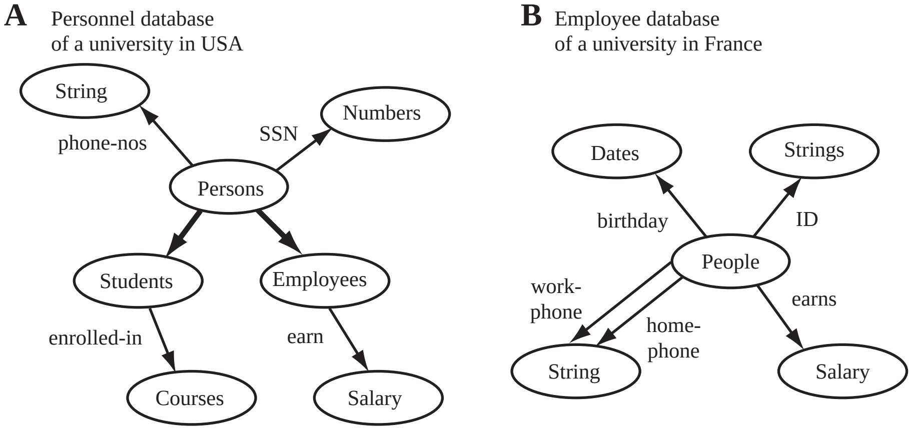


Figure 2 Examples of semantic heterogeneity.

5. *Universe of discourse:* The semantics of data are often hidden in the context. For example, the currencies used in A and B are presumably dollar and franc, respectively, considering their locations.

Resolution of semantic heterogeneity is at the center of interoperation in the CFDBS. Because of the difficulty of the problem, however, decades of research have been able to provide only primitive solutions to the problem, and there is little consensus on how to get beyond them. Among the first three incompatibility problems, we focus on the first one because locating relevant (i.e., categorically compatible) objects alone, setting aside their structural and unitary compatibilities, is a challenge in the CFDBS environment and because resolution of the first should precede that of the others.

A common approach to semantic heterogeneity resolution is to adopt a common ontology as a basis for mutual understanding. This introduces another level of agreement among participants in addition to an agreed-upon common data model. The remainder of this chapter is focused on this approach.

5.2.3 Common Ontology

An ontology is a collection of concepts and interconnections to describe information units. In particular, the common ontology in the CFDBS is to describe information exported from information sources. Fig. 3 shows a generic architecture for information sharing in the CFDBS. Information to be exported is first extracted from information sources and then translated from local data models to a common data model. Semantic heterogeneity among the exported information is resolved by mapping it into a common ontology. In other words, the common ontology is used to describe the exported information. Information sharing is facilitated by mediators provided for export, import, discovery, etc. There are several issues in working with a common ontology:

1. *Contents:* A common ontology could be as simple as a collection of concepts whose relationships are unspecified, or as complicated as a complete collection of concepts and their relationships that is enough to unambiguously describe all the exported information (like an integrated schema, for instance). Since neither of these two extremes are practical, most proposed systems adopt a common ontology that lies between the two. The contents of the common ontology strongly depends on the kinds of semantic heterogeneity that are to be resolved.
2. *Mapping:* Exported information needs to be mapped to (or described by) the common ontology. This process is typically the most labor intensive and time consuming one and is primarily carried out by domain experts. Thus, semi-automatic tools to assist this process would be very useful.
3. *Relevance:* Similarities and differences between two information units from different information sources or relevance of exported information to a given request must be determined at some point during the information-sharing activities.
4. *Maintenance:* Building a common ontology in the first place before any information sharing occurs is a challenging problem. Further, it is very helpful to allow evolution of the common ontology.

The problem of the common ontology has been addressed either implicitly or explicitly in several different contexts, including database interoperation, information retrieval, and Internet resource discovery.

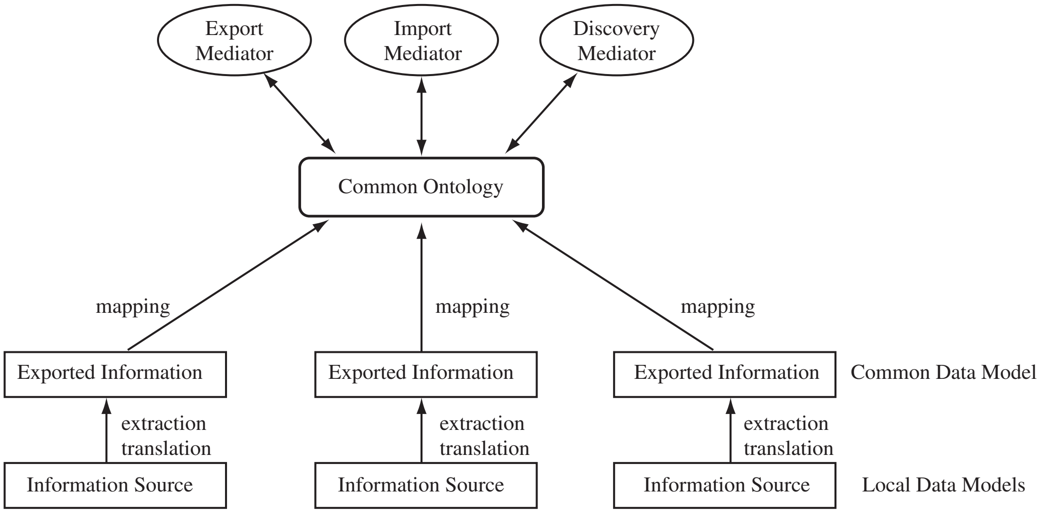


Figure 3 Information sharing in the CFDBS.

Database Interoperation

Early studies in database interoperation paid attention to tightly coupled federated database systems in which the common ontology is an integrated database schema (Batini *et al.*, 1986; Sheth and Larson, 1990). The focus of these systems is to build a database schema that supersedes all component database schemas and to define mappings between the integrated schema and component schemas. A number of techniques have been proposed for this purpose. Extensions of existing relational or object-based data models to improve capability of removing the ambiguity resulting from semantic mismatches among information units from different information sources have also been proposed. Some AI-oriented systems (Arens *et al.*, 1993; Levy *et al.*, 1996) use richer data models (those of the KL-ONE family) and focus on efficient query processing. This tightly coupled approach is not suitable for large systems such as the CFDBS. First, it is very difficult to construct an integrated schema if there are more than a few information sources. Second, a complete resolution needs to take care of detailed semantic conflicts, which could very well result in undesired complications. Third, evolution of the system is made difficult because every change in individual information sources must be reflected into the integrated schema.

Because of these difficulties, a more practical approach for the CFDBS is loosely coupled federated systems (Heimbigner and McLeod, 1985; Sheth and Larson, 1990; Wiederhold, 1992), in which the common ontology provides partial information about participating information sources. The choice of the common ontology in this approach strongly influences the functionality and capability of the system. Examples of proposed common ontologies include: a set of meta-attributes (Sciore *et al.*, 1992), a network of terms (Fankhauser and Neuhold, 1992), concept hierarchies (Yu *et al.*, 1991), summary schema hierarchies (Bright *et al.*, 1994), a set of canonical terms and transformer functions (Mena *et al.*, 1996), and a collection of concepts and relationship descriptors (Hammer and McLeod, 1993). Most of these systems emphasize relevance computation (or query processing) with a given common ontology and mappings; others are concerned with mappings/relationships to a given common ontology. A common drawback of these systems is that they do not deal with the problem of building and evolving the common ontology; the common ontology is defined in advance and more or less fixed.

Information Retrieval

Traditional information retrieval systems are concerned with instance-level (vs. type/class-level) information, as the type of information to be shared is documents with well-known properties such as title, authors, subjects, etc. As in database interoperation, common ontologies play an important role in these systems. In particular, the focus is on measuring the relevance of two documents or relevance of documents to a given request.

The simplest approach is to rely on keyword matching. That is, keywords are extracted from each document either manually or automatically, and two documents are compared based on the extracted keywords (Salton, 1989). The common ontology in this case is implicitly all words in a natural language with relationships among words nearly ignored. This can be improved by introducing synonyms or by replacing extracted keywords with their stems, but it is still too primitive to be useful in a more cooperative environments such as the CFDBS.

Another common approach is to take a collection of pre-classified subjects (a common ontology; see Samme and Ralston, 1982) and to assign a few of them to each document. While the pre-defined classification does include relationships between subjects, it has several undesirable features. First, it is hierarchical for the most part. That is, it contains only subsumption relations between subjects. Although cross-references between related subjects are often a part of the classification, they are not enough to represent overlapping relationships among subjects. Second, it tends to be static. Revision of the classification requires much time and effort. Consequently, it fails to accommodate dynamically changing usage of subjects. Third, it is typically huge and difficult to understand because it usually covers all disciplines and because it contains many artificial terms that are not commonly used in documents. These features make it difficult to apply this approach to the CFDBS environment.

An active area of research in information retrieval is to build term relationships from existing documents. Developed techniques include thesaurus-group generation (Chamis, 1988; Slaton, 1989), concept networks for concept retrieval (vs. keyword retrieval; see Chen and Lynch, 1992), and latent semantic indexing by singular value decomposition (Deerwester *et al.*, 1990). They basically rely on statistical analysis of term occurrence patterns in documents. This research is related to our approach, although our mechanism, assumptions, and context are quite different.

Internet Resource Discovery

It is interesting to observe that Internet resource discovery tools have followed in the footsteps of information retrieval systems. A number of systems based on keywords have been developed and are in use today. As expected, however, searching is not as efficient as desired due to their limitations; the precision of search results is so low that users need to spend much time to sort out retrieved information. To remedy such problems, some recent systems took the approach of classification. Yahoo

(2000), for example, takes a hierarchical classification of subjects and classifies URL objects by those subjects, which is reminiscent of the subject classification used in many library systems. Another one is Harvest (Bowman *et al.*, 1994), in which each broker specializes in a certain category such as technical reports, PC software, etc. It effectively divides the WWW space into several categories, and searching is carried out under each category. This is useful when users know which category of objects is relevant to their interests. In summary, a common ontology plays an important role in information sharing in the CFDBS environment. Many proposed systems adopt a common ontology as the basis for information sharing, but methodologies to construct and evolve the common ontology require more investigation.

5.2.4 Classification

Studies in cognitive science have shown that classification is the most basic scheme that humans use for organizing information and making inferences (Cagne *et al.*, 1993). Categories of objects have features that help identification of the categories, and objects are recognized by associating them with categories. For example, some children distinguish cats from dogs by the feature that cats have whiskers. In principle, all objects in the universe could be placed into a single classification tree; however, that is not the way humans picture the universe. Instead, there are categories at a certain level of generality (basic-level categories) on which people agree the most. When people were asked to list all the features of objects in categories such as trees, fish, birds, chairs, and cars, there was a high level of agreement among people with respect to the common features of those objects. Agreement is less prominent for superordinate categories such as plants, animals, furniture, and vehicles, as well as for subordinate categories such as robin, chicken, sedans, trucks, etc. Further studies showed that basic-level categories are the first ones learned by small children. An implication of these results is that classification is an effective method to organize a large amount of information, and it is natural to classify objects in two steps; the first is to classify objects into basic-level categories, and the second is to further classify objects in individual categories as necessary.

Following these research findings, our approach to constructing a common ontology is based on classification. That is, the common ontology will contain interrelated concepts that are just enough to classify exported information. In particular, the classification is organized around basic-level categories that are specific to the application domain. If independent information systems are in similar application domains, they are likely to agree on basic-level categories, regardless of their underlying data models and physical data structures. For example, most university databases will include information about courses, students, faculty, staff, and libraries at the top level. The agreement on basic-level categories would be very helpful for information sharing in the environment of a large-scale CFDBS.

Classification, in fact, has played an important role in information management. Most of the popular information management systems such as relational/object-based database management systems (DBMSs), the WWW, and hierarchical file systems provide constructs for classification. Some of them facilitate the two-step classification scheme that was mentioned above. To explore the representation of classification in the common ontology, we will examine each data model in turn.

In relational and object-based DBMSs, objects to be modeled are first classified into tables or classes. In the latter, objects in a class can be further classified into subclasses, resulting in class hierarchies. Classification mechanisms directly supported by the systems stop here. But, for a large amount of information, it is useful to classify objects in individual tables or classes. The systems provide an indirect mechanism for that; objects in a table or a class are implicitly classified by their attribute values. Fig. 4a shows a fragment of a library system. In this example, publications are broken down into three subclasses, where all four classes can be regarded as basic-level categories. In addition to the class hierarchy, publications are implicitly classified by subjects; that is, they can be grouped by the same or related subjects. Likewise, journals are classified by affiliated organizations as well as subjects.

Hierarchical file systems are supported in virtually all modern operating systems. A primary use of such file systems has traditionally been the management of personal files, such as documents and programs. Since computer communication networks became highly available, the file systems have been used as the primary storage of information by Internet resource-sharing tools such as WWW, Gopher, and anonymous ftp; they are now an important information organization tool. In hierarchical file systems, a directory can be used as a class of objects, and files in the directory can be regarded as objects in that class. If the number of files in a directory becomes large, they can be broken into subdirectories, resulting in a finer classification of objects. Fig. 4b shows a top-level directory structure of a typical anonymous ftp site. Many anonymous ftp sites have similar directory names and structure in the first one or two levels of directory trees; those directories tend to represent basic-level categories.

The WWW is an interesting invention for various reasons. It is basically a network of URL objects. A main strength of the WWW is that it supports diverse kinds of URL objects including HTML documents, images, video objects, and audio objects. It also provides gateways to Gopher, network news, and anonymous ftp sites. Its capability to organize information, on the other hand, is primitive. It is even more primitive than hierarchical file systems in the sense that it does not

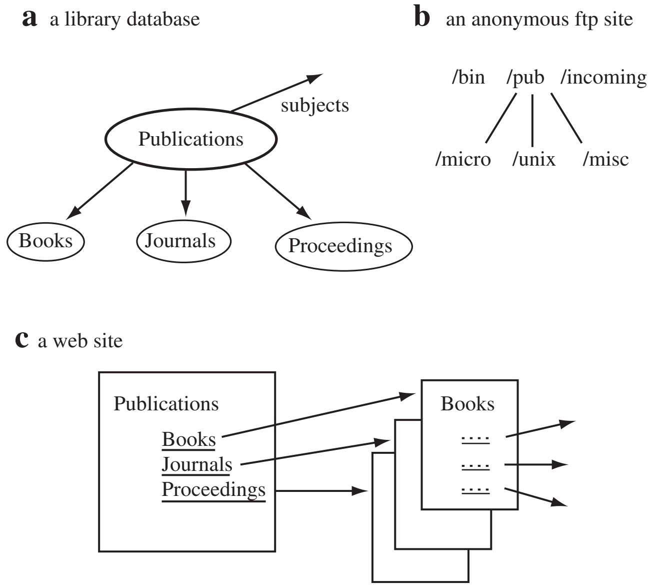


Figure 4 Classifications.

support any second-order modeling primitive that can be used for classification. That is, there is no notion of “type”, “class”, or “directory” as a collection of similar objects. Consequently, classification of objects is totally up to the person who manages information (see Fig. 4c, for example).

From these observations, relational and object-based data models are good candidates for the representation of classification; however, note that tables or class hierarchies with attributes (commonly referred to as a *database schema*) are insufficient to describe the two-step classification. For example, publications are implicitly classified by their subjects (Figure 4a), but that classification would not be very useful if the subjects are not well understood. That is, understanding relationships among subjects would be necessary to make that classification meaningful. The common ontology in our approach addresses this problem.

5.2.5 Dynamic Classificational Ontology

Information sharing in the CFDBS is centered around common ontologies, termed *Dynamic Classificational Ontologies* (DCOs). The CFDBS environment is characterized by a large amount of information with diverse semantic heterogeneity. Resolving semantic heterogeneity and setting up an interoperative environment are therefore extremely difficult and typically costly tasks. To assist in these tasks, the DCO keeps a small amount of meta-information on concepts (information units) exported from IFs; a DCO maintains a common ontology to describe and classify exported concepts.

If the number of participating IFs is more than a handful, it is very difficult to draw a total agreement on a common ontology. Among other difficulties, it is common in the CFDBS environment that certain concepts, such as “interoperability”, are loosely defined but frequently used. Moreover, usage of concepts will keep changing as the CFDBS evolves. It would therefore be impractical to make precise definitions of all concepts in advance and enforce them. To address these problems, the DCO dynamically develops a common ontology that accommodates different understanding of concepts and their relationships in different IFs. Further, evolution of the common ontology is based on the input from individual IFs; the common ontology is maintained by their collaboration. This reduces the central coordination and the cost of setting up a cooperative environment.

A DCO consists of a *base ontology* and a *derived ontology*. The base ontology contains an ontology to describe and classify concepts exported by IFs, and the derived ontology contains an additional ontology to help classification of exported concepts in finer grains. The former is typically static and maintained by a DCO administrator; the latter is dynamic and computed based on the base ontology and the population of exported concepts. Fig. 5 shows the flow of information among the DCO and IFs in a CFDBS. As we see here, export is a part of a learning cycle: it adds knowledge to the DCO while being guided by the knowledge in the current DCO.

Classificational Object-Based Data Model

Knowledge in the DCO is represented by the *Classificational Object-based Data Model* (CODM). The basic unit of information in this model is the *concept*; concepts are grouped into *classes*. A class may have one or more *properties*. A collection of classes and their

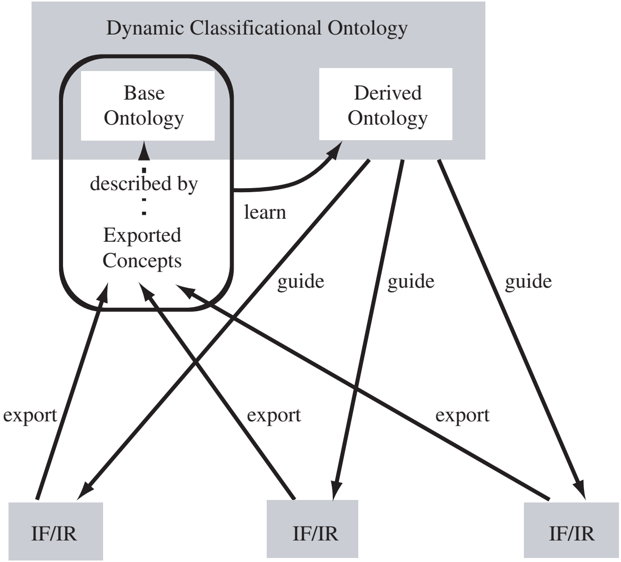


Figure 5 Information flow in a CFDBS.

properties is termed a *schema*. Looking ahead, Fig. 6a shows an example of the schema. The CODM supports generalization/specialization and inheritance of properties from superclasses to subclasses (Cardenas and McLeod, 1990).

In addition, the CODM supports conceptual relationships between concepts, and concept operators, which are useful for dynamic classification. A concept is essentially a representative of a set of real-world entities. Two concepts are *disjoint* if the two sets of entities that they represent are disjoint. One concept is a *superconcept/subconcept* of the other if the set of entities represented by the former is a superset/subset of that represented by the latter; otherwise, two concepts are *overlapping*. A concept operator takes one or more concepts and produces a new concept. There are three concept operators:

1. *Conceptual union* (*OR*): The concept (A *OR* B) represents a set of entities that is the union of the two sets represented by concepts A and B.
2. *Conceptual intersection* (*AND*): The concept (A *AND* B) represents a set of entities that is the intersection of the two sets represented by concepts A and B.
3. *Conceptual negation* (*NOT*): The concept (*NOT* A) represents a set of entities that is the complement of the set represented by A.

A concept is a *composite concept* if it can be decomposed into other concepts and concepts operators; otherwise, it is a *simple concept*.

A property is a mapping from a class to another class (a *value class*); that is, a property assigns to each concept of a given class a *value* which is composed from concepts of the value class. A property is *single valued, multi-valued, or composite valued* if the property value takes a simple concept, a set of simple concepts, or a composite concept, respectively. The first two are common in object-based data models, but the third is unique to the CODM.

The composite-valued property is introduced in the CODM because the multi-valued property generates ambiguities in some cases. To illustrate this point, consider the following examples, where pairs of subjects are given to describe some research articles:

1. {AI, knowledge representation}: If the article is about knowledge representation techniques in AI, it probably means (AI *AND* knowledge representation); that is, it covers the overlapping area of AI and knowledge representation.
2. {object-based data model, relational data-model}: If the article introduces and compares the two models, then (object-based data model *OR* relational data model) might be a better expression.
3. {database, network}: If the article is a survey of database technology, and a part of the article covers network-related materials(database *AND* network) *OR* (database *AND* (*NOT* network)) might well represent its intention, meaning that both network-related and not network-related materials are covered in the article.

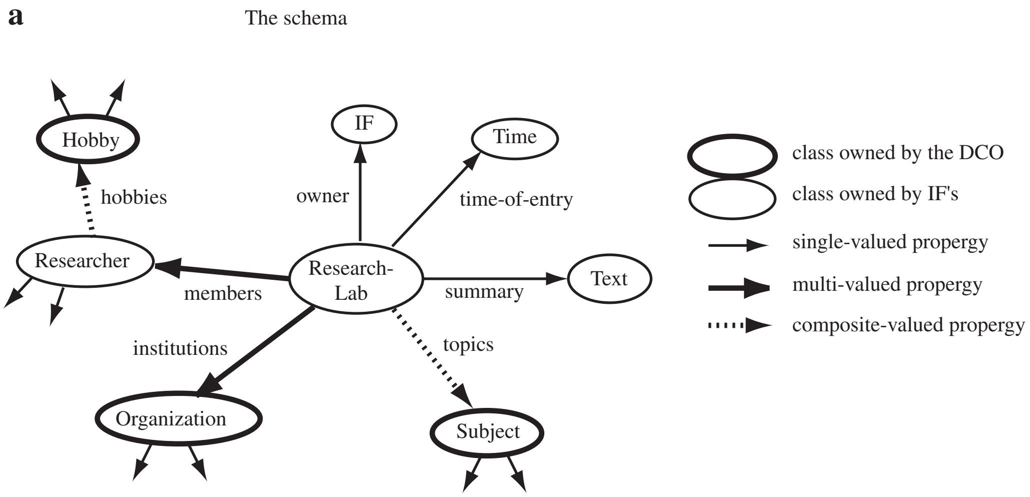
The ambiguities are present because subjects are not independent of each other. In general, a property may be defined as composite valued when concepts of its value class are inter-related with each other.

Base Ontology

A base ontology consists of a schema in the CODM and concepts of selected classes. Fig. 6a shows an example of the schema. Classes in the base ontology represent (basic-level) categories of concepts, and properties represent relationships between such categories. Every concept in the base ontology has two required properties: *owner* is the owner of the concept (see below), and *time-of-entry* is the time when the concept was recorded. Fig. 6b describes a concept DBNETLAB (for convenience, a concept will be identified by a textual string) of the class Research-Lab, where its property names and values are given (e.g., owner: usc-database-if indicates that the value of the property owner is the concept usc-database-if).

The owner of a concept is either the DCO or an IF. Concepts owned by the DCO (e.g., concepts of Organization, Subject, Hobby) are a part of the base ontology. Concepts of other classes (e.g., Research-Lab, Text, Time) are exported by IFs and owned by them. Only the owner of a concept may remove or change it. Concepts owned by the DCO along with the schema are used to describe exported concepts. For example, Fig. 6b shows that the concept of the class Research-Lab was exported by the IF usc-database-if, and the value of its property institutions was chosen from the concepts of the class Organization which are owned by the DCO.

The ownership of concepts depends, in part, on how much information is going to be managed by the DCO.



b A concept of Research-Lab

DBNETLAB
owner: usc-database-if
time-of-entr: 13:22:27/08/23/95
summary: Database Network is an environment where structured information units are shared and exchanged.
topics: (database AND network)
institutions: University of Southern California
members: Aslan, Kahng, Liao, McLeod

Figure 6 A base ontology.

For instance, Organization could be owned by IFs or by the DCO. If the DCO is planned to rigorously follow up information about organizations in any detail, the latter might be a good choice. In this case, the base ontology should include most of known organizations so that exported concepts may refer to them. On the other hand, Organization could be owned by IFs if, for instance, only the names of organizations are to be kept in the DCO.

Another (more critical) reason for the DCO to own and manage some concepts is to assist classification. In Fig. 6, research labs are implicitly classified by their topics (as well as by any other properties). This classification could be ambiguous, however, because subjects are interrelated with each other and the relationships tend to be non-objective. It is therefore necessary to understand conceptual relationships among subjects in order to make the classification useful. To deal with this problem, *topics* is defined as a composite-valued property, and the concepts of Subject are owned by the DCO in our example. Conceptual relationships among subjects are determined by statistical analyses, which will be discussed next.

Derived Ontology

A derived ontology records information about conceptual relationships. Specifically, if concepts of a class are owned by the DCO, and the class is the value class of a composite-valued property, conceptual relationships among those concepts enter the derived ontology. We will first discuss how composite concepts can be interpreted as they are the main source of information, and then describe how to derive conceptual relationships from them.

Interpretations of Composite Concepts

Suppose that simple concepts C1, C2, . . . , CN of a class are owned by the DCO. A composite concept C composed from these concepts can be interpreted in two different ways:

1. *Open interpretation:* C can be interpreted as (C *AND* (CS1 *OR* (*NOT* CS1)) *AND* (CS2 *OR* (*NOT* CS2)) *AND* . . . . *AND* (CSm *OR* (*NOT* CSm))), where CS1, CS2, . . . , CSm are simple concepts that do not appear in C. For example, (database *AND* network) means (database *AND* network *AND* (artificial-intelligence *OR* (*NOT* artificial-intelligence)) *AND* (operating-systems *OR* (*NOT* operating-systems)). . .). In other words, a composite concept may or may not be related to the concepts that are not explicitly mentioned in it.

2. *Closed interpretation:* C can be interpreted as (C *AND* (*NOT* CS1) *AND* (*NOT* CS2) *AND* . . . *AND* (*NOT* CSm)), where CS1, CS2, . . . , CSm are simple concepts that do not appear in C. For example, (database *AND*

network) means (database *AND* network *AND* (*NOT* artificial-intelligence) *AND* (*NOT* operating-systems) . . .). In this case, a composite concept is not related to the concepts that are not explicitly mentioned in it.

A composite concept given by a user might require different interpretations depending on his/her intention. If a user is searching for some information, and the composite concept is provided as a specification of desired information, the open interpretation is probably a better one. That is, the user may not care whether or not the information that he or she wants is also related to other information. On the other hand, if a user is asked to describe some information by a composite concept as precisely as possible, the closed interpretation may be closer to his or her intention. This is because the user would try not to leave out any relevant concepts.

The open interpretation is safer, while the closed one is more informative. If users are not forced to adhere to either of the two interpretations, it is most likely that they will produce composite concepts whose interpretation falls somewhere between the two. The DCO takes the closed interpretation for composite concepts given by exporters. We will later show how the DCO can help them progressively formulate informative composite concepts that are subject to the closed interpretation.

Conceptual Relationships

The population of exported concepts is the basis for the derivation of conceptual relationships. We first define the *frequency* of composite concepts: For a composite concept C of a class Q, and a composite-valued property p whose value class is Q, the *frequency* of C is the number of superconcepts of C among the values of p. For the example in Fig. 6a, if some values of topics are (database *AND* network), (database *OR* network), or ((database *OR* information retrieval) *AND* network), each of them counts toward the frequency of a concept (database *AND* network) as all of them are its superconcepts.

As in mining association rules (Agrawal *et al.*, 1993), we introduce a variable to indicate the significance of statistical data: A *minimal support* is the frequency such that any frequency below it is considered as statistically insignificant.

Thus, if the minimal support is 10 and the frequency of (database *AND* complexity theory) is less than 10, then there is not enough data to determine whether database and complexity-theory are related.

We introduce another variable, the *tolerance factor*, to indicate confidence of derived conceptual relationships. Conceptual relationships are defined with the tolerance factor: Suppose that concepts C1, C2, and (C1 *AND* C2) have frequencies f1, f2, and f3, respectively (see Fig. 7a), and t is the *tolerance factor*. When f3 is larger than the minimal support:

1. C1 and C2 are disjoint concepts within a tolerance factor t, if both f3/f1 and f3/f2 are smaller than t.
2. C1 is a subconcept of C2 within t (C1 < C2) or C2 is a superconcept of C1 within t (C2 > C1), if f3/f1 is larger than (1 - t).
3. C1 and C2 are equivalent concepts within t (C1 = C2), if C1 is a subconcept of C2 within t and vice versa.
4. C1 and C2 are overlapping concepts within t otherwise.

When f3 is smaller than the minimal support, C1 and C2 are considered as disjoint concepts.

Conceptual relationships between C1 and C2 are summarized in the table in Fig. 7a. The tolerance factor represents statistical variations. There are two main causes for such variations; the first is simply that IFs may make mistakes at the time of export, and the second is that different IFs may have somewhat different understandings of involved concepts.

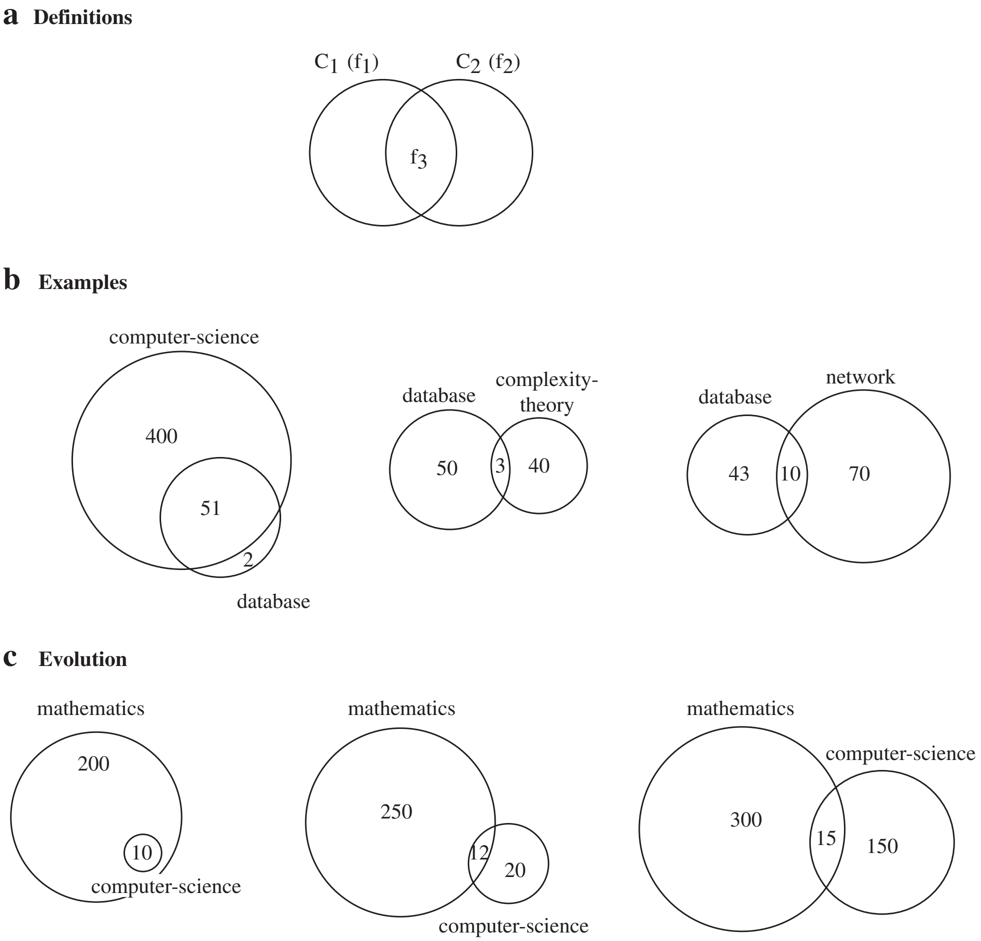
Conceptual relationships among two or more concepts can be best illustrated by diagrams, as in Fig. 7b; this figure shows some relationships among the concepts of Subject. Numbers in the figure indicate frequencies of the concepts; there are currently 400, 51, and 2 research labs whose topics include (computer-science *AND* (*NOT* database)), (computer-science *AND* database), ((*NOT* computer-science) *AND* database), respectively, and so on. Assuming a tolerance factor of 10%, the figure shows that database is a subconcept of computer-science, database and complexity-theory are disjoint concepts, and database and network are overlapping concepts. Fig. 7(c) shows how conceptual relationships may evolve. In the example, computer-science began as a part of mathematics and has grown out of it so that the two are now more or less separate disciplines.

The derived ontology is not fixed but dynamically changes as the population of exported concepts grows. Compared to using fixed conceptual relationships, such as pre-defined hierarchical classifications, this approach has several advantages: the derived ontology can be progressively built up, it may change as usage of concepts changes, and it will shape up in such a way as to reflect domain-specific usage of concepts. It is important to note that the aim of dynamically building the derived ontology is not to derive exact conceptual relationships, but to evolve a collection of reasonably agreeable conceptual relationships.

5.2.6 Mediators for Information Sharing

We will discuss in this section how the knowledge in the DCO is used by mediators provided for information sharing. First of all, the derived ontology depends heavily on information provided by IFs, and it might be unrealistic to expect IFs to provide precise descriptions of information to export from the beginning. We will show how the DCO can help them progressively formulate their descriptions. Discussed next will be discovery:

	$f_3/f_1 < t$	$t < f_3/f_1 < 1\text{-}t$	$f_3/f_1 > 1\text{-}t$
$f_3/f_2 < t$	disjoint	overlapping	$C_1 < C_2$
$t < f_3/f_2 < 1\text{-}t$	overlapping	overlapping	$C_1 < C_2$
$f_3/f_2 > 1\text{-}t$	$C_1 > C_2$	$C_1 > C_2$	$C_1 = C_2$



a Definitions

b Examples

c Evolution

Figure 7 Conceptual relationships.

in the presence of abundant information, one of critical problems is to sort out information that is relevant. It is therefore essential to measure the relevance of available information to a given discovery request. We will show how to do that with the help of the DCO.

Export Mediator

An IF exports a concept by submitting an entry using the schema of the base ontology as a template. The entry should include the name of the class to which the concept belongs, along with values for its properties; Fig. 8 shows an example. If the value of a property is concepts owned by the IF, it is accepted as entered (e.g., members: McLeod, McNeill). On the other hand, if it is owned by the DCO, it should be composed from existing concepts of the value class (e.g., institutions: University of Southern California). In particular, if the property is composite valued (e.g., topics, hobbies), the IF is allowed and encouraged to refine the value as precisely as possible with the help of the DCO.

CLASS: Research-Lab
owner: usc-bp-if
time-of-entry: 09:12:35/09/15/95
summary: The lab has been developing neuroscience databases that contain information about related literature and experimental data.
topics: (discovery AND scientific-db)
institutions: University of Southern California
members: McLeod, McNeill

Figure 8 An entry for export.

The export mediator utilizes the knowledge in the DCO to help IFs formulate the description of concepts that they export, especially when the description involves composite-valued properties and concepts owned by the DCO. It applies the following strategies to achieve this goal with minimal interaction with IFs.

STRATEGY 1

For a composite concept (as the value of a composite-valued property) given by the IF, concepts that are overlapping with it are retrieved from the DCO and presented to the IF so that the composite concept may be modified, restricted, or extended with them.

This is a rather straightforward strategy of utilizing conceptual relationships. Lines 1 through 3 in Fig. 9 show that application, heterogeneous-db, data-model, language, etc. turned out to be overlapping with the given composite concept (discovery *AND* scientific-db), and the IF added three of them to the composite concept.

STRATEGY 2

Among the overlapping concepts found by Strategy 1, only the concepts that are not subconcepts of others are presented to the IF to enable the IF to refine the composite concept from the top level to lower ones in progressive steps. This strategy reduces the number of related concepts to present to the IF so that it is not overwhelmed by a large number of concepts to choose from. Concepts that are left out will be further explored later only if their superconcepts are determined to be relevant by the IF. Lines 3 and 4 in Fig. 9 show that subconcepts of data-model (i.e., object-based-model and relational-model) are presented to the IF in the second round because data-model was selected by the IF in the previous round.

STRATEGY 3

If two concepts A and B given by the IF are disjoint, the IF is asked to choose either (A *AND* B) or (A *OR* B). It is useful to distinguish between (A *AND* B) and (A *OR* B) in order to improve the accuracy of the derived ontology. The list of concepts given by the IF is by default regarded as a conceptual intersection of those concepts. If A and B are disjoint, then (A *AND* B) is a non-existing composite concept (i.e., its frequency is insignificant), and it may not be what the IF intended. Lines 5 through 7 demonstrate this strategy. In this example, object-based-model and relational-model are assumed to be disjoint in the current DCO. If the IF insists that the previously given input is correct, it provides a basis for the composite concept (object-based-model *AND* relational-model) to develop and for the conceptual relationship between object-based-model and relational-model to change in the DCO.

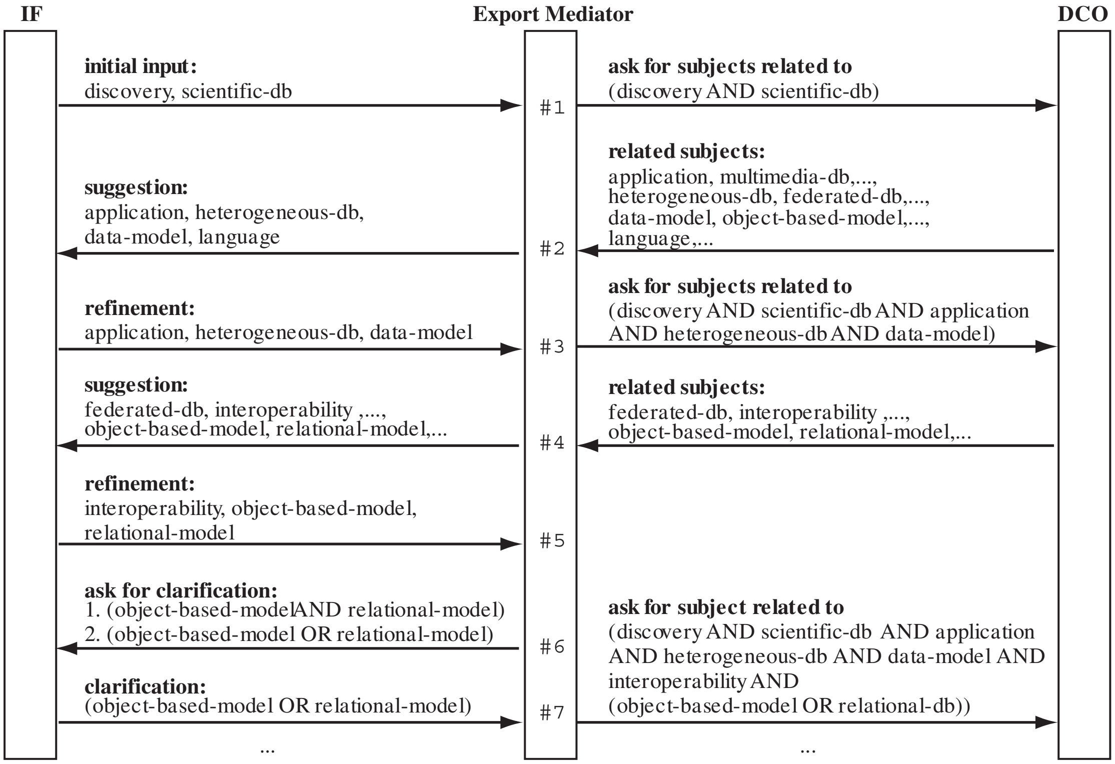

IF
Export Mediator
DCO
initial input:
discovery, scientific-db
#1
ask for subjects related to
(discovery AND scientific-db)
related subjects:
application, multimedia-db,...,
heterogeneous-db, federated-db,...,
data-model, object-based-model,...,
language,...
suggestion:
application, heterogeneous-db,
data-model, language
#2
ask for subjects related to
(discovery AND scientific-db AND application
AND heterogeneous-db AND data-model)
refinement:
application, heterogeneous-db, data-model
#3
suggestion:
federated-db, interoperability ,...,
object-based-model, relational-model,...
related subjects:
federated-db, interoperability ,...,
object-based-model, relational-model,...
#4
refinement:
interoperability, object-based-model,
relational-model
#5
ask for clarification:
1. (object-based-modelAND relational-model)
2. (object-based-model OR relational-model)
#6
ask for subject related to
(discovery AND scientific-db AND application
AND heterogeneous-db AND data-model AND
interoperability AND
(object-based-model OR relational-db))
clarification:
(object-based-model OR relational-model)
#7
...
...

Figure 9 Description of topics of a research lab.

Discovery Mediator

An IF submits a discovery request using the schema of the base ontology as a template, as for export. The request includes the class name of the concepts in which the IF is interested. It may also specify some or all property values of the concepts (see Fig. 10). As in export, the discovery request can be first validated and refined with the help of the DCO. This will make the discovery request precise so that the precision and recall of retrieved results will be high.

Once the request for discovery is constructed, the next step is to retrieve relevant ones from exported concepts. Critical in this step is to measure relevance of exported concepts to the discovery request. For that purpose, we introduce a *relevance factor* (RF), which measures the relevance using the knowledge in the DCO. The RF is first computed for each property, and the final RF is the product of all those relevance factors. Retrieved concepts will be listed with corresponding relevance factors in the decreasing order of the RF. In the following definition of the RF, we will use examples of the discovery request shown in Fig. 10 and the exported concept DBNETLAB shown in Fig. 6b.

For the property whose value is not specified in the discovery request such as owner and institutions:

$$RF = 1$$

For a single-valued or multi-valued property:

$$RF = \frac{\# \mid D \cap O \mid}{\# \mid D \cup O \mid}$$

where D is the set of the property values in the discovery request, O is that of the exported concept, and $|S|$ is the cardinality of the set S. For example, the RF for members is

$$RF = \frac{\# \mid \{McLeod,\ Smith\} \cap \{Aslan, Kahng, Liao, McLeod\} \mid}{\# \mid \{McLeod, Smith\} \cup \{Aslan, Kahng, Liao, McLeod\} \mid}$$

For a composite-valued property,

$$RF = \frac{\# \mid D\ AND\ O \mid}{\# \mid D\ OR\ O \mid}$$

where D is the property value in the discovery request, O is that of the exported concept, and $|C|$ is the frequency of the concept C. For example, assuming conceptual relationships given in Fig. 11, the RF for topics is

CLASS: Research-Lab
owner: *
time-of-entry: *
summary: *
topics: ((database OR network) AND information-retrieval)
institutions: *
members: McLeod, Smith

Figure 10 A discovery request.

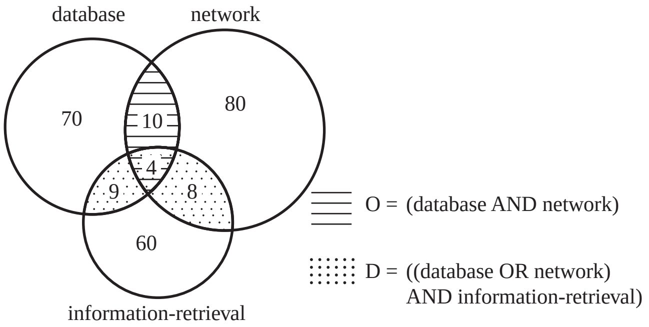

database
network
70
10
80
4
9
8
60
information-retrieval
O = (database AND network)
D = ((database OR network) AND information-retrieval)

Figure 11 Conceptual relationships among concepts of Subject.

$$RF = \frac{4}{4+10+9+8} = \frac{4}{31}$$

The final RF for the above examples is

$$RF = \frac{1}{5} \times \frac{4}{31} = \frac{4}{155}$$

To illustrate the advantages of the RF over relevance measurements in conventional information retrieval systems, suppose that $D = (\mathrm{A}\ AND\ \mathrm{B})$ and $O = \mathrm{A}$ for some composite-valued property in the above notation, and compare the RF with Radecki's coefficient, which measures similarity between two Boolean expressions (Radecki, 1982). It takes the same form as the RF for composite-valued properties, but the interpretation of #|C| is different: C is first converted into a disjunctive normal form $\mathrm{C} = (\mathrm{C}_1\ OR\ \mathrm{C}_2\ \mathrm{OR}\ \ldots\ \mathrm{C}_n)$ such that each C_i contains all the terms that appear in C, and then $\# \mid C \mid = n$. For our example, Radecki's coefficient is

$$\begin{aligned}\frac{\# \mid D\ AND\ O \mid}{\# \mid D\ OR\ O \mid} &= \frac{\# \mid A\ AND\ B \mid}{\# \mid A \mid} \\ &= \frac{\# \mid A\ AND\ B \mid}{\# \mid (A\ AND\ B)\ OR\ (A\ AND\ (NOT\ B)) \mid} \\ &= \frac{1}{2}\end{aligned}$$

regardless of how the two concepts A and B are related. In contrast, the RF is

1. 1, if A is a subconcept of B (i.e., (A *AND* B) = A).
2. 0, if A and B are disjoint concepts (i.e., (A *AND* B) does not exist).
3. Between 0 and 1, depending on how much A and B overlap with each other.

That is, the RF results in more meaningful relevance measurements by taking advantage of the knowledge in the DCO on generality of concepts as well as relationships among them.

5.2.7 Conclusions

We introduced the dynamic classificational ontology (DCO) for mediation of information sharing in the cooperative federated database system environment. In the presence of a large amount of heterogeneous information, it is beneficial to reduce the central coordination and distribute the maintenance cost. To this end, the DCO keeps only a small amount of meta-information, specifically a common ontology to describe and classify information exported by participating IFs, and it is maintained by their cooperative efforts. The classificational object-based data model was introduced as a model that facilitates two-step classification; concepts are classified into classes of basic-level categories, and concepts of each class can be further classified by their property values. While relying on partial agreement among participating IFs, the DCO is progressively established and dynamically adapts to changing usages of concepts.

We have developed an experimental prototype of the DCO, and applied it to document search problems in Medline (a medical information retrieval system provided by the Norris Medical Library at USC) in the context of the USC Brain Project (Arbib *et al.*, 2000). We have developed a datamining algorithm that is advantageous for library systems with deep hierarchies of terms such as Medline. Preliminary results indicate that the precision and recall of document searches in Medline can be significantly improved by the interactive query refinement and the relevance measurement that are supported by the DCO. We are currently extending the system to support browsing and pattern discovery based on the knowledge accumulated in the DCO.

References

Arbib, M. A. *et al.* (2000). USC Brain Project, http://www-hbp.usc.edu/

Arens, Y., Chin, Y., Chee, C.-H., and Knoblock, C. A. (1993). Retrieving and integrating data from multiple information sources. *Int. J. Intelligent Cooperative Inf. Syst.* **2(2)**, 127–158.

Agrawal, R., Imielinski, T., and Swami, A. (1993). Data mining: a performance perspective. *IEEE Trans. Knowledge Data Eng.* **5(6)**, 914–925.

Batini, C., Lenzerini, M., and Navathe, S. B. (1986). A comparative analysis of methodologies for database schema integration. *ACM Computing Survey* **18(4)**, 323–364, December 1986.

Bowman, C. M., Danzig, D. B., Hardy, D. R., Manber, U., and Schwartz, M. F. (1994). The Harvest Information Discovery and Access System. In *Proceedings of the Second International World Wide Web Conference,* October, Chicago, IL, pp. 763–771.

Bright, M. W., Hurson, A. R., and Pakzad, S. (1994). Automated resolution of semantic heterogeneity in multidatabases. *ACM Trans. Database Systems* **19(2)**, 212–253.

Cagne, E. D., Yekovich, C. W., and Yekovich, F. R. (1993). *The Cognitive Psychology of School Learning*. Harper-Collins, New York.

Cardenas, A. F., and McLeod, D. (1990). *Research Foundations in Object-Oriented and Semantic Database Systems*. Prentice Hall, Englewood Cliffs, NJ.

Chamis, A. Y. (1988). Selection of online databases using switching vocabularies. *J. Am. Soc. Inf. Sci.* **39(3)**, 217–218.

Chen, H., and Lynch, K. L. (1992). Automatic construction of networks of concepts characterizing document databases. *IEEE Trans. Syst. Man Cybernetics.* **22(5)**, 885–902.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *J. Am. Soc. Inf. Sci.* **41(6)**, 391–407.

Fankhauser, P., and Neuhold, E. J. (1992). Knowledge based integration of heterogeneous databases. In *Proceedings of the IFIP WG2.6 Database Semantics Conference on Interoperable Database Systems (DS-5)*. November, Lorne, Victoria, Australia, pp. 155–175.

Garcia-Molina, H., Hammer, J., Ireland, K., Papakonstantinou, Y., Ullman, J., and Widom, J. (1995). Integrating and accessing heterogeneous information sources in TSIMMIS. In *Proceedings of the AAAI Symposium on Information Gathering*, March, Stanford, CA, pp. 61–64.

Hammer, J., and McLeod, D. (1993). An approach to resolving semantic heterogeneity in a federation of autonomous, heterogeneous database systems. *Int. J. Intelligent Cooperative Inf. Syst*. **2(1)**, 51–83.

Heimbigner, D., and McLeod, D. (1985). A federated architecture for information management. *ACM Trans. Office Inf. Syst*. **3(3)**, 253–278.

Kent, W. (1989). The many forms of a single fact. In *Proceedings of the IEEE COMPCON Spring '89*. February-March, San Francisco, CA. pp. 438–443.

Kim, W., Choi, I., Gala, S., and Scheevel, M. (1993). On resolving schematic heterogeneity in multidatabase systems. *Distributed Parallel Databases* **1(3)**, 251–279.

Levy, A. Y., Rajaraman, A., and Ordille, J. J. (1996). Querying heterogeneous information sources using source descriptions. In *Proceedings of the International Conference on Very Large Data Bases*. Bombay, India. pp. 251–262.

McLeod, D. (1992). The remote-exchange approach to semantic heterogeneity in federated database systems. In *Proceedings of the Second Far-East Workshop on Future Database Systems*. April, Kyoto, Japan, pp. 38–43.

Mena, E., Kashyap, V., Sheth, A., and Illarramendi, A. (1996). OBSERVER. an approach for query processing in global information systems based on interoperation across pre-existing ontologies. In *Proceedings of the First IFCIS International Conference on Cooperative Information Systems*. June, Brussels, Belgium, pp. 14–25.

Radecki, T. (1982). Similarity measures for boolean search request formulation. *J. Am. Soc. Inf. Retrieval* **33(1)**, 8–17.

Salton, G. (1989). *Automatic Text Processing,* Addison-Wesley, Reading, MA.

Sammet, J. E., and Ralston, A. (1982). The new (1982). computing reviews classification systemsfinal version. *Commun. ACM*. **25(1)**, 13–25.

Sciore, E., Seigel, M., and Rosenthal, A. (1992). Context interchange using meta-attributes. In *Proceedings of the First International Conference on Information and Knowledge Management*. Baltimore, MD, pp. 377–386.

Sheth, A., and Kashyap, V. (1992). So far (schematically). yet so near (semantically). In *Proceedings of the IFIP WG2.6 Database Semantics Conference on Interoperable Database Systems (DS-5)*. November, Lorne, Victoria, Australia, pp. 283–312.

Sheth, A., and Larson, J. (1990). Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys* **22(3)**, 183–236.

Wiederhold, G. (1992). Mediators in the architecture of future information systems. *IEEE Computer* **25(3)**, 38–49.

Yahoo (2000). http://www.yahoo.com/.

Yu, C., Sun, W., Dao, S., and Keirsey, D. (1991). Determining relationships among attributes for interoperability of multi-database systems. In *Proceedings of the First International Workshop on Interoperability in Multidatabase Systems*. April, Kyoto, Japan, pp. 251–257.