MARIAN: Flexible Interoperability for Federated Digital Libraries

Marcos André Gonçalves*, Robert K. France*, Edward A. Fox*,
Eberhard R. Hilf†, Kerstin Zimmermann†, Thomas Severiens†

*Department of Computer Science
Virginia Tech
Blacksburg, VA 24061, USA
Email: {mgoncalv,france,fox}@vt.edu

†Institute for Science Networking
University of Oldenburg
Oldenburg, Germany
Email: {hilf,severiens}@egoiste.physik.uni-oldenburg.de

Abstract

Federated digital libraries are composed of distributed autonomous (heterogeneous) information services but provide users with a transparent, integrated view of collected information – respecting different information sources' autonomy. In this paper we discuss a federated system for the Networked Digital Library of Theses and Dissertations (NDLTD), an international consortium of universities, libraries, and other supporting institutions focused on electronic theses and dissertations (ETDs). The NDLTD has so far allowed its members considerable autonomy, though agreements are developing on metadata standards and on support of the Open Archives initiative that eventually will promote greater homogeneity. At present, federation requires dealing flexibly with differences among systems, ontologies, and data formats.

Our solution involves adapting MARIAN, an object-oriented digital library retrieval system developed with support by NLM and NSF, to serve as mediation middleware for the federated NDLTD collection. Components of the solution include: 1) the use of several harvesting techniques; 2) an architecture based on object-oriented ontologies of search modules and metadata; 3) diversity within the harvested data joined to a single collection view for the user; and 4) an integrated framework for addressing such questions as data quality, information compression, and flexible search. The system can handle very large dynamic collections. An adaptable relationship between the collection view and harvested data facilitates adding new sites to the federation and adapting to changes in existing sites. MARIAN's modular architecture and powerful and flexible data model work together to build an effective integrated solution within a simple uniform framework.

We present both the general design of the system and operational details of a preliminary federated collection involving several thousand ETDs in four different formats and two languages from USA and Europe.

INTRODUCTION

Networked or federated digital libraries are composed of autonomous, possibly heterogeneous information services, distributed across the Internet [Lag98]. The objective of federation is to provide users with a transparent, integrated view of heterogeneous and distributed sources of information. Challenges faced in building such an integrated system include interoperability among different digital library systems/protocols [PCW+98], resource discovery (selection of the best sites to be searched) [GGM97], and issues in data fusion (merging of results into a unique ranked list) [Fox94]. In this paper we focus on the interoperability problem, one of the most challenging in the field of digital libraries. Heterogeneity occurs in both information representation and services, and at four levels: system, structural, syntactic, and semantic.

An interesting example of a federated digital library where heterogeneity is a major problem is the Networked Digital Library of Theses and Dissertations [Pha99], an international federation of universities, libraries, and other supporting institutions focused on efforts related to electronic theses and dissertations (ETDs). Many libraries and universities run their own programs and services, but there also are consortia at the state (OhioLINK), regional (Catalunya, Spain), and national (Australia, Germany, Portugal) levels. NDLTD has particular characteristics that should be taken into account when trying to support interoperability across member systems:

1. Autonomy: (Groups of) universities manage services for their scholars.

2. Decentralization: Members are not (yet) asked to report either collection updates or changes in their metadata to central coordinators.

3. Minimal interoperability: Each source must provide unique URNs and metadata records for all stored works, but need not (yet) support the same standards or protocols.

4. Heterogeneity: There is diversity in terms of natural language, metadata, protocols, repository technologies, character coding, nature of the data (structured, semi-structured and unstructured, multimedia), as well as user characteristics, preferences, and capabilities.

5. Massive amount of data and dynamism: NDLTD already has over 100 members and eventually aims to support all those that will produce ETDs. New members are constantly added and there is a continuing flow of new data as theses and dissertations are submitted.

Due to the primary source nature of the ETD collection, the site selection process that is found in other systems, namely, identifying a small number of candidate databases to choose to search, is not important in our context. For example, a query related to an information need for new results in mathematics should properly retrieve information from almost every member university, and a search for a particular dissertation cannot ignore any member site, lest it miss the unique holding institution.

1. FEDERATED SYSTEMS: REMOTE SEARCH vs. LOCAL UNION

Transparent interoperability involves reconciling heterogeneity and integrating information sources at several levels (e.g., collections, services) [Ada00]. The most common architecture to deal with that problem uses mediators and wrappers [Wie92]. Mediators export a common data model of each source's data and provide a common query interface. Wrappers overcome some barriers of heterogeneity and produce source-specific queries. Wrappers also translate results between source and mediator data models. Within the mediated approach there are two possible architectures to deal with the problem of system integration [Flo98], namely: 1) the warehousing or union archive approach; and 2) the federated search approach.

In the union archive / information warehouse solution [Run00], information is in some way periodically extracted from different sources, processed, merged with information from other sources, and then loaded into a centralized data store – the union archive. Queries are posed against the local data without further interaction with the original sources. Modifications are filtered (for relevance, e.g., or update-time) and propagated in some manner to upgrade the union archive. The main advantage of this approach is that adequate performance can be guaranteed at query time. On the other hand, union archives cannot guarantee delivery of the most current information to users. Thus, concerns about data quality and consistency must be addressed.

In the federated search solution, data remains at the sources and queries to the integrated system are decomposed at run time into queries to those sources. Data is not replicated and is guaranteed to be fresh at query time. On the other hand, more sophisticated query optimization and fusion techniques are required. Performance is also a drawback (see, e.g., [Pow00]). Such factors must be considered as network latency and availability, amount of data to be transferred, etc. The overall performance is bounded by the worst-case situation. In both cases, the challenge is to develop reliable, inexpensive and non-intrusive mechanisms that require neither extensive changes to underlying data sources nor rigid standards that must be followed.

Despite surface similarities with the problems of heterogeneous databases and data warehousing, there are major differences in the digital library scenario: 1) strong multimedia and textual-based components call for information retrieval techniques, with their approximate inference and tolerance to inconsistencies inside the collection; 2) read-only architecture (with only additions and rare updates); 3) necessity to deal with structured, semi-structured, and unstructured documents; 4) relatively simple metadata structures (as opposed to complex database schemas).

In this paper, we present MARIAN, an object-oriented information retrieval and digital library system, and demonstrate how we have used its modular architecture and powerful and flexible data model to create a

federated system for NDLTD while solving most of the problems described above. Because of NDLTD's unique characteristics, and due to poor and inconsistent network connectivities in the global scenario, variability in server load and administration, and considerations about the complexity of query translations in such a heterogeneous environment, we have chosen an information warehouse / union archive architecture for a new version of our integrated system. Components of our solution include: 1) the use of several harvesting techniques; 2) an architecture for building a mediated union archive collection based on object-oriented ontologies of search modules and metadata; 3) a *collection view* mechanism comparable to database view techniques; 4) integrated addressing of such complex questions as data quality, information compression in the indexing of structured documents, and flexible search. Our work is original in the sense that we use the unique characteristics of our system to build a common integrated solution inside a unified framework.

This paper is organized as follows. Section 2 describes MARIAN's architecture and data model. Sections 3 and 4 discuss the union archive and how we make use of the unique characteristics of MARIAN to build our solution. Finally, Sections 5 and 6 approach future and related work.

2. MARIAN DIGITAL LIBRARY SYTEM

MARIAN is an indexing, search, and retrieval system optimized for digital libraries [Fox+93, Fra+99]. It is designed to support a large number of simultaneous sessions of the sort required for library catalogs, namely short sequences of often unrelated queries punctuated by browsing and quick examination of relatively small documents. Thanks to National Library of Medicine funding over the last 2 years, MARIAN is almost fully converted to Java – well over 150K lines of code.

2.1 MARIAN Data Model

The MARIAN data model is based on three main concepts: an *information network* of explicit nodes and links organized into a hierarchy of *classes* in an object oriented fashion where any collection of nodes or links can be *weighted* to represent how well they suit some description or fulfill some role. For instance, a collection may include classes of *Document, Person*, and *Organization* nodes together with *HasAuthor* and *HasAffiliation* links. Weights occur in the weighted *match sets* of objects that satisfy a particular query or in weighted classes, and may also occur in other roles. *HasAuthor* might be a weighted link class, for instance, if links with different provenance had different amounts of authority.

Classes are familiar from object-oriented programming languages like C++ and Java and object-oriented databases like O2. In the context of digital libraries, MARIAN defines classes of *information objects* that support methods, including calculating how well objects of that class match input descriptions. The classes form hierarchies, and behaviors and semantics are inherited from more general classes to more specific ones. The class system allows us to create synthetic union classes as needed. We see an example of that in this paper, where the union class of ETDs is created from classes of structured documents in different formats and from different sources.

Networks have long been used in traditional library and computer representation systems. Through hypertext and the World-Wide Web, information networks have become commonplace. In recent research, networks have become the preferred representation for semi-structured data, like BibTex, HTML, or XML [ABS99], and for translating among different DL systems [Mel00]. MARIAN search modules (searchers) are specialized for a universe where searching is distributed over a large graph of information objects.

Weights are mathematizations of such informal concepts as "importance," "uncertainty," or "goodness of fit". Weights are defined in comparison to other weights, where the principal concept is that of a *weighted set*: a set of objects whose relationship to some external proposition is encoded in their (decreasing) weight within the set. Weights have been successfully used in information retrieval systems [FBY92], probabilistic reasoning systems [Pea88], and fuzzy set theory [Zim91]. The MARIAN model extends them uniformly throughout the entire system.

2.2. MARIAN Architecture

MARIAN is made up of four layers, as shown in Figure 1: a Session Management layer, a User

Interaction Layer, a layer of Search Engines, and an underlying Database and File Management layer.

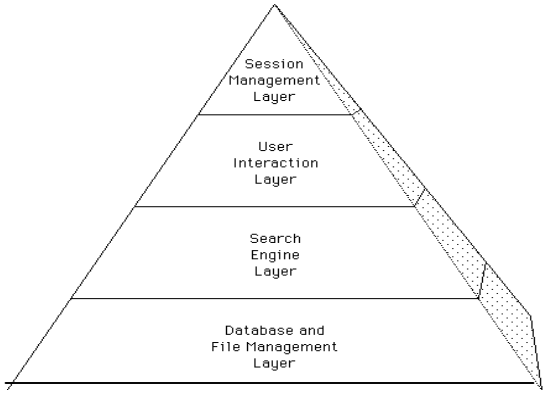

Figure 1: MARIAN architecture.

The Session Management is responsible for starting and terminating user sessions. It also keeps track of the progress of each query and can report to the user on their progress. The User Interaction Layer recognizes users and manages their preferences, history, and personal information storage. The Search Engine Layer provides flexible, content-based search, drawing on the data model discussed in the previous section. Raw data are stored in the Database and File Management Layer using a simple set of data abstractions.

MARIAN uses a set of class-based search engines (*searchers*) built on a common formal model and Application Program Interface (API) that can be used in the collection infrastructure of a wide range of digital library systems. In particular, searchers are available for classes of text objects, structured document objects, and both absolute and weighted links. Each searcher family is extensible to other object classes, ontologies, and models of information.

Lazy evaluation is another important design principle. All searchers in the MARIAN community are designed to do only the work required to return as many elements as are requested. By design and construction, the first elements developed by any searcher are those with the highest weight. This minimizes the delays that result from the typical highly skewed distributions found in large text collections. Lazy evaluation has its greatest pay-off in simple searches authored by human users, few of whom are interested in digesting more than a few dozen retrieved objects.

MARIAN's architecture combines powerful ideas from the information retrieval and database fields. In MARIAN, we make extensive use of object-oriented data and process abstractions to achieve physical and logical independence, common and useful concepts in the database field but neglected in the IR field. Most if not all current IR systems emphasize the physical level of term indexes and weight metrics, making it difficult to integrate systems at a conceptual level

Key features of MARIAN at the level of representation include: explicit nodes and links, classes to both organize data for easier design and exploit data regularities for better performance, and weights and weighted sets. Key operational features are lazy evaluation and reuse of methods, especially common operations of search and similarity matching in semantically related classes. The flexibility of the MARIAN data model is such that it can be used as an object oriented or a semi-structured database, a knowledge representation system or an information retrieval system. Its power comes from the smooth combination of a number of successful concepts of other fields like databases, information retrieval, programming languages, and artificial intelligence.

3. HARVESTING APPROACHES

Any warehouse approach must be based on two building blocks: 1) a mechanism to gather or harvest data from the sources; and 2) some way of combining gathered data for use. This section covers harvesting approaches; Section 4 describes our architecture for combining harvested data.

Electronic theses and dissertations are large, sometimes archived in the form of several files. Many authors include material that would be difficult or impossible to include in printed publications: audio, images, video, simulations, and large collections of primary data. In response to this, a de facto standard has emerged at NDLTD sites of requiring a *title page,* presented in HTML, to serve both as directory to document files and as a convenient point for collecting and publishing metadata. These metadata – title, abstract, committee members, subject descriptors, etc. – are created by the author, usually with faculty/staff oversight. At some sites additional metadata is

generated by trained catalogers. We choose to harvest all metadata – both controlled and uncontrolled – to create images of the sites in the union archive. We do not harvest the original documents, choosing instead to leave them at the remote sites.

Many popular systems provide only a single means of harvesting, concentrating for instance on HTML documents on the Web. Much of current work on federated DLs assumes a homogeneous structure or protocol. (e.g., Dienst [Lag98] and Z39.50 [Lyn97] – both supported by MARIAN). We have been working with two different paradigms for harvesting data from heterogeneous sites: the paradigm proposed by the Open Archives initiative and the one used in the Harvest™ system. In addition, a variety of data has been harvested using ad-hoc source-oriented approaches. The three approaches differ mainly in the support they require from source archives.

3.1 The Open Archive initiative

The Open Archives initiative (OAi) is a multi-institutional project to address interoperability of archives and digital libraries. The two main support mechanisms, as described in the Santa Fe Convention [SL00, SKN+00], are a common XML-based metadata model (the Open Archives Metadata Set, OAMS), and a common protocol based on Dienst for harvesting of metadata. The initiative emphasizes the distinction between data providers and service providers. The former may be the manager of an e-print archive, acting on behalf of the authors submitting documents to the archive. The latter is a third party, creating end-user services based on data in archives.

The OAi framework promotes a effective partial solution for interoperability, but particular archives must agree on implementing the protocol and on translating their exported (meta)data into the common standard, which creates a initial impedance to the solution.

In our efforts concerning Open Archives, we act as both data and service providers. The work of making MARIAN compliant with the Santa Fe convention is concentrated on three fronts: 1) make MARIAN serve as a harvester and a mediation middleware layer, able to deal with the heterogeneity of many specific sources and protocols, including OAi sources; 2) implement the Dienst OAi interface over MARIAN using a light-weight Java implementation; thus providing a search service; and 3) develop an XML transportation format for MARC records.

3.2 Harvest™ system

The Harvest™ system [BDH+95] corresponds to a set of integrated and customizable tools for harvesting information from diverse repositories and building topic-specific content indexes. The architecture of the system is based on two main components: *gatherers* and *brokers*.

Gatherers collect and extract indexing data from repositories. They act as directed crawlers able to get information from topic-specific listed sources. Several parameters can be configured to provide better performance and guarantee data quality from the information gathered. Gatherers extract summaries of content from harvested sources into a specific proprietary format (SOIF).

Brokers provide the indexing and the query interface to the gathered information. They retrieve information from one or more Gatherers or other brokers and incrementally update indexes. Brokers also can filter or refine the information from other brokers. In reference to the Open Archives architecture, brokers can be seen as indexing and searching services over data harvested by the gatherers. Unlike the Open Archives initiative, however, no metadata standard is forced. The gathering crawler both extracts existing document attributes and itself generates some meta-information, like timestamps and identifiers. External metadata standards (e.g., Dublin Core) can be incorporated. Some specific and useful entry points in the collection (e.g., a browsing list of ETDs) must be provided to allow effective control of the crawler and guarantee quality of the generated data.

3.3 Other data sources

Data from many library collections can be harvested using the Z39.50 paradigm. This paradigm is better suited to federated search systems than union archive systems, but can be used to harvest with the correct support at the server end. MARIAN can import records harvested through Z39.50 servers in the MARC standard format for library interchange.

We have faced situations where we cannot use any of these approaches, but where specific ways to gather data from sources exist. For example, in sources that use the Dienst protocol but do not implement the complete OAi interface, we use other Dienst services to harvest data. Also, because Virginia Tech functions as coordinator of NDLTD efforts, we developed ad hoc ways to gather metadata in heterogeneous formats. A case in point is the Virginia Tech ETD collection, which provides us with dumps of local thesis and dissertation metadata from an SQL database, which we then translate into OAMS format. The obvious drawback to ad hoc conversions is that they require development of specific solutions that are strongly dependent on the source.

4. UNION ARCHIVE ARCHITECTURE

The NDLTD union catalog uses a mediation / wrapper architecture, modified to make it more extensible. We have prepared a special XML DTD in an extension of [PF98] as a language to describe digital library sources. In our union archive approach, we use several object-oriented metadata document classes, which abstract the characteristics of the several standards, to produce an intermediate MARIAN representation file, useful for stemming, parsing, and indexing.

Just as there are many differences among the participating institutions in NDLTD, there also are differences among the collections, especially regarding document format. NDLTD does not specify the format in which participant institutions maintain either documents or their metadata, and although standards are emerging for both, it is unlikely that a single format will win out over all others. Consequently, our union collection must cope with a multiplicity of formats.

In the prototype union collection described here, we have harvested metadata in four formats. The German Physics collection of ETDs (called here "PhysDis") comes to us in SOIF format, including both Dublin Core and uncontrolled attributes produced by their local crawler. The Virginia Tech ETD collection is available both as metadata created by authors in the Open Archives Metadata Standard and as records created by professional catalogers in MARC format. And the MIT ETD collection comes to us as RFC1807 records received via the Dienst protocol. Our approach is to work with the metadata as it arrives rather than trying to translate these different formats into a single format, and to fuse the collections as we present a unified view to users. Advantages of this solution will be explored below.

The primary device for unification is the *union superclass*. Digital information objects in the MARIAN system are all part of a class hierarchy. In particular, all ETDs and all the metadata formats mentioned are subclasses of *StructuredDocument*. We begin by defining a class for the local image of the document collection belonging to each institution. In a case like Virginia Tech where we have disparate but overlapping sources of metadata, we create a class for each format and a general superclass to mediate among them. Finally, we define the synthetic superclass *ThesisDissertation* (Fig. 2). Since all the subclasses involved are *StructuredDocuments*, any *Thesis-Dissertation* will be a *StructuredDocument* as well; mapping its structure onto the structure of the subclasses is the main part of its class definition.

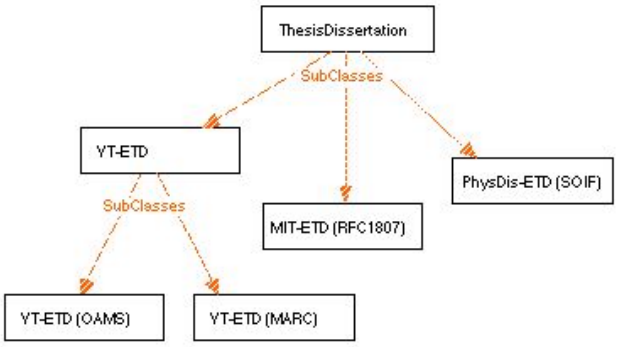

Figure 2: The ThesisDissertation class is a superclass of ETDs from separate institutions. Some institutions (in this prototype, Virginia Tech) may support separate versions of their collection that need to be merged before being reported.

Harvesting itself serves as a device to suppress some differences in finding aids such as indexes and ontologies. Once we have harvested metadata from each remote collection and built local images for each, we can treat the local data with a unified set of text parsing, indexing and retrieval tools. Document (metadata) text fields such as *title, abstract,* or *body* are reduced to their individual terms using the same set of parsers, then matched to users' queries using the same search algorithms and ranking formula. This way we can ensure that the smallest atomic components, the text fields, will receive uniform treatment.

The next problem to address in combining collections is that these atomic text components serve different purposes in different collections; or in other words that the structure of documents is different in different collections. Different collections support different document attributes, and represent those attributes with different structures of data. Similar structures can be given different names by different collections. Structures with similar names may have very different semantics. Finally, the same purpose can be addressed by semantically different fields.

Even within collections, there may be differences in document structure. MARC records in the VT-ETD collection make a strong distinction between personal and corporate authors, while the <*author name*> field of OAMS records may contain either. As another example, some documents from the PhysDis collection are represented with Dublin Core metadata, including *dc.subject*, while others describe the subject with lists of automatically extracted keywords.

4.1. Presenting collection views

In summary, documents in the harvested collections have multiple attributes – some redundant and some complementary. (This same situation can obtain within a single source collection; see the discussion of the PhysDis collection in Sec. 4.2). This complexity is potentially confusing for users of the union. We address these structural issues by capturing the structure of each collection in a way that is sensitive to the collection, then create a *collection view* ontology analogous to a database view but for the information network model. The view is composed of superclasses of entities and attributes from member ontologies, which are combined by the searchers for each superclass to create the presented view. Each remote collection image in the union collection has a different structure (Fig. 3). We use node and link classes to faithfully capture that structure, thus minimizing labor expended and information lost during harvesting.

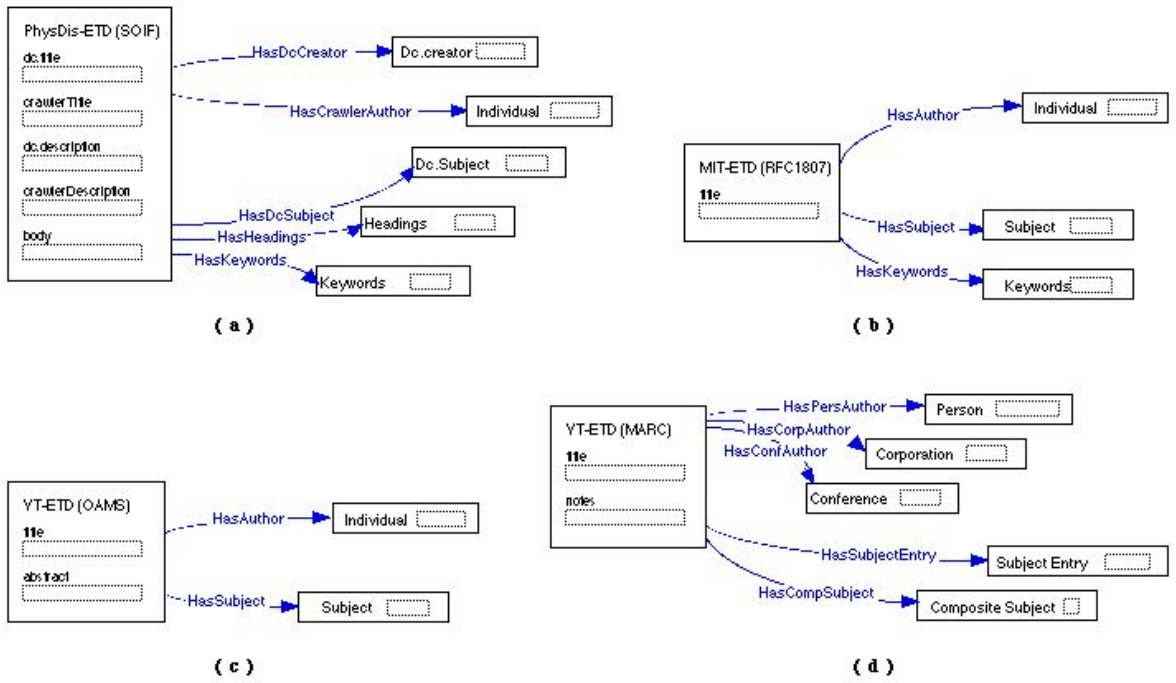

Figure 3: Images for (a) the SOIF PhysDis collection, (b) the MIT RFC1807 collection, (c) the VT OAMS collection, and (d) the VT MARC collection, all represented as class networks.

For purposes of simplicity and familiarity, we have chosen to present a global view based on the Dublin Core model (Fig. 4). Four attributes are presented to the user: title, author, subject, and description. In accordance with MARIAN's networked information model, the view ontology actually consists of three

classes of objects: *ThesisDissertation*, *Individual* and *Subject*, together with *HasAuthor* and *HasSubject* links. The *Individual* class subsumes both persons and corporate individuals, and the *Subject* class covers a welter of possible treatments. This view can be modified or extended as future usability requires. More importantly, the connections between the view and the underlying structure can be modified without affecting what the user sees.

Figure 4: The synthetic view presented to users consists of the ThesisDissertation class, a class of Individuals linked to ThesisDissertations by an authorship relation, and a linked class of Subject descriptions.

4.2. Extended example: the PhysDis Collection

To illustrate the complexity of representing heterogeneous collections, consider the example of the PhysDis collection. We have harvested 1256 documents from the collection using Harvest™. All of these present fields in SOIF format, although there is no single field that is present in every document. In addition, 166 of the documents contain controlled Dublin Core metadata, which is presented together with uncontrolled metadata in the SOIF record.

The two sorts of metadata are neither closely related nor mutually exclusive. All documents presenting Dublin Core metadata also present some form of uncontrolled metadata but there is no single uncontrolled attribute occurring in every Dublin Core document. The nearest is *keywords*, which occurs in 97% of documents with Dublin Core attributes while appearing in only 88% of the collection as a whole. There is little overlap between the values in Dublin Core and Harvest metadata attributes even where the attributes can be expected to have similar semantics, in spite of the fact that a small but significant fraction of documents have both Dublin Core and uncontrolled attributes. When we consider the values that appear in the attributes, the differences appear in significant ways. This can be attributed, at least in part, to imprecision on the part of the local crawler, which generates titles like "Dissertation" (58 occurrences), "Dissertationen" (52 occurrences), and "Archiv Publikationen URZ TU Chemnitz" (42 occurrences).

Of the document attributes returned by the Harvest™ software for the PhysDis collection, we have chosen to represent ten in our union collection: the Dublin Core attributes *dc.creator, dc.description. dc.subject* and dc.*title*, and the free attributes *author, body, description, headings, keywords,* and *title*. We regard the various sorts of titles and descriptions, as well as *body*, as attributes of the documents themselves, while regarding the names in *author* and *dc.creator,* as well as the descriptive strings in *dc.subject, headings* and *keywords* as first-class objects connected to the documents with *HasAttribute* links (Fig. 3a).

We maintain all this structure and complexity in the local image of the PhysDis collection. At the same time we want to be able to present a simplified view to the user. We serve both goals, as well as the ultimate goal of providing a simple view of the complete global union collection of ETDs, by the judicious use of ontologies of superclasses representing collection views.

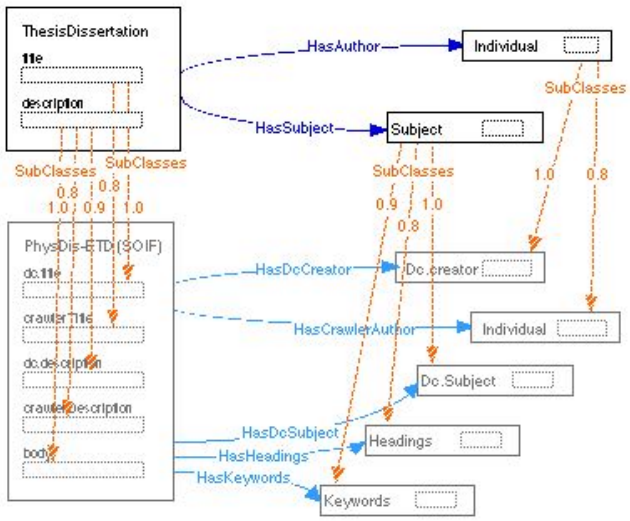

Figure 5: The collection view is abstracted from the PhysDis data to increase retrieval and usability

The PhysDis collection provides a good example of view presentation and the use of weights to enhance data quality. Each text class in the view corresponds to two or three classes in the underlying collection: a Dublin Core class and at least one uncontrolled class (Fig. 5). Our observations of the data indicate that the Dublin Core texts are of better quality than the

uncontrolled texts. The superclass searchers capitalize on this by giving more weight to DC subclasses. In addition, the *Description* superclass depends more heavily on the PhysDis *Body* attribute than on either DC or uncontrolled description attributes, because we have observed that *Body* text tends to be a better representation of document content. All of these weights can be tuned as our experience with the union collection increases.

Similar connections are made between the view superclasses and the other ETD collections, with weights chosen to maximize reliability and effectiveness of retrieval. In addition, weights can be tuned between the classes, both to promote one subclass over another, as in the PhysDis example, and to mediate artifactual differences between the collections. An example of the latter would be trimming similarity values caused by statistical differences between the various collections. Most of the weight-values functions used to measure similarity between a query and a document (text) are sensitive to the distribution of attributes or terms across the collection. We can impose adjustments at the superclass levels to mitigate these disparities.

4.3. Combining Heterogeneous Collections

Each remote collection image in the union collection has a different structure, and none has exactly the structure of the presented collection view (compare Figures 2 and 3). We use node and link classes to capture the structure of the remote collections as faithfully as possible, thus minimizing the labor expended and the information lost during harvesting. Superclass relationships are then used to define the collection view in terms of the image structures (Fig. 6).

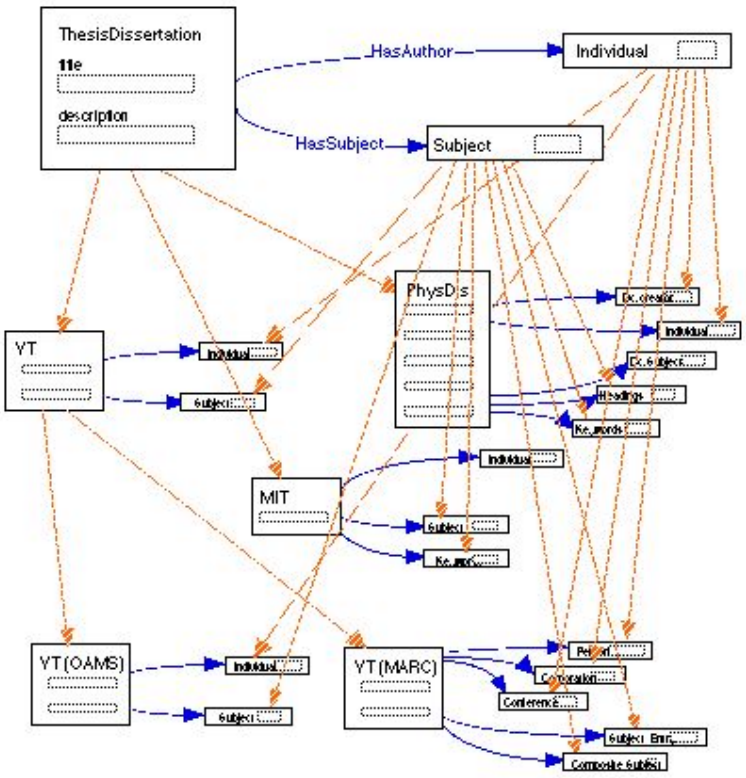

Figure 6: The complete union collection relates the collection view to each remote collection image.

The organization of the retrieval system mirrors this information organization. Each class of objects and links in the representation is managed by a *class manager*, which functions as a searcher for objects in that class. All the searchers needed for the union archive are of five standard types: superclass searchers, text and structured document searchers, and weighted and absolute link searchers. A resource

manager allows class managers to discover each other. During search, queries are disassembled by the invoked class manager; any parts that the class manager cannot handle itself are passed to others. Thus for instance an author / title search over the entire collection begins at the *ThesisDissertation* class. The title portion of the query is handled locally; the author portion is passed to the link class manager for *HasAuthor* links, which passes the operation of finding matching people or corporations to the *Individual* class manager. In this way, each piece of text can be treated appropriately. MARIAN, for instance, treats title text as a special sort of natural language sequence, with various rules for capitalization, punctuation, and sentence formation, but treats person's names as vectors of fixed strings.

In a simple MARIAN collection, both the *Individual* class manager and the title searcher within *ThesisDissertation* would be text searchers, but in the union archive both are superclass searchers that copy their query portions to the managers for their subclasses, then reassemble the subclass match sets into a single unified set. Again, differences among the different fields are handled by different treatment in the target searchers. In NDLTD, for instance, we can assume that titles from VT and MIT will be in English, so we invoke an English language text searcher to recognize English terms under various morphological transformations. Titles from the PhysDis image may be in either English or German, and we have no way of telling which are in which language, so the searcher for PhysDis titles does not use morphological analysis.

Statistical differences among the collections can be addressed by tuning the relative weights among the subclasses of a superclass searcher, as can differences in reliability of the data. It is in this step, for instance, that we choose between controlled and uncontrolled attributes of both the PhysDis and VT collections, balancing data quality against availability. Finally, structural differences are handled in this step. The *ThesisDissertation Description* manager, for instance, is a superclass manager that mediates among the different types of free text fields among the collection images (compare Figure 3). Subclass weights allow us both to mitigate the statistical vagaries induced by some free text fields being much larger than others on the average, and to impose figures of merit on how much a match in a particular text field counts as a match in the document as a whole.

5. FUTURE WORK: Merging Ontologies

Combining collection images into the union collection as depicted in Figure 6 involves a fair amount of redundancy in the four images. For instance, several site images include classes for *Individual* and *Subject.* This redundancy raises a design issue: what should be the ontology for the overall union collection?

The current prototype (Fig. 6) embodies one extreme, where all the images are completely separate and only subclass-superclass relationships tie the object classes together. This approach has the disadvantage of data duplication: the same object (e.g., the subject heading "Computer Engineering") may appear in several classes. Such redundancy wastes storage space in the class managers, and can increase retrieval time when multiple classes must be searched simultaneously.

At the other extreme, we could immediately force all harvested data into our collection view by processing all incoming documents into structures with a single title and a single description field, all types of individuals into a single class, and all types of subjects and keyword lists into a class of subject strings. This would have the disadvantage of forcing us to combine fields as unlike as the PhysDis *Body* and *DcDescription* fields into a single text, with corresponding losses to indexing specificity. It also would mean losing the information that sometimes we *do* know when an individual is a person, or when a subject heading comes from a controlled vocabulary.

Most importantly, however, pre-processing incoming data into the collection view ontology would mean giving up the ability to adjust to changing circumstances. Once our image of a remote collection has been cooked, we can no longer reconstruct it in its raw state. On the other hand, the more original structure we retain, the better we can react to changes in the original collection, to addition of new collections, and especially to changes in our understanding of the semantics of the original data and how best we can present it to fill the needs of our users.

Between these two extremes lies a third alternative: we can merge image classes when these have sufficiently similar semantics, and keep classes separate when the semantics are different. Figure 7 shows this approach for the four images currently in

the NDLTD union collection. The document hierarchy is as it was shown in Figures 2 and 6. The *Individual* class has been analyzed into classes of (human) *Persons* and *Corporations*. OAMS, RFC1807 and Dublin Core author fields and MARC x00 fields, all of which require the name of a human person, are mapped to *HasAuthor* links to the *Person* class, while the MARC x10 fields produce links to the Corporation class. An *UncategorizedIndividual* class provides a sink for those formats that make no such distinction, like the uncontrolled *author* field of the PhysDis collection. A similar breakdown of the *Subject* superclass into individual subject entries, composite strings with multiple entries, and sets of keywords provides sink classes for all types of subject fields within the union collection.

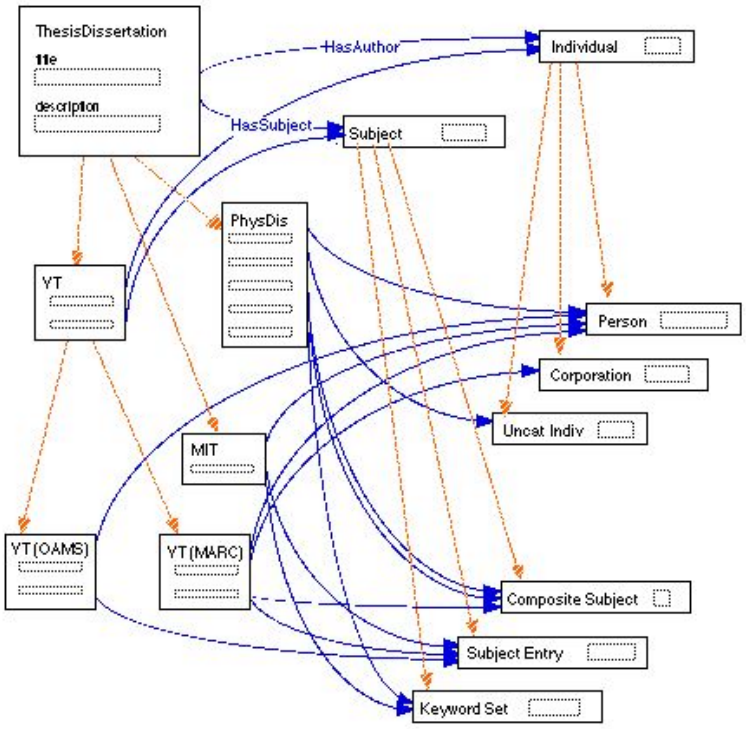

Figure 7: A more sensitive approach to the union catalog allows overlapping of semantically similar object classes.

This approach simplifies the union collection ontology, with corresponding benefits in administration time and effort. It also allows for savings in string storage and retrieval time. On top of these advantages, it admits added functionality to the system.

One option that NDLTD wants to provide its users is the ability to search either in a single institution or across the union catalog. As is often the case, while this gives the user more power it also opens the possibility of mistakes and miscommunications. For instance, imagine a user searching for a given author who has accurate knowledge of the author's name, but believes mistakenly that her degree was awarded at Virginia Tech. Searching for documents of class *VT-Etd* with links to the author name would produce no results, even though that author was recognized by the *Individual* class.

Working with the union ontology of Figure 7, however, we can ameliorate this problem by redirecting all requests for theses and dissertations through the *ThesisDissertation* class, but invoking a different balance of subclass weights for theses that are presumed to come from a particular institution. For instance, the relative weights on *HasAuthor* links from each institution might be relatively equal where no affiliation was known, but where an affiliation was supposed,

the weights for all other institutions could be set an order of magnitude lower than the expected institution. When the author in question was found at the expected institution, this would effectively screen out any "drops" at other institutions, but when matching authors were only found at other institutions, they could still be reported to the user.

Thus, combining weights, networks, and class structures enables us to both respect the data as it is harvested and provide simplified virtual collection views to users. It also makes it easy to change either the collection ontology or the underlying data without changing the view presented to the user or to change the view presented to the user without restructuring the underlying representation or data. Moreover, it provides a unified framework to enhance retrieval effectiveness in the union archive system by providing the flexibility to use different configurations and priorities on the same underlying data.

6. RELATED WORK

As pointed out, our work is related to a number of efforts. However, most of the current efforts for interoperability among federated digital libraries assumes some level of homogeneity in some of the main components, as in the case of NCSTRL [Lag98] or Z39.50 [Lyn97].

The Stanford InfoBus project [Bal97, Mel00] presented a approach based on a federated search and used high-level descriptions for mapping between de-facto metadata standards (e.g., Dublin Core, USMARC, Z39.50 Bib1) to achieve interoperability between heterogeneous repositories. Metadata attributes of those standards (e.g., *author* in Dublin Core) are considered first-class objects with their own descriptions that specify type and semantic content (which is specified using human-readable descriptions). Using these descriptions, application developers manually build one-to-one translation services. As discussed before, federated search has its drawbacks in our context. Moreover, in Stanford's project there is no consideration for integrated information retrieval searching services, aspects of data quality, flexibility and scalability of the solution.

Most of those approaches do not consider several aspects of semantic interoperability, for example, vocabulary or conceptual mismatching. The full problem of semantic interoperability is even comparatively more difficult and involves modeling, capturing, representing, and reasoning about the semantics of the several pieces of the systems.

Recently some systems have paid more attention to semantic aspects and conflicts. In particular ontologies have been used in those systems, since they enable a standardization of semantics content. The OBSERVER system [Men96] uses ontologies and description logics to solve the problem of aligning different vocabularies, which describe similar information across domains. User queries are rewritten using inter-ontology relationships, which allow semantics-preservation translations. Velegrakis et al. [VCV99] employ description logics to enhance Z39.50 wrappers with declarative semantic descriptions, thus providing a high-level mapping from Z39.50 attributes to underlying source data structure and semantics. TopiCA [PH99] uses semantic information represented as ontologies about metadata in different systems to contextualize and partition information in several subject-specific categories. Those systems work with different spaces of the problem or with different architectures and none of them tackles so many aspects of the problem or are so comprehensive as our system.

References

[ABS99] Abiteboul, S., Buneman, P. Suciu, D., *Data on the Web: from relations to semistructured data and XML*. Morgan Kaufmann, 1999

[Ada00] Adam, N., Atluri, V., Adiwijaya, I., "Systems Integration in Digital Libraries", *Communications of the ACM*, **43**(6), 2000, pp. 64-72

[Bal97] Baldonado, M., et al., "The Stanford Digital Library metadata architecture." *International Journal on Digital Libraries*, 1997. **1**(2): pp. 108-121.

[BDH+95] Bowman, C. M., Danzig, P. B., Hardy, D. R., Manber, U., Schwartz, M. F., "The Harvest information discovery and access system", *Computer Networks and ISDN Systems*, **28**(1–2), 1995, pp. 119–126

[CGP98] Chang, K. C., Garcia-Molina, H., Peace, A., "Predicate Rewriting for Translating Boolean Queries in a Heterogeneous Information System", *ACM Transactions on Information Systems*, **17**(1), January 1999, pp. 1-39

[Fox+93] Fox, E.A., R.K. France, E. Sahle, A.M. Daoud, and B.E. Cline: "Development of a Modern OPAC: From REVTOLC to MARIAN. *Proc. of the 16th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1993: pp. 248-259

[Fra+99] France, R.K., L.T. Nowell, E.A. Fox, R.A. Saad, and J. Zhao: "Use and usability in a digital library search system." CoRR cs.DL/9902013:

[GGM97] Gravano, L., Garcia-Molina, H, "Merging Ranks from Heterogeneous Internet Sources", *Proc. of the 23rd International Conference on Very Large Databases, 1997*, pp. 196-205]

[FBY92] Frakes, W.B. and R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms.* Prentice-Hall, 1992.

[Flo98] Florescu, D., Levy, A., Mendelzon, A. "Database techniques for the World-Wide Web: A Survey", *SIGMOD Record*. **27**(3) 1998, pp. 59-74

[Lag98] Lagoze, C., Fileding, D., Payette, S., "Making Digital Libraries Work: Collection, Services, Connectivity Regions, and Collection Views", *Proc. 3rd ACM Conference On Digital Libraries*. 1998, pp. 134-143

[Lyn97] Lynch, C., "The Z39.50 Information Retrieval Standard - Part I: A Strategic View of Its Past, Present and Future", *D-Lib Magazine*, April 1997.

[Mel00] Melnik, S., H. Garcia-Molina and A. Paepcke, "A Mediation infrastructure for digital library services." *Proc. 5th ACM Conference On Digital Libraries (San Antonio, June 2-7, 2000)* pp.123-132.

[Men96] Mena, E., *et al.* OBSERVER: "An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies." in *Proc. of the First International Conference on Cooperative Information Systems (CoopIS'96)*. 1996, pp. 14-25

[PCW+98] Paepcke, A., Chang, C. K., Winograd, T., Garcia-Molina, H., "Interoperability for digital libraries worldwide." *Communications of the ACM* **41**(4), 1998, pp. 33–42.

[PH99] Papazoglou, M.P. and J. Hoppenbrouwers, "Contextualizing the Information Space in Federated Digital Libraries". *Sigmod Record*. **28**(1) , 1999: pp. 40-46

[Pea88] Pearl, J., *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufman, 1988

[Pha99] Phanouriou, C., Kipp, N. A., and Sornil, O., Mather, P., and Fox, E. A., "A Digital Library for Authors: Recent Progress of the Networked Digital Library of Theses and Dissertations", *Proc. 4th ACM Conference on Digital Libraries*, 1999, pp. 20-27

[Pow00] Powell, A.L. and J.C. French, "Growth and server availability of the NCSTRL digital library." *Proc. 5th ACM Conference On Digital Libraries (San Antonio, June 2-7, 2000)* pp. 264-265.

[PF98] Powell, J., Fox, E. A., "Multilingual Federated Searching Across Heterogenous collections", *D-Lib Magazine*, September 1998

[Run00] Rundensteiner, E., Koeller, A., and Zhang, X., "Maintaining Data Warehouses over Changing Information Sources", *Communications of the ACM*, **43**(6), 2000, pp. 57-62

[SL00] Sompel, H., Lagoze, C., "The Santa Fe Convention of the Open Archives Initiative", *D-Lib Magazine*, February 2000

[SKN+00] Sompel, H., Krichel, T., Nelson, M. L., Hochstenbach, P., Lyapunov, V. M., Maly, K., Zubair, M. K., Liu, X., O'Connell, H., "The UPS Prototype project: exploring the obstacles in creating a cross-print archive end-user service", *D-Lib Magazine*, February 2000

[VCV99] Velegrakis, Y., V. Christophides, and P. Vonstanopoulos. "Declarative Specification of Z39.50 Wrappers Using Description Logics". *Proc. European Conference in Digital Libraries (ECDL'99)*. 1999**,** pp.383--402

[Wie92] Wiederhold, G., "Mediators in the Architecture of Future Information Systems", *IEEE Computer*, **25**(3), 1992, pages 38-49.

[Zim91] Zimmermann, H.-J., *Fuzzy Set Theory and its Applications.* Kluwer, 1991 (2nd. Ed).