NSC90　2213　E　009　082

90　　8　　1　　　91　　7　　31

91　　　　　10　　25

The Research on A Metadata Management System of Distributed Digital Libraries

(CSIM)

Abstract

The distribution of a large amount of information over the Internet depends on a well-established architecture to support digital library (DL) queries. However, the conventional keyword-based query is a non-semantic way to search for DL information. This study considers the relationships between the two most important integral aspects of digital libraries, i.e., content and service. A Content and Service Inference Model (CSIM) is also proposed for semantic DL queries. CSIM categorizes content and service into 15 relationships and clarifies the interactions among them. The manipulating functions in each relationship are described and a method for semantic queries of both content and service are provided, using these functions. CSIM can be applied to a DL semantic search to provide a recommended list of available services and contents, advising users to combine related contents and services for achieving their original semantics. Experimental results show that CSIM outperforms the conventional keyword-based method. CSIM improves the DL query as a semantic and effective method, alleviating the administrative load of developing new DL content and services.

Keywords: Semantic query, content and service inference model, digital libraries, information retrieval.

Introduction

The rapid revolution in information technology has accelerated globally the worldwide access to information. A well-designed architecture is required to coordinate effectively and efficiently the distribution of a large amount of information on the Internet (Monch, 1998; Nikolaou, 1999; Grossman 1995). Having received considerable attention in recent years, digital libraries represent an Internet-based architecture to access all kinds of information from anywhere in the world. Most digital libraries use a conventional keyword-based approach to search for information. Such an approach is a non-semantic way to retrieve information because different information may contain various keywords with the same semantics. Consequently, the eagerness for information retrieval has led to the use of metadata to annotate information, since metadata can include extensive semantics.

Metadata describe information about data in a structured manner (Tim, 1997). Metadata can enhance the power of a Digital Library (DL) query because they provide additional semantic and structural annotations of original raw data. Some research has addressed the feasibility of using metadata to describe resources in digital libraries, using approaches including the Warwick Framework (Lagoze, 1996), Dublin Core (1999), Resource Description Framework (RDF, 1999), Metadata Modeling Language (MML)(Huang, 2000; Ke, 2001), and several others

(Kahn, 1995; Berners-Lee, 1997; McCray, 1999). The Warwick framework is a conceptual model of metadata that aggregates metadata packages into containers and then relates these packages to each other. This framework separates the management and responsibility of specific metadata and allows access to various different sets of metadata. The container technology has influenced subsequent developments, including Dublin Core and RDF. Dublin Core, which defines 15 basic metadata elements, focuses mainly on resources for Internet-based applications. RDF provides a standard means of representing metadata by XML, employing statements to describe properties of and relationships among items on the Web. Many studies have applied RDF to resource discovery (Jenkins, 1999, Hu, 1999; Huang, 2000; Ke, 2001;). For example, the Multimedia Description Framework (MDF) uses RDF to describe multimedia contents (Hu, 1999). Another example is Metadata Modeling Language (MML) (Huang, 2000; Ke, 2001), which regards metadata in a framework to manage the collection and as a foundation for representing and delivering data in digital libraries at National Chiao Tung University (NCTUDL). Transformations among various MML metadata enable digital libraries to diversity of information and enhance the interoperability of metadata.

Considerable attention paid towards DL architecture has enabled DL queries across distributed DL services using metadata (Paepcke, 1996; Nikolaou, 1999). A well-designed DL architecture should support not only the effective manipulation of digital contents but also the autonomous management of collections of distributed services (Arms, 1995; DLPS, 1999; Melnik, 2000). In mediation schemes (Paepcke, 1998), metadata describe and translate information. The metadata architecture proposed by Stanford University, called InfoBus, is one realization of mediation architectures. Infobus uses five service layers to enable uniform access to distributed heterogeneous information resources and services (Baldonado, 1997; Roscheisen, 1997). In InfoBus, *Attribute Model Proxies* model metadata[1] as first class objects, and *Search Service Proxies* hold metadata that describe the general characteristics of DL services. These two kinds of metadata help Stanford digital libraries to communicate their content and establish an attribute model (metadata), supported by a specific DL service.

Metadata in DL architecture also attempts to facilitate communication for interactive architectures. Agent-based architecture is an example of such DL architecture (Grossman, 1995; Derbyshire, 1997). The agent-based architecture helps new DL agents to be developed and desired agents to be located. The University of Michigan proposed an agent-based

_____

[1] The term *attribute model,* used by Stanford University, has an equivalent meaning as metadata.

architecture, called University Michigan Digital Libraries (UMDL). UMDL refines the basic agent architecture to satisfy the needs for an open information economy. UMDL expresses agents using ontological semantics and employs metadata to represent information. UMDL classifies ontology using a hierarchical taxonomy and each agent contains it own ontological set. Agents communicate with each other by Knowledge Query and Manipulation Language. Furthermore, a market-based auction protocol in UMDL architecture can easily locate an agent that can fulfill user requirements.

Digital libraries handle interactions and manage the interaction among digital content and services. Many aspects of digital libraries, mentioned in the literature are outlined below (Lynch, 1995; Monch, 1998):

- *A digital library is a collection of widely distributed, autonomously maintained services.*
- *A digital library stores material in electronic format and effectively manipulates large collections of such material.*
- *A digital library is a repository of on-line information, to support a broad user community to exchange information, cooperate and solve problems.*

In sum, a digital library first collects digital information (content) into its repositories and then communicates with users via services. Accordingly, a digital library can be considered as an interaction of content and service. Restated, *content* and *service* are the two most important elements of any digital library. Content represents all material stored in a digital library, including texts, images, and videos. Service is an application that interacts with users via an interface, and converts content into various formats to meet the need of user.

Metadata that include semantic information concerning DL content and service motivate the derivation of semantic relationships among the metadata. The aim of this work is to analyze the semantics of the metadata concerning DL content and services, to assist DL queries. In addition to presenting a digital library framework that exploits metadata to model DL content and service, this work attempts to elucidate the interactions among the metadata.   This framework is called the Content and Service Inference Model (CSIM). CSIM defines 15 relationships among DL content and service. These relationships enable semantic DL queries to be answered, and the performance of querying DL contents and services to be enhanced using semantic DL queries. Moreover, CSIM contains an inference algorithm that derives related service lists and content from the semantics in metadata. Given the derived service lists, DL designers could develop new services that involve existent services components and translation rules. In this manner, the reusability of DL components can be

increased and the administrative load of maintaining a DL can be reduced. In content retrieval, semantic DL queries expand conventional keyword-based queries by returning content with single semantic or content that can be translated into a single format demanded by the queries. A series of experiments are conducted to demonstrate that CSIM outperforms keyword-based DL queries. Applying CSIM to the digital library raises DL queries to a semantic and effective level (retrieve data with a single semantics) and reduces the administrative load associated with developing new content and service. CSIM can also be applied in several domains, including the DL user interface, semantic DL queries and library resource planning systems. In the following section, CSIM in a virtual union catalog system was demonstrated to perform very well.

Content and Service Inference Model (CSIM)

As stated above, the feasibility of using metadata to facilitate DL queries has received considerable attention. The conventional keyword-based approach cannot easily clarify two critical elements of DL queries - content semantics and service capability. Metadata are useful for manipulating semantic information on the content schema and service capability. Manipulating content and service metadata improves DL queries in many ways. First, metadata that describe content schema and service capability contain comprehensive semantics concerning content and service. These semantics help to provide accurate information in response to DL queries. Second, according to the semantic information embedded in metadata, relationships between DL content and service can be derived; manipulating these relationships yields further semantics in the metadata. Third, metadata can be easily stored and indexed to support the retrieval of data, due to their structure format.

Content and services are two integral aspects of digital libraries. Content is the material in which semantics are stored in digital libraries, includes web pages, holding records, and multimedia data, like text, images, and videos. Restated, all information handled by services can be considered as content. Service refers to application that transforms content into useful information. Service interacts directly with users via input and output interfaces. Additionally, content and service can be structured by metadata. This work presents a novel framework called the Content and Service Inference Model (CSIM). CSIM derives relationships between content and service, by examining their metadata. These relationships can raise DL queries to the semantic level, such as using content semantics, service capabilities and the relationships between content and service.

Typically, the content metadata in all instances contain an identifier, data schema, format of presentation and semantics. The metadata of services include an identifier and a statement of a set of service capacity. Content interacts with services according to many relationships. A total of fifteen relationships are defined, as illustrated in Fig.1. These relationships can be categorized into four types.

Figure 1. Content and Service Inference Model

Service to Service. Two services can be related according to their capabilities and input/output interfaces. Eight relationships of this type are defined - *Identical*, *Inclusive*, *Homonymous*, *Synonymous*, *Replaceable, Translatable, Combinable*, and *Combine*. For example, two services are "*Synonymous"* if they involve the same capabilities but use different input/output interfaces. One service is "*Inclusive"* of another if the first has more capabilities than the second.

Content to Content. Two contents can be related based on their semantics and schema. Five relationships of this type are defined - *Identical*, *Homonymous*, *Synonymous* and *Translatable*, *Inheritfrom*. For example, two contents are "*Identical"* if they have identical semantics and format. Additionally, translation rules can apply among content. Different types of content can be translated into a single schema by applying the translation rules. A Metadata Modeling Language (MML) is developed to translate one piece of metadata into another (Huang, 2000).

Service to Content. One service can produce content. One relationship of this type is defined - *Produce*. For example, a WebPAC system may produce a data set in Dublin Core format.

Content to Service. One item content can be produced by a service. One relationship of this type is defined - *ManipulatedBy*. For example, various data sets can be manipulated by a virtual union catalog system to produce a synthetic data set.

Semantic Digital Library Query

Applying CSIM to digital libraries can realize powerful semantic querying concerning content and service retrieval. In CSIM, content and service semantic can be elaborated more finely than conventional keyword-based approach as described in the introduction. Using this abundance of semantic information, CSIM accurately retrieves results and derives alternative answers that conventional approaches can not generate. For example, content

with particular semantics can be retrieved by deriving other content in the same schema hierarchy and with identical semantics, but in a different format, such as synonymous related content. Content with various formats, yet translatable into a single format, can be also retrieved in response to a semantic digital library query. In particular, in a semantic service query, CSIM can infer a list of recommendation to suggest that a user concatenates available services into the desired service, using combinable and translatable relationships.

Content query

A content query inquires about content specified in the query. A user can specify an exact or ambiguous query. An exact query returns content that entirely satisfies the query. An ambiguous query determines all content that "can be" the same semantics as the metadata specified in the query. "Can be" means that the content may be translated into, or inherited from the target content.

Basic Content Query

Example: Determine content with the schema "Dublin Core".

Query Statement: *Select C.Id From Content C where C.Sche = "Dublin Core"*

Algorithm: ContentQuery(Attributes A, Contents C){
1. Locate content c, let c $\in$ *CSR* and c.*Id* = C.*Id* and c.*Sche* $\prec_{Sche}$ C.*Sche*, c.*Pres* = C.*Pres* and c.*Sems* $\prec_{Sems}$ C.*Sems*;
2. If A $\in$ *AMBIGIOUS*, for each r $\in$ $\Pi_{Translatable}$(r, C), c = c $\cup$ r;
3. For each k $\in$ c, return k.A;

Advanced Content Query

Example: Determine the content inherited from the "Dublin Core" data model.

Query Statement: *Select C₁.Id From Content C₁, C₂ where C₂.Sche = "Dublin Core" and InheritFrom(C₁, C₂)*

Algorithm: AdvancedContentQuery(Attributes A, Contents C, Relationship R){
1. Locate content c, c $\in$ *CSR* and c.*Id* = C.*Id*, c.*Sche* $\prec_{Sche}$ C.*Sche*, c.*Pres* = C.*Pres* and c.*Sems* $\prec_{Sems}$ C.*Sems*;
2. If A $\in$ *AMBIGIOUS*, for each r $\in$ $\Pi^c_{Translatable}$(r, C), c = c $\cup$ r;
3. For each i $\in$ Π(R)
c = c $\cup$ ContentQuery(id, i)
4. For each k $\in$ c, return k.A;

Service query

A service query inquires about services in the query. A user can specify the query to be exact or ambiguous. An exact query returns services that entirely satisfy the query. An ambiguous query determines all services that "can be" the capabilities specified by the metadata in the query. "Can be" implies that the services can be concatenated or translated into the target service.

Basic service query

Example: Determine services with the service capability "Catalog_System".

Query Statement: *Select S.Id From Service S where S.Caps = "Catalog_System"*

Algorithm: ServiceQuery(Attributes A, Services S){
1. Locate service s, s $\in$ *CSR*, s.*Id*=s.*Id*, and s.*Caps* $\prec_{Caps}$ S.*Caps*;
2. If A $\in$ *AMBIGIOUS*, for each r $\in$ $\Pi^s_{Translatable}$(r, C.Sche), c = c $\cup$ r;
3. For each k $\in$ c, return k.A;

Advanced service query

Example: Determine services that has the same capabilities as a service whose capability is "Catalog_System".

Query Statement: *Select S₁.Id From Service S₁, S₂ where S₂.Caps = "Catalog_System" and Inclusive(S₁, S₂)*

Algorithm: AdvancedServiceQuery(Attributes A, Services S, Relationship R){
1. Locate service s, s $\in$ *CSR*, s.*Id*=S.*Id*, and s.*Caps* $\prec_{Caps}$ S.*Caps*;
2. If A $\in$ *AMBIGIOUS*,
Locate service rs $\in$ *CSR* where S.*Caps* $\prec^\sigma_{Caps}$ rs.*Caps*
c = c $\cup$ rs;
3. For each i $\in$ R
c = c $\cup$ ServiceQuery(id, i)
4. For each k $\in$ c, return k.A;

Ranking Function

A result of a CSIM semantic query can be classified as one of the following types:

1. **Exact match**. The result conforms to the query statement without additional translation, inheritance or combination.
2. **Ambiguous Match**. The result is a recommendation that may not completely satisfy the query statement, but can do so by translating, inheriting, or combining many available services or content.

A ranking function is proposed to evaluate the fitness of the results of the query, which may not totally fulfill the user's requirements. The ranking function *W* can be separated into $W_{Content}$ and $W_{Service}$. These definitions are as follows.

Ranking function $W_{Content}$(Service A, Service Results)
(Equation 1)
=**1**   if $A \prec_{Sches}$ Results and *Num*(Results)=1
=Π**(1-Ti)** if $A \prec^\sigma_{Sches}$ Results and *Num*(Results)>1

Ranking function $W_{Service}$(Service A, Service Results)
(Equation 2)
=**1**   if $A \prec_{Caps}$ Results and Num(Results)=1
=$(\Sigma W_{Results}^i * \Pi(1-Ti))/ (Num(Results)-Num(T_i))$
if A $\prec^\sigma_{Caps}$ Results and *Num*(Results)>1

where

$W_{Results}{}^i$: Num(Results^i.Caps)/Num(A.Caps) * $W^S_{Result}{}^i$
$W^S_{Result}{}^i$: User-specified service weight of Result^i
Ti : Overhead of translation rules in Results
Num(**Items**) : Total number of items

Ranking functions $W_{Content}$ and $W_{Service}$ are proposed to rank content and services, respectively. Rank follows the number of semantic concepts or capabilities that meet the criteria, and amount of content and service that are to yield the result.
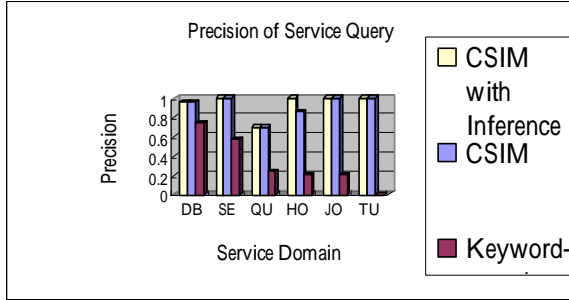
Experiments
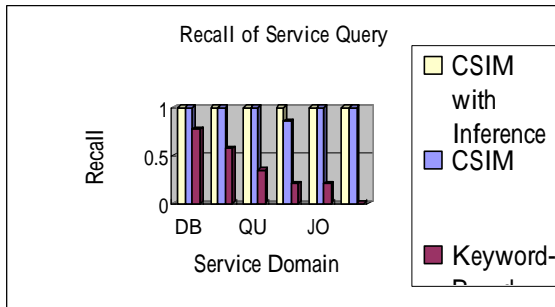

Figure 2. Precision of Service Query

Figure3. Recall of Service Query

Fig. 2 and 3 present the precision and recall of the service query. 0.5 is selected as the threshold $T_{service}$. Both figures indicate that the CSIM approach outperforms the keyword-based approach. In all service domains, the keyword-based approach was imprecise and had poor recall and precision when retrieving services, mainly because most services do not include specific keywords in their description. This finding explains why the systems that exploit concept approaches (which treat synonyms as a single attribute among schemas, as do in CSIM and other metadata approaches) responding to semantic DL queries. Notably, the CSIM approach with inference outperforms the pure CSIM in the holding domain because the former derives new integrated services from existing services. This result may encourage libraries to develop add-on services by integrating available services wherever reduce the develop cost.

Fig. 4 and 5 plot the precision and recall of the content query. 0.8 was selected as the threshold $T_{content}$ and 0.2 as the translation overhead. Setting 0.2 as the translation overhead allows only one translation in this experiment. Fig. 4 reveals that the recall of CSIM is greater than that of the keyword-based approach,

mainly because CSIM refers to CIT to obtain more conceptually-related attributes of schemas. For example, the "author" field in NCTUDL may exist in other university Webpack system with different name or connect with other field when keyword-based approach can not retrieve. Additionally, the CSIM approach with inference outperforms the pure CSIM approach, because translation services derive the content of more related data field. Fig. 5 shows that CSIM and CSIM with inference approaches outperformed the keyword-based approach.
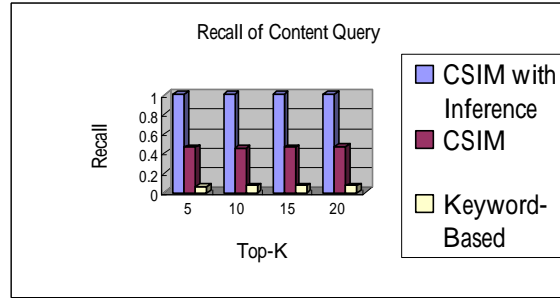
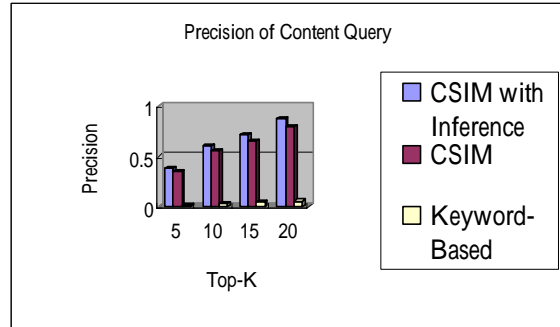

Figure 4. Recall of Content Query

Figure5. Precision of Content Query

In summary, the performance of the semantic DL query is highly promising. The CSIM model represents a dramatic improvement in responding to both service and content queries. The CSIM model not only enhances the recall and precision of a DL query but also helps how librarians should integrate available components into a desired service. Moreover, elaborated ontology and translation service alleviate the effort for librarians in developing union service like virtual union catalog system.

Prototype System

A prototype System called CSIM@NCTUDL has been implemented in our previous efforts (show as Fig. 6). This system supports advanced semantic DL queries for users to retrieve content and services in the NCTUDL. Moreover, using this system, librarians can be consulting to determine reusable components in the NCTUDL before new services are established. CSIM@NCTUDL includes four main areas.

1. **Selection area**: The selection area allows the user to specify which one of the three query types is to be issued - service query,

content query and compound query. Compound query performs complex service and content queries to retrieve content or service.

2. **Edit area**: The edit area allows the user to specify the query predicates, including input/output schema content semantics and service capability. Multiple capabilities can be selected from the ontology area.

3. **Ontology area**: The ontology area contains all ontological hierarchies. The ontology is developed by domain experts from all information in the NCTUDL.

4. **Result area**: The result area includes the results that satisfy the predicates in the edit area. If the result contains more than one item, this result recommends the users to combine these items together to obtain new content/service. The rank of each result is also given. A numeral in front of each service in the list of recommendations indicates the fitness of search criteria. The result area includes a set of lists to advice librarians; nevertheless, the system leaves the task of confirming the feasibility of the list of recommendations to the users.
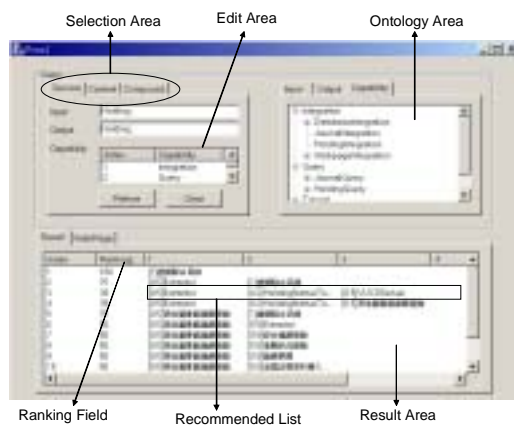

Figure 8 CSIM@NCTU

Conclusions and Future Work

Information overload incurs problems in retrieving content and services from a digital library query. Conventional keyword-based searches are ineffective. This work presents a novel content and service inference model (CSIM), which categorizes 15 relationships among content and service to handle semantic DL queries. This work enumerates these relationships and realizes them as manipulating functions, $\prec$ *and* Π *operations*. $\prec$ *operations* return TRUE or FALSE for a specific relationship and Π *operations* determine all content or service that exhibit a specific relationship. The proposed semantic DL query applies CSIM and consist of two types of queries. An exact query returns the answers that match the predicate, and an ambiguous query yields recommends answers that can be inherited, translated or combined from available content or services, as well as those answers that match the corresponding exact query.

We have also applied CSIM in the digital library of National Chiao Tung University (NCTUDL) and a Virtual Union Catalog System (VUCS@NCTUDL) and CSIM@NCTUDL for service consultant were constructed. Experimental results showed that the CSIM outperformed the conventional keyword-based approach to handling DL queries. An ambiguous DL query with CSIM recommends additional results beyond those of the conventional keyword-based approach, improving the recall and precision of both content and service retrieval. Applying CSIM to digital library queries has been shown to reduce the administrative load when constructing new service and content. Digital library designers can generate new content and services from those available, by considering the recommended results of an ambiguous query. Using metadata, translation rules, services and content can be translated and reused, to the benefit of an object-oriented or component-based digital library design.

CSIM can be applied to many DL applications. For a DL resource-planning system, the inference capability of CSIM suggests that librarians should reuse available components to easily generate new DL services. Furthermore, CSIM can assist librarians to manage versatile data semantics using ontological tables. Various data fields can be categorized into semantic hierarchies to simplify data transformation among them. This reduction of ambiguity facilitates the combination of different data fields. Such combination frequently occurs when union systems are created. The experiments presented here have demonstrated that applying CSIM in a virtual union catalog system can improve performance over that of the conventional keyword-based approach. Moreover, applying CSIM to a DL user interface improves the performance of the semantic DL query. In most digital libraries, services are distributed on the Web, making the system map to sloppy and large. Storing information as metadata and applying a semantic search of DL services makes such searches simple and easy. Users can designate a service's input type (like library holdings), output type (like web pages) and capabilities (like search systems). This semantic assignment and search bridges the gap between the desires of the user and service capabilities.

Future research will focus on accommodating broader range of semantics and developing new schemes to obtain more knowledge from metadata that contains abundant semantics can facilitate digital library queries. In addition to investigating the optimization of operations in CSIM, our future works will design more efficient indexing mechanisms to access related data structures.

[1] Su-Hsien Huang, Hao-Ren Ke and Wei-Pang Yang, "Enhancing Semantic Digital Library Query by a Content and Service Inference Model (CSIM)", submitted to Information & Processing Management.

[1] Arms, W. Y. (1995) Key concepts in the architecture of the digital library, *D-Lib Magazine*, http://www.dlib.org/.

[2] Baldonado, M. & Chang, C. C. K. & Gravano, L. & Paepcke, A. (1997) The stanford digital library metadata architecture, *International Journal on Digital Libraries*, 1(2), 108-121.

[3] Berners-Lee, T. (1997) Metadata architecture - documents, metadata, and links, http://www.w3.org/DesignIssues/Metadata/.

[4] Derbyshire, D. & Ferguson, I. A. & Muller, J. P. & Pischel, M. & Wooldridge, M. (1997) Agent-based digital libraries: driving the information economy", enabling technologies: infrastructure for collaborative enterprises, *Proceedings of the 6th Workshop on Enabling Technologies Infrastructure for Collaborative Enterprises.*

[5] Digital Library Production System - DLPS (1999) Supporting access to diverse and distributed finding aids: A final report to the digital library federation on the distributed finding aid server project, http://www.umdl.umich.edu/dlps/dfas/dfas-final.html.

[6] Dublin Core metadata initiative (1999), http://purl.oclc.org/dc/.

[7] Grossman, R. & Qin, X. & Xu, W. (1995) An architecture for a scalable, high-Performance digital library, *Mass Storage Systems, Proceedings of the Fourteenth IEEE Symposium.*

[8] Hu, M. J. & Jian, Y. (1999) Multimedia Description Framework (MDF) for content description of audio/video documents, *Proceedings of the Fourth ACM International Conference on Digital Libraries.*

[9] Huang, S. S. & Ke, H. R. & Yang, W. P. (2000). Interoperability of cooperative databases with metadata, *The Fourth World Multiconference on Systemics, Cybernetics and Informatics*. Orlando U.S.A.

[10] Jenkins, C. & Jackson, M. & Burden, P. & Wallis, J. (1999) Automatic RDF metadata generation for resource discovery, *Computer Networks.*

[11] Kahn, R. & Wilensky, R. (1995) A framework for distributed digital object services, http://www.cnri.reston.va.us/cstr/arch/k-w.html.

[12] Ke, H. R. & Huang, S. S. & Yang, W. P. (2001). The study of interoperability of digital libraries with metadata. *University Library Journal*, 5(1), 49-78.

[13] Lagoze, C. (1996) The Warwick framework, *D-Lib Magazine*, http://www.dlib.org/.

[14] Lynch, C. & Garcia-Molina, H. (1995) Inteoperability, scaling and the digital libraries research agenda, *Informaiton Infrastructure Technology and Applications (IITA) Digital Libraries workshop.*

[15] McCray, A. T. & Gallagher, M. E. & Flannick, M. A. (1999) Extending the role of metadata in a digital library system, *Research and Technology Advances in Digital Libraries IEEE* (pp. 190-199).

[16] Melnik, S. & Garcia-Molina, H. & Paepcke, A. (2000) A mediation infrastructure for digital library services, *Proceedings of the Fifth ACM International Conference on Digital Libraries*, (pp. 123-132), San Antonio, U.S.A.

[17] Monch, C. & Drobnik, O. (1998). Integrating new document types into digital library, *IEEE Research and Technology Advances in Digital Libraries .*

[18] Nikolaou, C. & Marazakis, M. (1999) System infrastructure for digital libraries: A survey and outlook, International Conference of Software SEMinar, SOFSEM99, (pp. 186-203), Slovakia.

[19] Paepcke, A. & Chang, C. C. K. & Garcia-Molina, H. & Winograd, T. (1998) Interoperability for digital libraries wroldwide, *ACM Communication*, 41(4).

[20] Paepcke, A & Cousins, S. B. & Garcia-Molina, H. & Hassan, S. W. & Ketchpel, S. P. & Roscheisen, M. & Winograd, T (1996) Using distributed objects for digital library interoperability, 29(5), *IEEE Computer.*

[21] Resource Description Framework (RDF) Schema Specification (1999), *W3c Recommendation*, http://www.w3.org/TR/PR-rdf-schema/.

[22] Roscheisen, M. & Baldonado, M. & Chang, K. & Gravano, L. & Ketchpel, S. & Paepcke, A. (1997) The stanford InfoBus and its service layers, http://www-diglib.stanford.edu.

[23] Tim, B. (1997) Metadata architecture - documents, metadata, and links, http://www.w3.org/DesignIssues/Metadata.

[24] Weinstein, P. C. & Birmingham, W. P. & Durfee, E. F. (1999) Agent-based digital libraries: decentralization and coordination, *IEEE Communications Magazine.*