DelosDLMS: Infrastructure and Services for Future Digital Library Systems

Ceri Binding[6], Gert Brettlecker[1], Tiziana Catarci[3], Stavros Christodoulakis[4], Tom Crecelius[7], Nektarios Gioldasis[4], Hans-Christian Jetter[5], Mouna Kacimi[7], Diego Milano[1,2], Paola Ranaldi[1,2], Harald Reiterer[5], Giuseppe Santucci[3], Hans-Jörg Schek[1,2], Heiko Schuldt[1], Douglas Tudhope[6], Gerhard Weikum[7]

1 Database and Information Systems Group, University of Basel, Switzerland
2 Database and Information Systems Group, University of Konstanz, Germany
3 Department of Information Systems, Università degli Studi di Roma "La Sapienza", Italy
4 Laboratory of Distributed Multimedia Information Systems & Applications, TU Crete, Greece
5 Human-Computer Interaction Group, University of Konstanz, Germany
6 Hypermedia Research Unit, University of Glamorgan, United Kingdom
7 Databases and Information Systems, Max-Planck-Institut für Informatik Saarbrücken, Germany

Abstract

DelosDLMS is a prototype of a next-generation Digital Library (DL) management system. It is the result of integrating various specialized DL services provided by partners of the DELOS network of excellence into the OSIRIS platform. OSIRIS is a middleware environment for the reliable and scalable distributed execution of processes. Processes, in turn, are DL applications that are built from the specialized services available in the integrated system. DelosDLMS provides support for content-based retrieval in image, audio, video, and 3D collections and combination of any of these media types with keyword queries. It allows annotating retrieved information, provides a rich set of advanced graphical user interfaces to browse and explore large collections, and supports users in interacting with the system using a speech interface and interactive paper. Finally, it can even be applied in settings where DL content is available in a distributed P2P system.

Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.1 Content Analysis and Indexing, H.3.2 Information Storage, H.3.3 Information Search and Retrieval, H.3.4 Systems and Software, H.3.7 Digital Libraries

General Terms

Algorithms, Performance, Reliability

Keywords

Digital Library Management System, System Integration, Service-oriented Architecture

1 Introduction

DelosDLMS is a prototype implementation of a next-generation Digital Library management system [Schek and Schuldt 2006]. DelosDLMS provides functionality in a single system that is not available in any known system so far. It combines text and audio-visual searching, offers personalized browsing using new information visualization and relevance feedback tools, provides novel interfaces, allows retrieved information to be annotated and processed, integrates and processes sensor data streams. The system is built over OSIRIS, a middleware environment initially developed at ETH Zurich and then expanded and maintained at the University of Basel. OSIRIS allows building process-based DL applications starting from services (and already existing processes alike), and executing them in a distributed P2P fashion. It acts as glue which is able to

compose specialized DL services into full-fledged DL applications, and provides a highly reliable and scalable distributed execution environment for the resulting applications. DelosDLMS features a rich library of DL services including ISIS, an existing set of services for multimedia search developed at the ETH Zurich and later maintained at the University of Basel, and several services provided by partners in the DELOS network of excellence.

The philosophy behind DelosDLMS is that DL applications can be easily built starting from specialized DL services produced independently from each other. The goal of the DELOS integration activities is to integrate DL services developed by DELOS partners into DelosDLMS, and use them as building blocks for DL applications that showcase the capabilities of the system. Some services and applications have been already integrated into DelosDLMS in previous integration phases (see [Agosti et al 2007]). This paper reports on the latest integration results, describing in detail all services that have been added to DelosDLMS during the latest integration phases. They are the MedioVis user interface (University of Konstanz), the STAR SKOS services (University of Glamorgan) for thesaurus browsing and query expansion, the Minerva P2P search engine (MPI Saarbrücken), the DARE visualization interface (University of Rome), and the CoCoMA/OntoNL services (Technical University of Crete) for semantic multimedia annotations management and Natural Language understanding.

The paper is organized as follows. Section 2 provides an overview of DelosDLMS, describing both the middleware layer over which DelosDLMS is built and its library of DL services. Section 3 presents in detail the services and DL functionality that have been added and architectural details on the integration. Section 4 concludes.

2 Overview of DelosDLMS

DelosDLMS is a prototype for future Digital Library Management Systems (DLMS) and a showcase of DL applications, produced by the joint work of partners of the DELOS Network of Excellence. It is a partial implementation of the reference model for digital libraries produced by the DELOS NoE [Candela et al 2007]. The philosophy behind DelosDLMS is to allow fast and easy development and flexible management of complex DL applications by relying on a management system for Service Oriented Architectures and on a rich, extensible library of specialized DL services and interfaces. These components are used as building blocks for full-fledged DL applications, with the SOA middleware providing the necessary “glue” between them.

2.1 OSIRIS

DelosDLMS uses as its basic layer OSIRIS (Open Service Infrastructure for Reliable and Integrated Process Support) [Schuler et al. 2006], a middleware environment for P2P process execution. OSIRIS allows combining distributed services into processes, and executing them in a distributed way. Process design is possible without the need to care about many details regarding services, including their physical location and availability. Processes, in turn, are wrapped by a service interface, and can be started and invoked as other services. Process invocation abstracts from specific service instances and frees the user or application from having to deal directly with failure handling. These features enable true “programming in the large”.

Compared to other service composition and process execution environments, OSIRIS has some key aspects that make it extremely reliable and scalable. Processes in OSIRIS follow the model of transactional processes [Schuldt et al. 2002]. Starting from knowledge of the transactional behaviour of the service invocations that constitute the execution of a process, including information on compensation (how can the effects of a service execution be semantically undone) and on whether a failed service can be re-invoked (retriability), transactional guarantees on the process executions themselves are provided. Furthermore, process execution in OSIRIS is managed in a decentralized P2P fashion, without a centralized execution engine/scheduler. This avoids any single point of failure during process execution. OSIRIS also incorporates sophisticated load-balancing mechanisms in order to distribute load among available peers. Among the features

introduced in OSIRIS at the University of Basel, a notable one is the support for integration and processing of continuous streams of data, e.g. sensor data.

All these features are supported by a set of internal *System Services*, used for communication and to coordinate distribute process execution and load management. These services constitute an OSIRIS layer, which is run at any node in an OSIRIS managed network. The integration of new *Application Services*, those used to actually build applications and processes, into OSIRIS can be done by deploying them over an OSIRIS layer running at some node in a network of peers. Services deployed in this way are called *tightly coupled services*. Tightly coupled services are under the full control of the OSIRIS Layer, and for them it is thus possible to exploit the full power of the OSIRIS architecture. *Loosely coupled services* are instead deployed on nodes without a local OSIRIS layer. These services can be remotely invoked from any OSIRIS node (via a specific system service), and transparently used as the other services, except for the fact that the system cannot start them or shut them down.

OSIRIS comes with an integrated graphical administration tool, called O'Grape (OSIRIS Graphical Process Editor), which allows the design of new processes through a boxes and arrows approach. The intuitive interface of O'Grape makes application building in OSIRIS easy and fast.

OSIRIS exists in two versions, a C++ version for the Microsoft Windows platform, whose development began at ETH Zurich, and a cross-platform Java version, developed at the University of Basel. The O'Grape tool is also written in Java.

2.2 ISIS and additional DL services in DelosDLMS

On top of OSIRIS, DelosDLMS features a wide and ever growing collection of DL services and front-ends. The core of the service layer is ISIS (Interactive SImilarity Search) [Mlivoncic et al. 2004], a set of Digital Library services that support content-based retrieval in multimedia collections. Similar to OSIRIS, ISIS has also been initially developed at the ETH Zurich and is currently maintained at the University of Basel as part of the DelosDLMS project. Notable features of ISIS are the capability of mixing textual (metadata-based) and content-based queries, a high-performance indexing and retrieval service based on the VA file data structure [Weber et al. 1998] enabling multi-feature indexing over multimedia collections, and relevance feedback tools for iterative refinement of query results. The indexing engine of ISIS is feature-neutral, which means that new features (and new media types) can be added to DelosDLMS by just plugging in feature-extraction services, without need to change the indexing service. ISIS also comes with a graphical user-interface for querying collections and browsing results.

Beside the ISIS library, a number of specialized DL services produced over time by DELOS partners have been integrated in various phases inside DelosDLMS. These services add new functionalities to those already available in ISIS. They include:

- **Feature-extraction services** for content-based search on several kind of media, namely audio files (TU Vienna), 3D objects (Univ. of Firenze), and video (Univ. of Firenze, Tech. Univ. of Crete);
- **Annotation services.** The FAST service (Univ. of Padua) allows retrieved information to be annotated. Annotations are stored persistently, and can be later queried together with other textual metadata;
- **User-centric interfaces and services.** These components are designed to enhance the user experience in querying DL content by allowing multi-collection queries, integrated web searching starting from query results, personalized browsing. Daffodil (Univ. of Duisburg-Essen) is a system for management and browsing of multiple collections. It has been integrated with DelosDLMS, adding content-based search to its original search capabilities. MedioVis (Univ. of Konstanz) is a graphical user interface that provides the user with multiple, highly customizable zoomable views over query results. The STAR SKOS services allow search and browsing access to thesauri in the W3C SKOS format. Building on this, it is possible to provide the user with query expansion services to improve search over annotated collections;

- **Alternative user interfaces**. These interfaces allow non standard interaction with DL functionalities. iPaper (ETH Zürich), is a paper-based interface, that can be used to access the content-based query service offered by DelosDLMS, and it is particularly suitable for use in scenarios such as museums. Also available is a natural language voice interface from the CoCoMA/OntoNL DELOS task that is able to analyze natural language sentences and formulate appropriate queries to the system (Tech. Univ. of Crete);
- **Advanced visualization services**. These services provide advanced graphical means to explore and analyze large query results and entire collections. They are thought to help advanced users and DL system administrators. Among them, a Self Organizing Map (SOM) visualization of the feature space (Univ. Konstanz), that helps visualizing a collection in terms of content-based similarity clusters and to locate query objects and results inside the collection, and the DARE system (Univ. Roma), that provides an interactive 3D view of large high dimensional datasets with structured metadata;
- **Large-scale P2P search capabilities.** Minerva (MPI Saarbrücken) is a P2P search engine designed to support efficient textual queries in large P2P networks by selecting the most appropriate peers to answer a query. The node selection capabilities of Minerva have been made available in DelosDLMS to allow for textual and content based search over collections distributed in large scale systems.

Some of the services mentioned above (namely, the feature extraction services for audio and 3D objects, the FAST annotation service, the SOM visualization and the Daffodil user interface) have been integrated into DelosDLMS during the first integration phase, and are described in detail in [Agosti et al 2007]. In the next sections of this paper we describe the services that have been added recently to the system.

The features and capabilities of DelosDLMS are showcased through a variety of DL applications built from services and interfaces on top of a set of large multimedia collections. These applications show the potential of DelosDLMS, allow for multimedia, multi-object metadata and content-based searching with relevance feedback over image, audio, video and 3D objects collections, through standard and non-standard interfaces and with several alternative visualizations of query results.

3 Integration Activities

3.1 MedioVis

MedioVis is a visual information seeking system developed by the Human-Computer Interaction Group at the Department of Computer and Information Science of the University of Konstanz. MedioVis demonstrates novel ways of information seeking in digital libraries for novice and non-expert users by integrating novel interaction and visualization techniques in a consistent user interface design. The project is funded by the Deutsche Forschungsgemeinschaft (DFG) to provide libraries in academic and scientific institutions with a highly usable and versatile visual user interface as an open source alternative to traditional OPAC systems [MedioVis 2007]. As a test case MedioVis has been introduced into the Library of the University of Konstanz in 2003 where it has been in use since then and where its superior user experience has been empirically researched [Grün et al. 2005].

A central design goal of the MedioVis interface is to offer interaction design that supports realistic human search behaviour. Therefore not only analytical queries but also browsing-oriented and opportunistic retrieval strategies are supported, for example by allowing visual overviews, visual filtering, zooming into details, and "surfing" between the search results' metadata and full text. This has lead towards a user experience which Hearst et al. have formulated as a design goal for search systems: "a browsing the shelves sensation for large collections of information items" [Hearst et al. 2002]. This experience is also based on the rich metadata and full text in MedioVis harvested from web sites and web services to provide the user with additional information ranging

from biographies to geographic maps or video files. Furthermore the integration of MedioVis into the Delos DLMS system offers the user new feature-based similarity search options, e.g. to find paintings or portraits similar to the ones the user has selected. This furthermore supports natural information seeking strategies.

Visually handling this amount of metadata and content is possible by using novel zoomable user interface components and visualizations (e.g. HyperGrid [Jetter et al. 2005] or HyperScatter) which allow a direct manipulation of the displayed level of detail and therefore gradually blur the boundaries between metadata, full text and external data sources.

Further visualizations like social networks or bar grams unveil tacit relations between qualitative and quantitative data as has been successfully demonstrated in the IEEE InfoVis Contest 2007 with the award-winning “Blockbuster” system based on MedioVis [Rexhausen et al. 2007].

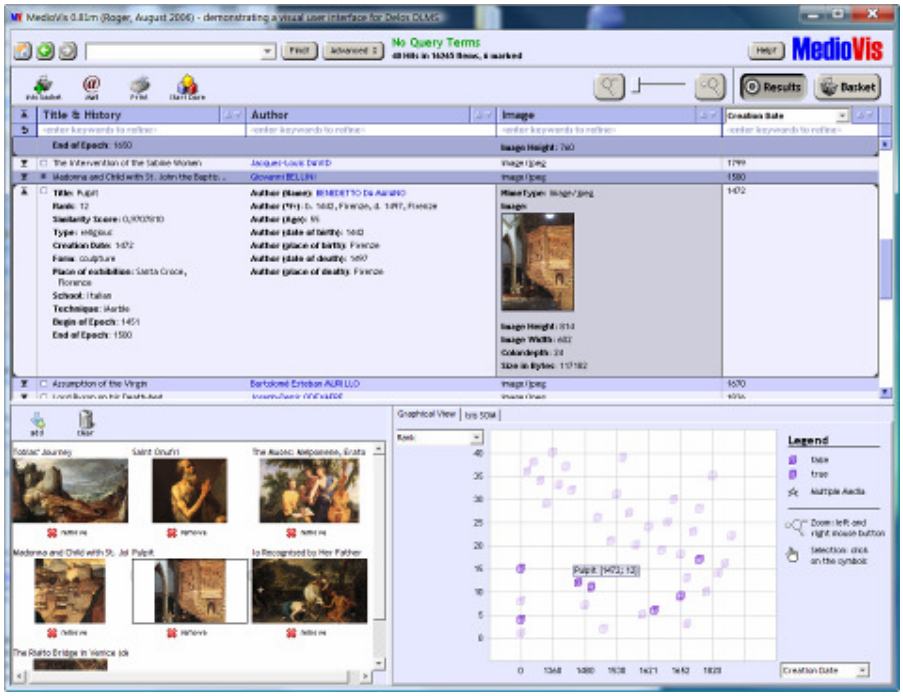

Figure 1: MedioVis with DelosDLMS similarity search

3.2 DARE

The design of an information system requires a friendly interface that allows the final user to easily access the data of interest. Very often, such an interface exploits the power of visualization and direct manipulation mechanisms. However, it is not sufficient to associate ``any'' visual representation to a database, but the visual representation should be carefully chosen to effectively convey all and only the database information content. This issue is addressed by DARE: Drawing Adequate Representations [Catarci and Santucci 2001], a system implementing a general theory [Catarci et al. 1999] for establishing the adequacy of a visual representation, once specified the database characteristics. The DARE system is based on a knowledge base containing different kinds of rules, namely:

- Visual rules, that characterize the different kinds of visual symbols;
- Data rules, that specify the characteristics of the data model, the database schema, and the database instances;
- Mapping rules, that specify the link between data and visual elements;

- Perceptual rules that tell us how the user perceives a visual symbol (e.g., a line), relationships between symbols (e.g., the mutual placements of two figures), and which is the perceptual effect of relevant visual attributes such as color, texture, etc.

The actual DARE prototype has been implemented as a general purpose application, able to automatically adapt to a large variety of domains; moreover, a visual OLAP engine has been added to the prototype, allowing the end user to easily aggregate the data in order to get a more abstract view of the dataset. Such a feature has been successfully exploited to deal with the 2005 Infovis Contest [Bertini et al. 2005].

A set of basic DARE services has been implemented for the DelosDLMS Infrastructure and DARE has been used to manipulate the data outcoming from the Mediovis environment described in the previous Section (see Figure 2) and to handle the metadata coming from the 3D searching engine developed at the University of Konstanz [Bustos et al. 2007], allowing for a better understanding of feature vectors behaviours (see Figure 3).

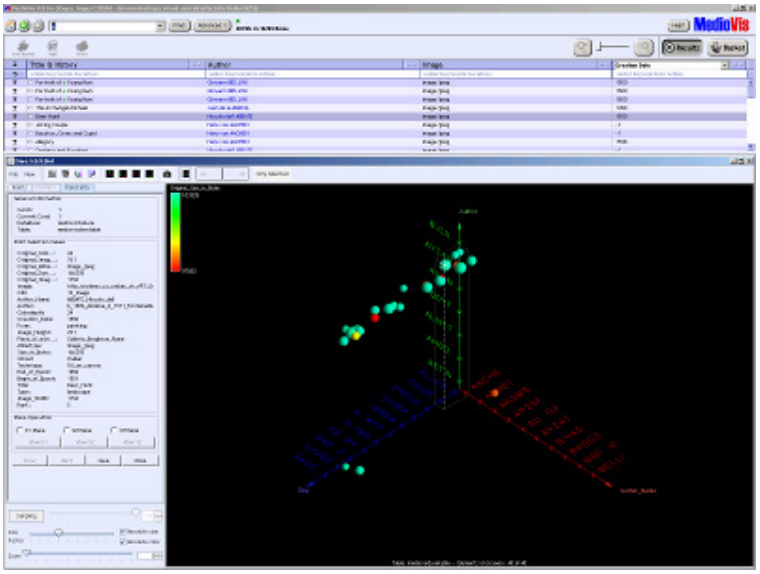

Figure 2: The integration of Dare into Mediovis

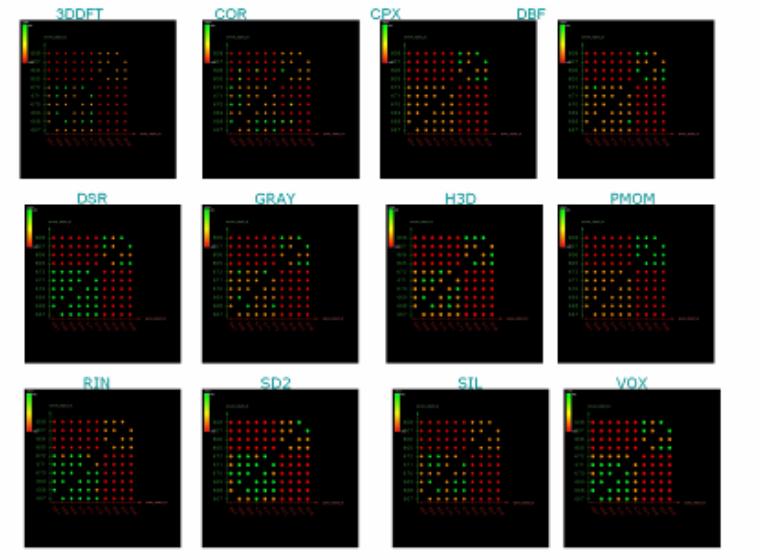

Figure 3: Comparing 12 feature vector performances for the 3D Konstanz search engine

3.3 CoCoMA and OntoNL

The CoCoMA Delos task aims to provide users of digital library systems with a solution for intelligent personalized retrieval from large media collections. Among others, the CoCoMA task has focused on the representation and management of semantic multimedia annotations and indexing. The DSMIRF framework has been developed to allow standard multimedia metadata schemas (MPEG-7) to be enhanced with domain knowledge. The DS-MIRF [Tsinaraki et al. 2007] framework allows the integration of MPEG-7 and OWL-encoded ontologies in order to allow domain specific semantics to be captured. Furthermore, CoCoMA has developed a MPEG-7 oriented Query Language (MP7QL) that allows the unified retrieval of multimedia content based on both semantic and content based criteria. MP7QL has been implemented as a multimedia retrieval service over the CoCoMA MPEG-7 Multimedia Metadata Repository. For fine-grained multimedia content delivery, CoCoMA has developed a multimedia segmentation service which extracts in real time video segments that correspond to particular result items identified by the multimedia retrieval service. Thus, a complete multimedia retrieval scenario (search, selection, real-time content segmentation and delivery) can be supported.

The Delos OntoNL task focuses on the development of a natural language interface generator framework [Karanastasi et al. 2006] that will automate the development of natural language interfaces on top of knowledge repositories. As a result, a reusable service (OntoNL service) has been developed which can automatically generate NL interfaces for knowledge repositories based only on the domain knowledge provided as an OWL ontology.

CoCoMA and OntoNL services have been integrated into Delos DLMS as a complete sub-system that allows digital library users to search for audiovisual content by posing semantic-based criteria through a speech interface. This sub-system is a proof-of-concept that both DSMIF (MPEG-7 enhanced with domain knowledge) and OntoNL (Natural Language understanding based on domain knowledge and automatic interface generation) frameworks can be valuable in large scale audiovisual digital library systems allowing users to express complex queries in their natural language and in their own knowledge domain.

Figure 4 shows the multi-layered service-oriented architecture of this sub-system. The three defined layers are:

- **User Interface Layer** which accommodates components used to allow end users to submit knowledge acquisition requests in Digital Libraries. Requests can either be spoken and recognized by a Speech Recognition Component (SRC), or typed directly in a Natural Language style through a specific Graphical User Interface (GUI).
- **Natural Language Processing Layer.** This layer contains the Natural Language Processing Service (OntoNL Service) which disambiguates the NL requests based on the ontologies of the application domain (i.e. sports) and generates one or more structured knowledge acquisition requests encoded in SPARQL. These structured requests can further be transformed in other query languages (SQL, XQuery, MP7QL) based on the interface of the underlying knowledge repository.
- **Digital Library Layer.** The Digital Library Layer represents knowledge and knowledge manipulation services provided by a Digital Library Management Systems. In the proposed architecture, the DL layer is materialized by the CoCoMA MPEG-7 Metadata Repository and the services that are offered on top of this.

The integration points of this architecture into Delos DLMS are illustrated as dotted rounded rectangles in Figure 4. That is, DLMS has undertaken the integration and coordination of the various services used in the sub-system by defining, managing, and executing the related process. The proposed approach is general enough to be used with any knowledge (metadata) repository in the Digital Library Layer. The key idea behind is that a separate layer of natural language processing can be deployed on top of existing digital libraries to provide effective retrieval interfaces.

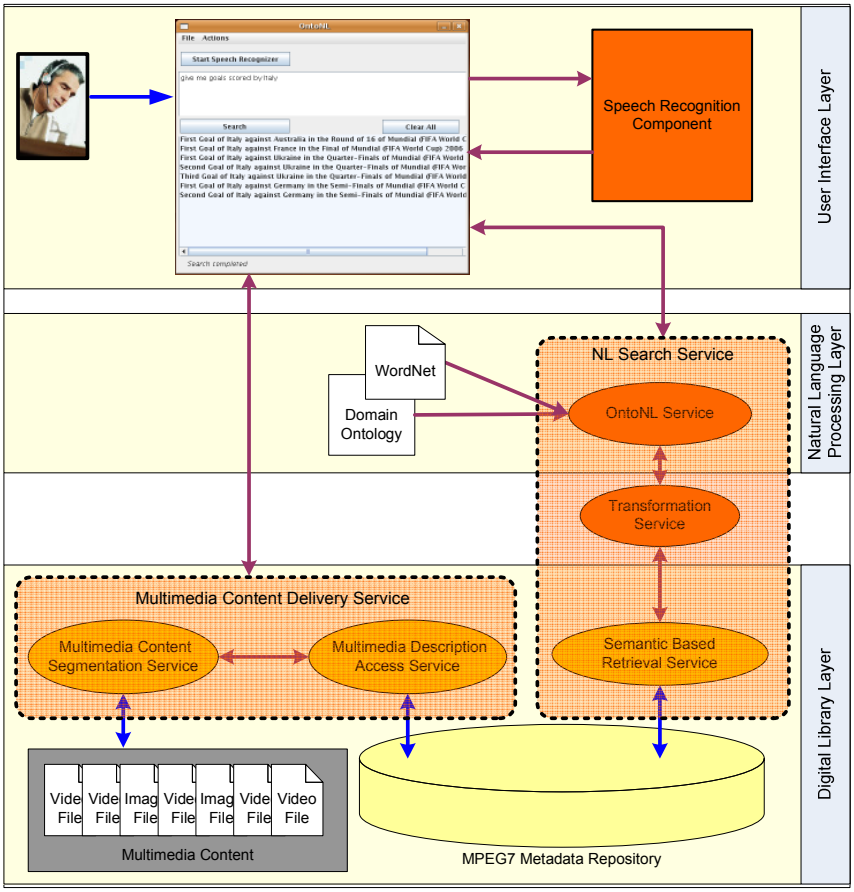

Figure 4: Architecture of the CoCoMA and OntoNL Integration into Delos DLMS

3.4 STAR SKOS Services

SKOS_WS is a SOAP web service for vocabularies represented in SKOS Core Vocabulary [SKOS 2004]. It was developed as part of the AHRC funded STAR project [STAR], building on previous work by the Hypermedia Research Unit on terminology web services [Binding and Tudhope 2004][Tudhope and Binding 2006]. The service is written in C#, running on Microsoft .NET framework (version v2.0.50727) and is based on a subset of the SWAD Europe SKOS API (see [SKOS 2004]), with extensions for concept expansion.

The service works with thesauri or related Knowledge Organization Systems (KOS) represented in SKOS format. SKOS is a formal RDF/XML representation standard for the large family of vocabularies and concept structures, with an informal semantics designed for information retrieval purposes. This offers a lightweight, cost effective approach for annotation, search and browsing oriented applications that don't require first order logic.

At the time of writing, the service accesses and searches across 6 separate thesauri, which were converted to SKOS RDF in the initial phases of the STAR project:

- English Heritage Archaeological Sciences Thesaurus
- English Heritage Evidence Thesaurus
- English Heritage Building Materials Thesaurus
- English Heritage Monument Type Thesaurus
- MDA Object Type Thesaurus
- Alexandria Digital Library Feature Type Thesaurus

The service consists of 8 function calls (see Table 1), which can be integrated into a textual or metadata based search system. Functionality includes a facility to look up a user provided string in the controlled vocabularies of all KOS known to the server, returning all possibly matching concepts. The ability to browse concepts via the semantic relationships in a thesaurus is provided. Semantic expansion of concepts for purposes of query expansion is also possible; (configurable) automatic traversal of SKOS relationships yields a ranked list of semantically close concepts. More details of semantic expansion are given in [Tudhope and Binding 2006].

The services allow search to be augmented by KOS-based vocabulary and semantic resources. Users may browse a concept space to explore and become familiar with specialist terminology or may browse to directly access data linked to concepts. Queries may be expanded by synonyms or by semantically related concepts. For example, a query is often expressed at a different level of generalisation from document content or metadata, or a query may employ semantically related concepts. This provides an augmented textual search capability to complement existing OSIRIS content-based retrieval.

GetTopmostConcepts(uri : String) : Concept[] Get the Concepts at the 'top' of the hierarchies for the specified ConceptScheme - returns an array of Concept.
GetConceptScheme(uri : String) : ConceptScheme Get the specified ConceptScheme. Returns a single ConceptScheme object.
GetConceptSchemes() : ConceptScheme[] Get all ConceptSchemes supported by the service. Returns an array of ConceptScheme objects.
GetConcept(uri : String) : Concept[] Get the specified Concept. Returns a single Concept object.
GetAllConceptRelatives(uri :String) : ConceptRelative[] Get all Concepts directly related to the specified Concept. Returns an array of ConceptRelative objects.
GetConceptsMatchingKeyword(keyword : String, includeNPT : Boolean, matchType : MatchTypeEnum) : Concept[] Find lexical term matches across all supported ConceptSchemes. There is an option to include non-preferred terms in the search, and the type of match to look for (exact match, starts with, or contains). Returns an array of Concept objects.
ExpandConceptSimple(uri : String) : ConceptRelative[] A simple Concept expansion function, using internally fixed relationship traversal cost parameters. Returns an array of ConceptRelative objects.
ExpandConcept(uri : String, costBT : Double, costNT : Double, costRT : Double) : ConceptRelative[] A more configurable Concept expansion function, allows specification of traversal costs for the core relationship types. Returns an array of ConceptRelative objects.

Table 1 - SKOS_WS function calls

3.5 MINERVA

Minerva [Bender at al. 2005], developed at MPI Saarbrücken, is a fully operational P2P search engine combining local index structures of autonomous peers with a global directory based on a distributed hash table as an overlay network. Each peer has a local data collection that can be built from the peer's own contents or Web crawls, or imported from external sources related to the user profile. The data collection of a peer is locally indexed using inverted lists, where for each key (e.g., keyword or term) is associated the list of the corresponding URLs of Web pages or other files. Minerva maintains a metadata directory that holds very compact, aggregated summaries of the peers' local indexes. The directory implementation is based on Past [Rowstron and Druschel 2001], a freely available implementation of a distributed hash table (DHT). It uses FreePastry's

route primitive to support the two hash table functionalities (insert (key,value) and retrieve(key)). Minerva passes (term,synopsis)-tuples to Past, which transparently stores it at the node in the network that is currently responsible for the key term. A query initiator selects a few most promising peers based on their published per-term summaries, e.g., by executing a distributed top-k algorithm [Cao and Wang 2004], [Michel at al. 2005]. Subsequently, it forwards the complete query to the selected peers which execute the query locally. This query execution does not involve a distributed top-k query execution since each peer maintains a full-fledged local index with all information necessary to execute the query locally. Finally, the results from the various peers are combined at the querying peer into a single result list.

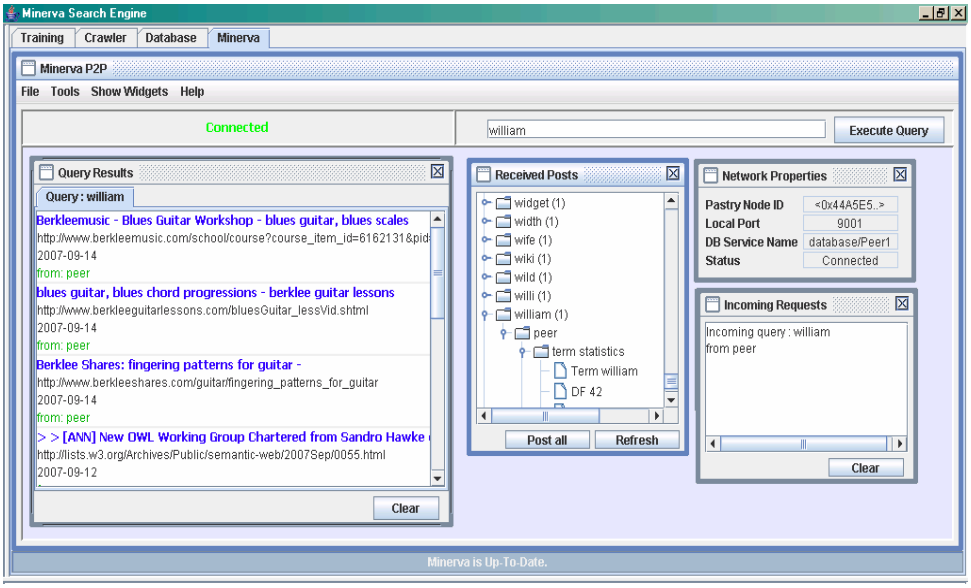

Figure 5: Minerva Search Interface

Minerva is released as open source and available under **http://www.minerva-project.org**. As shown in Figure 5, Minerva offers a simple search interface, that allows a user to enter query terms, which (transparently to the user) starts the global query execution: for each term appearing in the query, Minerva retrieves all applicable synopses from the directory, which serve as the input to query routing, i.e., selecting a small subset of promising peers that are most likely to provide high-quality results for a particular query. Minerva uses FreePastry's route primitive to send the user query to these selected peers, which evaluate the query using their local TopX [Theobald et al. 2005] engines on top of their local indexes and return their top-matching URLs to the query initiator. Minerva appropriately combines the URLs from these autonomous sources (result merging) and displays the results to the user. After inspecting the results, the user can simply click on any result to open the document in the default Web browser.

3.6 Summary of Integration Activities

The DelosDLMS system that has been produced as result of the DELOS integration activities is depicted in Figure 6. In addition to audio and 3D features which have been added to the VA-file index of DelosDLMS, also the FAST annotation management, the SOM visualization, the iPaper front-end and the Daffodil user interface have been integrated in the first phase. During the second phase, DelosDLMS has been enriched with additional video features. Keyword queries can receive valuable extension due to the STAR SKOS query expansion service. Moreover, the COCOMA services allow for natural language queries and provide a speech interface to DelosDLMS. With MedioVis and DARE, it is now also able to visualize large collections with structured metadata. Finally, the integration of the MINERVA P2P search engine allows for the execution of textual and content based queries over collections distributed in large scale systems.

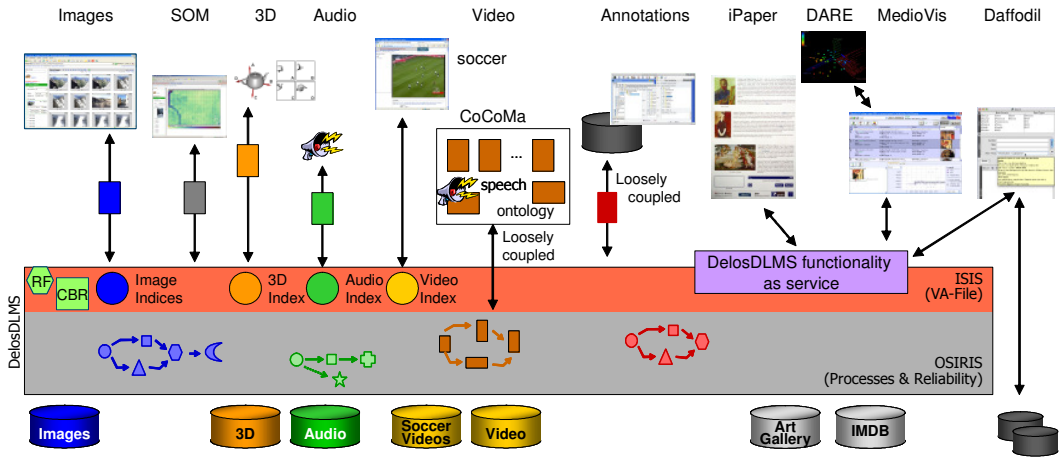

Figure 6: DelosDLMS Overview

4 Conclusion

In this paper, we have presented DelosDLMS, a joint effort of partners in the DELOS Network of Excellence to develop a prototype for a next generation Digital Library Management System that combines a rich set of features that is not available in this combination in any existing system. DelosDLMS has been developed in two phases. The results of the first phase have been presented in [Agosti et al 2007]. In this paper, we have concentrated on the integration activities of the second phase. It has to be noted that DelosDLMS is still an ongoing effort and will be further enriched with additional functionality. Current activities consider the integration of personalization services from the University of Athens and the extension of the OSIRIS middleware which will finally allow making use of DelosDLMS and its services on mobile devices.

References

M. Agosti, S. Berretti, G. Brettlecker, A. del Bimbo, N. Ferro, N. Fuhr, D. Keim, C.-P. Klas, T. Lidy, D. Milano, M. Norrie, P. Ranaldi, A. Rauber, H-J. Schek, T. Schreck, H. Schuldt, B. Signer, M. Springmann 2007. DelosDLMS - the Integrated DELOS Digital Library Management System. In: Post-Proceedings of the 1st DELOS Conference on Digital
Libraries, Tirrenia, Italy. Springer LNCS.

M. Bender, S. Michel, P. Triantafillou, and G.Weikum 2005. Minerva: Collaborative p2p search. In VLDB.

E. Bertini, L. Dell'Aquila, G.Santucci 2005. Discovering USA companies insights using DARE and SpringView - honorable mention of the Infovis contest 2005, Proceedings del IEEE Symposium on Information Visualisation Infovis05, 23-25 ottobre, Minneapolis, Minnesota, USA.

C. Binding and D. Tudhope 2004. KOS at your Service: Programmatic Access to knowledge Organisation Systems. Journal of Digital Information, 4(4).

B. Bustos, D. Keim, D. Saupe, and T. Schreck 2007. Content-based 3d object retrieval. IEEE ComputerGraphics & Applications, special issue on 3D documents, Jul/Aug 2007. to appear.

L. Candela, D. Castelli, Y. Ioannidis, G. Koutrika, P. Pagano, S. Ross, H.-J. Schek, H. Schuldt and C. Thanos 2007. Setting the Foundations of Digital Libraries – The DELOS Manifesto. D-Lib Magazine Volume 13 Number 3/4.

P. Cao and Z. Wang 2004. Efficient top-k query calculation in distributed networks. In PODC, pages 206–215.

T. Catarci, M.F. Costabile, G. Santucci 1999. Foundations of the DARE System for Drawing Adequate Representations. In Proc. of DANTE'99, Japan, 1999.

T. Catarci, G. Santucci 2001. The prototype of the DARE system - nei Proceedings della conferenza internazionale SIGMOD2001, USA, Maggio 2001.

C. Grün, J. Gerken, H. Jetter, W. König and H. Reiterer 2005. MedioVis - A User-Centred Library Metadata Browser. In Proceedings of the 9th European Conference, ECDL, Research and Advanced Technology for Digital Libraries, Wien, Springer, 174-185.

M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen and K.P. Yee 2002. Finding the flow in web site search. In: Communications of the ACM 45. Nr. 9, S. 42–49.

H. Jetter, J. Gerken., W. König, C. Grün and H. Reiterer 2005. HyperGrid – Accessing Complex Information Spaces. In People and Computers XIX - The Bigger Picture, Proceedings of HCI 2005, Springer Verlag, Edinburgh, UK.

A. Karanastasi, A. Zotos and S. Christodoulakis 2006. "User Interactions with Multimedia Repositories using Natural Language Interfaces - OntoNL: an Architectural Framework and its Implementation", In the Journal of Digital Information Management (JDIM), Volume4, Issue 4.

S. Michel, P. Triantafillou, and G. Weikum 2005. Klee: A framework for distributed top-k query algorithms. In VLDB, pages 637–648.

Mlivoncic M., Schuler C. and Türker C. 2004. Hyperdatabase Infrastructure for Management and Search of Multimedia Collections. In DELOS Workshop: DL Architectures : 25–36.

S. Rexhausen, M. Demarmels, H. Jetter, M. Heilig, J. Gerken and H. Reiterer 2007. Blockbuster – A Visual Explorer for Motion Picture Data. In INFOVIS: Proceedings of the IEEE Symposium on Information Visualization (INFOVIS'07), IEEE Computer Society.

A. Rowstron and P. Druschel 2001. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In Middleware.

Schek H.-J. and Schuldt H 2006. DelosDLMS - Infrastructure for the Next Generation of Digital Library Management Systems. In ERCIM News, Special Issue on European Digital Library 66: 22-24.

Schuldt H., Alonso G., Beeri C. and Schek H.-J. 2002. Atomicity and Isolation for Transactional Processes. In ACM Transactions on Database Systems, 27(1): 63–116.

C. Schuler, H. Schuldt, C. Türker, R. Weber and H.J. Schek 2005. Peer-to-peer Execution of (transactional) Processes. Int. Journal of Cooperative Information Systems 14(4) 377–406.

M. Theobald, R. Schenkel, and G. Weikum 2005. An efficient and versatile query engine for topX search. In VLDB.

C. Tsinaraki, P. Polydoros and S. Christodoulakis 2007. "Interoperability support between MPEG-7/21 and OWL in DS-MIRF", In Transactions on Knowledge and Data Engineering (IEEE-TKDE), Special Issue on the Semantic Web Era.

D. Tudhope and C. Binding 2006. Towards Terminology Services: experiences with a pilot web service thesaurus browser. ASIS&T Bulletin 32(5), 6–9, June/July 2006.

R. Weber, H.-J. Schek and S. Blott 1998. A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces. In: Proceedings of the 24rd International Conference on Very Large Data Bases, 194–205, San Francisco (USA).

MedioVis 2007. Project Website: http://hci.uni-konstanz.de/MedioVis (last accessed 30 Oct 2007).

SKOS 2004, Simple Knowledge Organisation Systems. W3C Semantic Web Deployment Working Group. http://www.w3.org/2004/02/skos/

STAR – Semantic Technologies for Archaeological Resources
http://hypermedia.research.glam.ac.uk/kos/star/