A Distributed Multimedia Information System for Cultural Heritage Identity preservation

Aldo Aiello, Mario Mango Furnari, and Antonio Massarotti

Abstract— *In this paper we describe the architecture of a Distributed Content Management System developed to organize the cultural information in such a way that the cultural identity of a community could be easily communicated. One of the major problems investigated was how the information integration could simplify the "relevant" information looking for process. Here, the notion of contents community was developed, together with a set of tools, in order to organize the document information space. Each content provider publishes a set of ontologies, to collect metadata information, organized and published through the Contents Community Authority. These methodologies were deployed setting up a prototype that connects about 20 museums in the city of Naples (Italy).*

Index Terms—***Multimedia Information System, Distributed Information Systems, Cultural Heritage Content Management***

I. INTRODUCTION

In the last decade, cultural heritage has received an increased attention and has been recognized as an important aspect for social groups in order to preserve human being community identity. Many efforts have been devoted to cope with the Cultural Heritage preservation, promotion, and economic exploitation problems. To a greater degree, technology is solving one of the largest problematic issues concerning cultural heritage assets: their *nondestructive public access*. Never before, there have been greater opportunities to explore and discover in great details these marvels of earth and of humankind without the fear of irreparable damage. Cultural heritage preservation technology is now establishing new ways to explore complex problems and to provide solid pathways to real solutions.

Many information systems and international initiatives were started up to collect and manage information about *cultural heritage artifacts*. Furthermore, to gain a wide audience and to promote a standardization process, many efforts are on going, for example [2], [3], [4], and [8].

With the wide acceptance of the *World Wide Web* (WWW) metaphor, most systems were transformed to replace the notion of record with that of *document* as elementary information entity around which to design the information systems. On designing a Multimedia Information System to promote cultural community identity this change over is becoming more evident and it is stressing out the browsing and portal approaches limitations. In fact, the cultural identity of a community is only partially represented by the cultural heritage artifacts organized in Museums. We think that a more comprehensive representation is better given by showing up all the relationships that there exist between museum artifacts and social-urban tissue.

The objectives of Cultural Heritage Information Systems should be established as a federated network of culture related information providers, where all contents should be available to mass public,

Manuscript received June 30, 2004. This work was supported in part by the Italian Ministry of Research & University, under Grant C29/M3/P12

The authors are with *Istituto di Cibernetica Eduardo Cainiello, Via Campi Flegrei, 34, 80078 Pozzuoli Italy* (e-mail: mf@cib.na.cnr.it).

professionals and market operators through cooperating information systems. For such systems, the cooperation process should be focused on the re-organization and unification process of the existing relevant information resources. The cooperation would account for heterogeneous, dynamically changing and autonomous services to be combined into a single logical service. One promising approach to be exploited on pursuing the previous goals is given by the Semantic Web Initiative [1].

In this paper, we address the problem of making existing distributed document collection repositories mutually interoperable at semantic level. We argue that emerging semantic web technologies offer a promising approach to facilitate semantic information retrieval based on heterogeneous document repositories distributed on the web. Here, we also describe the authors' efforts in designing and implementing a test bed to verify on the field some of the emerging web technologies to be deployed in order to experiment the semantic web approach, on the cultural heritage promotion arena.

The rest of the paper is organized as follows: In Section II the architecture and the implementation of the proposed Distributed Multimedia Document Information System are given. In Section III the implemented test bed is described. In Section IV, the advantages of the proposed architecture are summarized by comparing them with other efforts.

II. THE ARCHITECTURE OF A DISTRIBUTED CONTENTS MANAGEMENT SYSTEM

As distributed contents management system design criteria we chose the WWW paradigm: where *document* plays the role of elementary information and it is the basic information systems building block. Furthermore, we adopted the multi-tiers web architecture, with the application server playing the central role of business logic driver. In other words, we have an http server taking care of all the interactions with the external world and being also in charge to asynchronously send messages to the application server. The application server takes care of the *request* messages, interprets them creating chains of actions that it dispatches to corresponding specialized components. Once an invoked component has terminated the computations, it sends back the responses to the application server that assembles them into *response* messages and sends them back, through the http server, to the clients located into the external world. The main identified components belonging to the *application layer* are:

- *Document Repository System* (DRS). The DRS stores and organizes the documents together with the associated metadata.
- *Document Access System* (DAS). The DAS creates friendly and flexible user interfaces to discover and access the contents.
- *Contents Authority Management System (CAS).* The CAS stores and manages the ontologies used by each participating node to facilitate the DRS semantic interoperability.

All these systems communicate among them exchanging XML encoded messages over http, according to well-defined protocols that represent the XML communication bus core. The components of each multimedia documents repository node are implemented as a collection of servers coordinated by an application server and published on Internet by an http server.

From the functional point of view, the distributed system is built as a collection of multimedia documents repository nodes, glued all together by a communication infrastructure that takes care not only for the communication channels but even for the necessary services/components for nodes registration, for XML messages routing, and so on. Since, in a distributed setting, the network infrastructure and services must be continuously operating, the communication infrastructure must be augmented with configuration and monitoring systems. All these services and components are grouped into the *network layer*.

Summarizing, we have that the proposed architecture is based on the following three structuring layers:

- *network*, where all the software components are described, in term of web-components and application logic;
- *application*, described in terms of

execution chains starting from the available multi-tiers web components and coordinated by the application servers;

- *contents*, where the information describing a particular domain are encapsulated in digital documents and collections.

The proposed architecture advantages are: a) ease of deployment on Internet, high reliability and fault-tolerance, and efficient use of the network infrastructures; b) flexibility and generality as needed in order to evolve and meet future needs; c) scalability without fundamental changes in the structure of the resource name spaces.

In the rest of this section, the components are described according to the architecture previously defined and sketched in Figure 1.

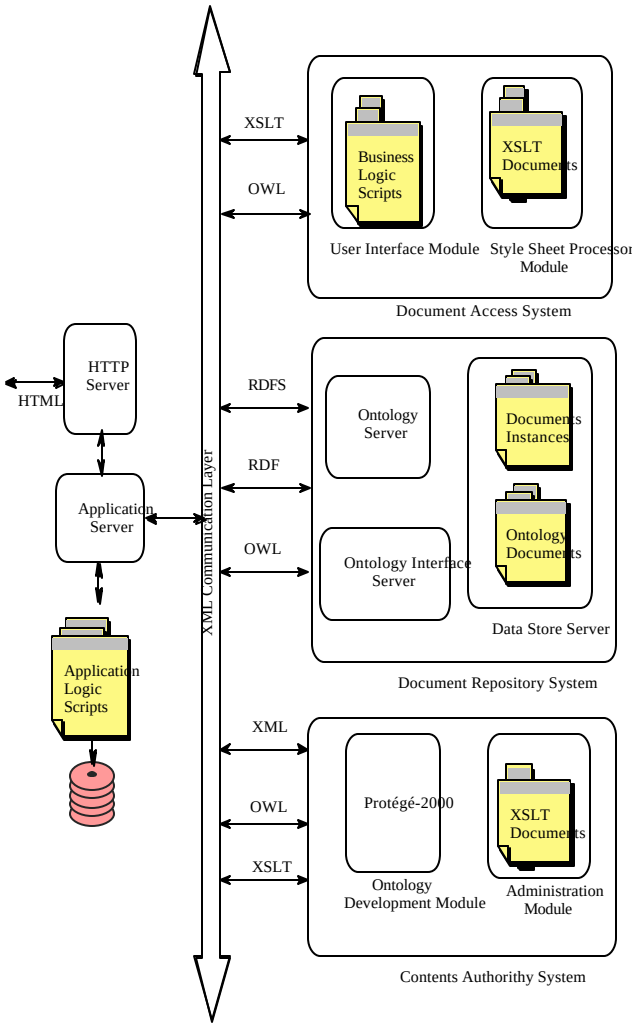

Figure 1: Distributed Contents Management architecture.

A. The Documents Repository System

A new methodology for describing the documents topics is emerging from the semantic web researches. It is based on the notion of ontology. Ontologies are utilized to define the underlying concepts and the corresponding semantic interpretation that maps terms and descriptions to the concepts. Since ontology-based information technology promises to simplify the interoperability problems, we chose to design the Document Repository System around an ontology server.

In fact, our Document Repository System is built starting from three kinds of servers:

Data Store Server

The documents are represented as digital objects together with the associated metadata information. Here, metadata are organized using domain ontology. The Data Store Module is composed of a document media repository, which stores the digital representations of the document contents according to a set of XML applications, and a metadata repository, that stores all the document annotations that are XML-encoded and organized according to RDF model [7] and [17]. This kind of documents structuring and coding strategy makes possible to separate the document layout implementation from its contents.

The Sesame package [14] is the main Data Store Module software component. It is an open source, platform-independent, RDF Schema-based repository, provided with querying facility written in Java. The low level persistent storage is achieved using Postgresql [16], one of most widely used public domain database environment. The Sesame environment offers three different levels of programming interfaces: the client API, for client-server programming; the server API; and the lower level *Storage and Inference Layer* (SAIL) API, for the RDF repositories.

Ontology Server

The ontology server provides the Document Repository System with the basis for the semantic interoperability capabilities. Conceptually, it is the most important type of servers since it manages the OWL/RDF [15] schema for the stored data, and determines the interactions with the other servers and/or

modules, through the ontology exchange protocol [10].
Each ontological feature is associated with a domain ontology; for example, ontologies for artifact, material and techniques have been defined according to the Italian Istituto Centrale di Catalogo e Documentazione (ICCD) standard, adopted by several museums mangers to archive art crafts data. The ontology descriptor is an RDF descriptor that summarizes the covered domain. It is used to annotate the documents, for each ontology component. The ontology RDF descriptor and the corresponding ontologies are stored into the metadata repository, and can be accessed through the ontology exchange protocol.

Ontology Interface Server

The Ontology Interface Server consists of a set of functionalities for walking through the ontology graph and the associated attributes. At runtime, these functionalities are used by the Document Access System to build the user interfaces, the browsing structures, the application services, and so forth. For example, to build the management user interface, it is necessary to create a set of dynamic forms, according to a classification schema, synthesized into the corresponding ontology. The Ontology Interface Server can be queried about the ontology class hierarchy, and/or the class properties, giving back an RDF document that could be transformed into HTML forms.

B. The Document Access System

The user will interact with the system through a conventional browser; the DRS appears and behaves like a traditional web site whose appearance is shown in Figure 2.

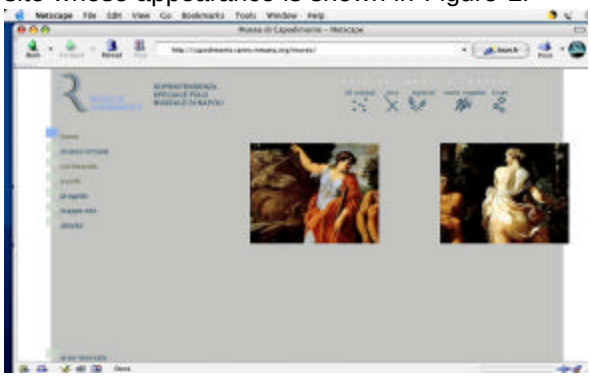

Figure 2: Home page of participant museums

Documents must underway a text processing before to be displayed, and programmed according to a sequence of transformations expressed using the Extensible Stylesheet Language Transformation (XSLT) [6]. The Document Access System manages this document composition process, whose business logic could be summarized as follows: the ontology client makes the first step by extracting the information from the data store and wrapping this information with XML tags. The extraction is done querying the ontology server. The second step involves the application of the appropriate stylesheet transformations to the XML data and thereby the creation of a corresponding HTML page. The foregoing step is carried out by the XSLT package included in the application server. The output of that transformation is the HTML page that is directly sent to the browser.

Style Sheet Processor Module

This module is responsible for taking RDF data from the data store and converting it to the appropriate HTML document to be presented to the user. The Stylesheet Processor module consists of the Python XSLT package managed by the Zope [18] application server.

User Interface module

The User Interface module is the communication point between the user and the application. An administrator has the ability to input new information through the system, while a visitor can easily (indirectly) query the document repository through HTML links. This module provides the implementation of ontology clients that extract the needed information from the ontology server.

C. The Contents Authority System

The Contents Authority provides the museum managers with the possibility of interacting with heterogeneous and distributed document repositories. Actually, it allows for the dynamic construction of collections that could belong to the different repositories. Furthermore, it guarantees the necessary autonomy to museum managers in organizing their contents space.

To achieve these goals the CA is equipped with the following modules:

Ontology Development Module. We used the Protégé-2000 ontology editor for developing and editing the ontologies. Thereafter, those ontologies were uploaded into the data store module and used to populate it through the administration module. We developed an extension for the OWL Protégé-2000 Plug-in in order to store the ontology directly on the Data Store Module using the client/server metaphor.

Administration Module. The application uses Zope managed forms for inputting new information according to the adopted ontologies. The administration module provides the ability to add new publications or new collection on the document repository. The forms are processed through Zope, the data are transformed into OWL instances, and the latter are sent to the Data Store Module to be made persistent.

For what concerns many communication issues pertaining to the interchange of ontologies, as well as "pieces" of ontologies, we designed and adopted ezXML4OWL, our own ontology representation language [19]. This permits to reduce an OWL ontology in autonomous meaningfully interchangeable objects called OWL-modules. CAS handles such modules in order to manage partial variations of ontologies as well as to merge ontologies coming from different partners. We designed various specific types of OWL-modules and sub-modules and ezXML4OWL is provided with constructs for serializing them in XML elements and documents.

III. THE MUSEO VIRTUALE DI NAPOLI TEST BED

Usually, the aim of any ordinary museum visitor is something quite different from trying to find certain objects. Visitors would rather want to learn about the past and they experience it with the help of the collections contextual information. In physical exhibitions, the cognitive museum experience is often based on the thematic combination of exhibits and their contextual information. In order to realize how much it would be complex to achieve these goals, the research and development project* "Museo Virtuale di Napoli: Rete dei Musei Napoletani"[1] (REMUNA) is carried out at the Istituto di Cibernetica E. Caianiello. Its main objective is to develop a distributed multimedia contents management environment, to allow for contents integration and management and to facilitate the cooperation among different but correlated knowledge domains. The expected results are an increased museum manager's synergy based on the information integration, and the deployment of a large distributed document repository system on top of the emerging web technology, using cost affordable hardware, software and networking infrastructures.

As a case study, the collection document repositories of eighteen Neapolitan museums are used. These document repositories use different technologies, have different conceptual schemas and are physically located in different districts of Naples. Each museum is equipped with multimedia information system and communication infrastructures. From the museum managers' (information providers) perspective each information system allows him to make available the managed artefacts' information through the REMUNA environment, just after registering them into the system. In fact, during the document registration process, the system extracts metadata from the cultural heritage artifact document description and creates a new digital document that encapsulates the various pieces of metadata about the given artifact. All this information is encapsulated into a digital object that plays the role of a handle for the actual artifact information. No assumption about fixed attributes names' schemata is taken, so the application builder can create new attributes as needed just modifying the associated ontology without changing the internal database schemata.

The information provider[2] could also organize a set of related documents, in document collections, according to some relationships defined on top of the

* The project "Museo Virtuale di Napoli: Rete dei Musei Napoletani" is supported by Ministero dell'Università, Ricerca e Tecnologia, under contract C29/P12/M03, from here on denoted with REMUNA

[2] In this paper we assume that *information provider* means the responsible, inside the museum organization, of the cultural heritage goods information.

associated ontology. The adopted notion of collection is a recursive one, in the sense that a collection could contain other collections. Each digital document is allowed to belong to multiple collections and may have multiple relationships with other documents. This nesting feature forms the document repository collection graph, and allows the system to deliver more than one logical view of a given digital documents asset.

To assure the necessary operational autonomy to the information provider, without reducing the cooperation opportunities with other information providers, we deployed the Cultural Heritage Content Authority as an intermediate coordination organization that is in charge to register and syndicate document contents, to guarantee their quality. Since the presence of a content authority could create a bottleneck, the notion of delegation was introduced. In other words, the top authority could delegate another organization to operate as Cultural Heritage Contents Authority on its behalf, for a more specific content domain.

IV. CONCLUSION

This paper describes the methodological choices on which we developed an integrated Web environment, equipped with web components and applications for producing, archiving, and disseminating information about the Neapolitan Cultural Heritage, over Internet. We choose Naples as our test bed since it has been a very active cultural center since the ancient Greek age, was one of the European capitals, until the Italian unification, and still plays an important role in the European heritage.

One of the most interesting technological aspects investigated was how to design document repositories systems that allow the museum manager to organise the cultural heritage heterogeneous information space spread in many autonomous organizations. To cope with this aspect we proposed the Cultural Heritage Content Authority component that is in charge of organizing and structuring the contents of the participating institutions. In other words, the Cultural Heritage Content Authority does not directly manage contents but manages the rules to be adopted in organizing the contents space on a specified area.

These Cultural Heritage Contents Authority definition and deployment are similar to those of the Publishing Authority used in the “Networked Computer Science Technical Report Library” (NCSTRL) project, started up and carried out at Cornell University [9]. The NCSTRL collection is logically and administratively divided into publishing authorities, and each publishing authority has control over addition and administration of documents in its own sub-collection repositories.

The Ontology Exchange Protocol and tools were implemented to exploit the Multimedia Document Information System federation settlement. This protocol is very similar to the Dienst [9] collection service, where the main difference relies on the fact that in our case the collections are entities built on top of an ontology describing the domain of the documents content and not predefined ones. To a certain degree, our usage is similar to that of the CIMI project [3]. In fact, it has become increasingly evident that simple application-specific standard, such as Dublin Core (DC) [4], cannot satisfy the requirements of communities such as BIBLINK [2] and OAI [13] that need to combine metadata standards for simple resource discovery process.

Our work successfully showed that an RDF data store (Sesame) could be used as a backend document repository for a distributed Contents Management System (CMS). It was clearly laid a framework for possibly implementing semantic capabilities.

As the Semantic Web begins to fully take shape, this type of distributed CMS implementation will enable agents to understand what is actually being presented in distributed CMS, since all content within the system is modeled in machine understandable OWL/RDF.

Starting from these encouraging results we are planning to actively pursue some of the goals foreseen by the Semantic Web Initiative [1], [11], and [12]. For example, to gain more semantic information we are

exploiting pieces of well-known and supported ontologies, like ICOM-CIDOC [8]. Also, more advanced semantic capabilities (searching and browsing) can be implemented. In fact, lastly, with the aid of the underlying ontology, suggested links (both inside and outside the system) are provided to users, while they are browsing the site.

ACKNOWLEDGMENT

Acknowledgments are expressed to all the people of Istituto di Cibernetica E. Caianiello that worked on the ReMuNa project, for their help, and fruitful discussions, and also to all the staff members of the Soprintendenza ai Beni Archeologici delle Province di Napoli e Caserta, Soprintendenza ai beni Artistici, Storici e Demo Antropologici della Provincia di Napoli, Soprintenda ai Beni Architettonici ed Ambientali della Provincia di Napoli, Archivio di Stato di Napoli, to the people of Direzione Musei of Comune di Napoli, and the Assessorato alla Cultura of Comune di Napoli, without their assistance the REMUNA project and activities would not exist.

REFERENCES

[1] Berners-Lee, T., "WWW: Past, Present, and Future", IEEE Computer, 29, (1996)
[2] The BIBLINK Core Application Profile, http://www.schemas-forum.org/registry/biblink/BC-schema.html
[3] CIMI. (http://www.cimi.org/)
[4] The Dublin Core Metadata Initiative, http://www.purl.org/dc/
[5] Davis J. and Lagoze C. , The Networked Computer Science Technical Report Library, Cornell CS TR96-1595
[6] Extensible Style Language for Transformation: http://www.w3c.org/Style/XSLT
[7] Lassila O., Swick R., Resource Description Framework (RDF) Model and Syntax, World Wide Consortium Working Draft
[8] ICOM/CIDOC Documentation Standard Group, Revised Definition of the CIDOC Conceptual Reference Model, 1999, http://cidoc.ics.forth.gr/
[9] Lagoze C., Shaw E., Davis J.R. and Krafft D.B., Dienst: Implementation Reference Manual. May 5, 1995.
[10] Mango Furnari M., Aiello A., Caputo V. Barone V., Ontology Server Protocol Specification, ICIB TR-12/03
[11] McIlraith, S. A., Cao Son, T., Zeng, H., "Semantic Web Services", IEEE Intelligent Systems, pp. 46-53, 2001
[12] McIlraith, S. A., Cao Son, T., Zeng, H., "Mobilizing the Semantic Web with DAML-Enabled Web Services", Semantic Web Workshop, HonKong, China, 2001
[13] Open Archives Initiative, http://www.openarchives.org
[14] http://www.openrdf.org/documentation.jsp
[15] OWL Web Ontology Language Overview, http://www.w3.org/TR/2003/PR-owl-features-20031215/
[16] http://www.postgresql.org/
[17] Resource Description Format: http://www.w3c.org/RDF
[18] http://www.zope.org
[19] [19] Mango Furnari M., Aiello A., Massarotti A., ezXML4OWL. An easy XML for OWL, ICIB TR-06/04.