CiVeDi: A Customized Virtual Environment for Database Interaction *

Pietro Mazzoleni [1], Elisa Bertino [2], Elena Ferrari [3], Stefano Valtolina [1]

[1] Dip. di Informatica e Comunicazione, University of Milan, Italy
{mazzoleni,valtolina}@dico.unimi.it
[2] Computer Sciences Dep., Purdue University, USA
bertino@cs.purdue.edu
[3] Dip. di Scienze della Cultura, Politiche e dell'Informazione, University of Insubria at Como, Italy
elena.ferrari@uninsubria.it

Abstract

This paper presents CiVedi, a scalable system providing a flexible and customizable virtual environment for displaying multimedia contents. Using CiVeDi, both the final users and the exhibition curators can personalize the content of the visit as well as the visit appearance and its duration. The proposed solution aims to be used transparently over different media objects either stored into a database or dynamically collected from online digital libraries.

1 Introduction

During the last few years, there has been a growing interest in digitalizing and publishing cultural heritage information and in developing innovative and easy-to-use tools for their accesses (e.g., [7, 8]). However, whereas storing and retrieving data objects from a database are well-established processes - current multimedia DBMSs (MMDBMSs) offer, in fact, tools for efficiently managing all different types of multimedia data - the development of flexible publishing techniques, suitable for users with different skills and needs, is still an open issue. In this respect, modern information and communication technologies have made available to exhibitors and in general cultural institution administrators a large variety of tools and techniques to address their communication and presentation requirements. Among the various alternatives today available, Virtual Reality (VR) seems to be one of the most promising technology to bolster interest for heritage [1, 2, 3]. Although several projects have been undertaken in this field, most of them focused on creating realistic reconstructions of relevant historical places, or pre-defined 3D tours of important museum exhibitions [4, 5, 6]. By contrast, no comparable amount of work has been carried out to enable dynamic interaction with multimedia contents and customization for final users which, in most cases, either have to follow a predefined tour or they can choose only among a limited set of predefined options. On the other hand, most of the museums and other cultural institutions have already large amount of multimedia data stored into databases that are usually accessible only to experts, through a standard web-based interface which does not fit the needs of a large public [7, 8].

Therefore, there is the need of designing alternative, non-conventional ways of presenting multimedia contents to a large public. In this respect, we believe that a promising approach is that of combining MMDBs and VR or, more precisely, exploiting VR as a solution to retrieve information from a MMDB and to present them. The integration of VR and DB technologies entails, however, addressing several issues, mainly related to the fact that the types and number of objects returned by a user query cannot be know in advance. Thus, the VR system must be able to dynamically re-arrange the VR assignment, in order to be able to accommodate query results.

In this paper, we describe *CiVeDi* - a scalable system, whose main goal is that of providing a flexible and personalized virtual environment for accessing multimedia contents. By using CiVeDi, the final user not only can specify the information he/she is interested in but he/she can fully customize the appearance and the duration of the visit. User preferences are also integrated with the requirements coming from the curator of the exhibition, which, on the basis of his/her knowledge about the contents, can provide the system with additional information in order to generate tours that better facilitate the fruition of the available cultural heritages.

The remainder of this paper is organized as follows.

* This work has been partially supported by DELOS Research Network.

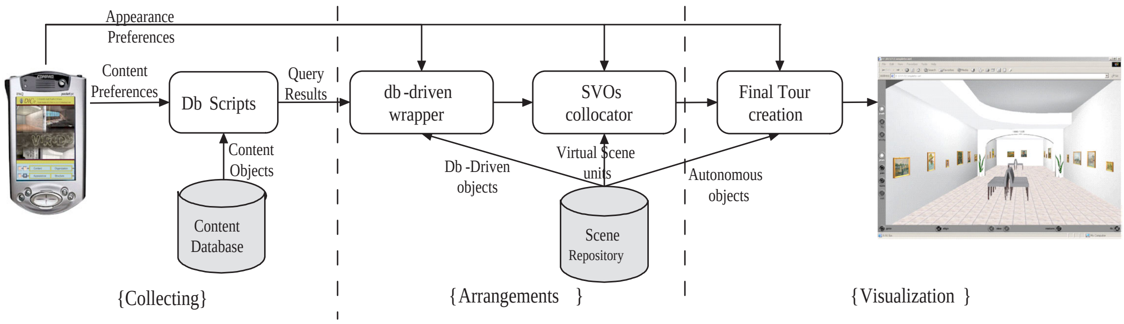

Figure 1: Tour generation phase

Section 2 presents an overview of the proposed methodology. Sections 3 and 4 detail the two main phases of the process, that is, the data specification and the tour generation phase. Section 5 describes the prototype we have developed. Section 6 reviews the state of the art, whereas Section 7 concludes the paper outlining future work.

2 Overview of our approach

In this section, we present the methodology we have designed to build customized virtual environments for delivering multimedia contents. The process can be divided into two main phases: the *Data specification phase* and the *Tour generation phase*.

1. *Data specification phase*: it is performed by the responsible of the cultural institution which owns the data, before making the system available to large public. Based on his/her knowledge about the content, the curator provides the system with information useful to generate the tours, with the goals of letting the users to come as better as possible in contact with the heritage available into the database. Specifically, the curator specifies information concerning possible user queries and a set of feasible museum-architectures where to arrange the exhibition.

2. *Tour generation phase*: it is the core of CiVeDi with the aim of generating a customized virtual tour based on user requests. As illustrated in Figure 1, this phase can be further divided into three sub-phases: the *Collecting* phase, in which the users specify their requirements and the query they wish to submit to the MMDB; the *Arrangement* phase, in which the system collects query results and arranges them within the virtual environment making use of a set of heuristics; and, finally, the *Visualization* phase, which creates the final tour and deploys it back to the user who can then start the visit.

3 Data Specification

In this section, we describe the main components of the data model underlying CiVeDi and describe how a museum curator can make use of them to customize the system for the deployment of a certain content. Main components of the data model are the multimedia contents to be displayed and the virtual elements that can be used to create an exhibition.

One of the main features of CiVeDi is to be independent from the database storing the source data. In other words, the system can be applied over any DBMS, independently from its data model or the information it contains. Therefore, in the following we present the various types of data supported abstracting from specific implementation details.

3.1 Multimedia Contents

CiVedi, even if extensible to support requests and visualizations of a wide range of multimedia objects, has been initially tailored to create expositions of paintings stored as images into the database.
We assume that each image is stored into the database along with some *metadata*, which provide additional information (such as for instance the author, the description, of the physical size of a painting) about the artifact. Metadata are returned as result to a query along with the corresponding objects and they are used by the system in the tour generation phase to decide the final appearance of an object into the virtual environment.

The way metadata are represented does not depend from our system but it is related to the DBMS collecting source elements. We could thus have them as tuples, in case of relational DBMSs, objects for OODBMSs, or XML documents, in case of XML-native databases. To deal with heterogeneous data models, we have built ad-hoc wrappers which organize metadata into an XML-based structure used into the following phases of the process. Figure 2 shows an example of an image object and the corresponding metadata.

Name	value
Title	The starry night
Author	V. Van Gogh
Painted in	Saint Remy
Date	June 1889
Technique	Oil on Canvas
Location	MOMA, NY

Figure 2: Example of MM object and its metadata

3.2 Scene repository

The scene repository collects virtual elements which are used as templates to create customized virtual environments. These objects, which are stored separately from the multimedia contents, are loaded on demand into the system, adapted to the query results, and finally included into the virtual tour. We can classify such elements based on how they are used by the system. In this respect, we distinguish *virtual scene units*, *db-driven objects*, and *autonomous objects*, described in what follows.

Virtual Scene Units. A *Virtual Scene Unit*, *VSU* is a 3D template for an area (e.g., a room) which has been modeled in advance and can be loaded by the system to arrange multimedia objects. In addition to a geometry (geometrical model) and an appearance (textures, lights and so on), which are directly related to the virtual model itself, VSUs contain additional information which identify a specific museum architecture (e.g., a modern museum, a church, the "Galleria Uffizi", and so forth), the physical size (along x,y and z axis), and some additional parameters the system uses to resize the template when needed. In fact, a VSU does not have fixed dimensions, instead it can be expanded or reduced according to the multimedia objects it has to contain. Moreover, VSUs contain information about the physical areas in which the system can arrange multimedia elements. We refer to such information as *Active Expositive Areas*. An active expositive area of a VSU represents an available area in the VSU in which information returned from the multimedia database can be visualized. By mean of expositive areas, modellers are able to exactly define spaces where elements can be arranged within a particular architecture. Figure 3 (A) shows a sketch of a VSU where dashed rectangles represent active expositive areas and black areas denote doors.

Db-Driven Objects. Db-driven objects are 3D models that represent abstractions used by the system to wrap data returned by the database, before arranging them into a VSU (e.g. different frames for images) Similar to VSUs, also for db-driven objects additional information are associated with the 3D model. Such information are the type of data they are willing to wrap (e.g., images), the architectural style (e.g., modern frame, XIX century frame), and a function representing the amount of expositive area, referred to as *passive expositive area*, occupied by a multimedia object wrapped using the considered db-driven object (i.e., including the size of the multimedia object, the area occupied by the frame, and some information about the minimum distance from an artifact to another).

Autonomous Objects. These are 3D elements which are not related to the information returned from the MMDBMS. Instead, they are placed into a scene mainly for aesthetic purposes so to have a more realistic environment. Examples of autonomous objects are windows, plants, benches, chandeliers and so on. Autonomous objects are not modeled together with the scenes but dynamically arranged by the system based on some heuristics. Again, autonomous objects are identified by their 3D models, a distinguishing architectural feature (e.g., a window of a church or a modern flowerpot), a passive expositive area (resizable if needed), and additional information about the type of element (plant, door, chair, etc) the autonomous object represents.

CiVeDi assumes VSUs, db-driven and autonomous objects to be available to museum curators which intend to use our system to disseminate their content. During the data specification phase, the curator uses his/her knowledge about the content to specify the set of queries a user can submit (for instance, queries based on a specific time-period and/or the names of the authors) and the set of museum architectures (including both db-driven and autonomous objects) which better fit the information stored into the MMDBMS.

4 Tour Generation

In this section, we discuss the main activities carried out by the system to create customized virtual environments for accessing multimedia content . According to Figure 1, the tour generation phase is organized in three sub-phases: *Collecting phase*, *Arrangement phase*, and *Visualization phase*, which are briefly described in the following.

4.1 Collecting Phase

It is the beginning of the process and its goal is to collect all user requirements needed to create the virtual tour and to submit the query to the MMDB. Users can specify three types of requirements to customize their visit: *tour content*, *tour appearance*, and *tour duration*, which give them the feeling of personalizing both the content and the rendering according to their needs.

Tour content. The first set of preferences a user can specify concerns the artifacts displayed in the virtual exhibition. Based on his/her knowledge, experiences, age, education and interests, users specify which information, from the ones made available by the curators, he/she is willing to have presented into his/her customized museum.

Tour Appearance. Other than the content, user is allowed to customize the appearance of the virtual environment itself. Users can specify the museum architecture, selecting among various templates indicated by the museum curator, and define how the various rooms should be combined to generate, for example, a linear or a more complex non linear tour.

Tour Duration. The latest set of parametrizable features concerns the specification of the overall duration of the customized virtual visit. We propose three different approaches to control the tour-duration: a *rubber-band technique*, that gently pushes the user from a room to the next one as time goes by; a *time-based rollover* among artifacts visualized into the scenes; and, finally, a *query result pruning*, based on information stored into the MMDBMS.

Finally, a user can specify one or more clustering criteria that the system will consider to decide which artifacts, among the all returned from the database, should be arranged together into a scene.

Content requirements information are then used to generate the query and submit it to the content database.

4.2 Arrangement Phase

This phase takes as input the artifacts returned from the content database and comprises two main modules: the *db-driven wrapper*, whose task is to assign a db-driven object to each returned artifact, and the *VSU collocator*, which assigns a VSU to each artifact based on user preferences. The Db-driven wrapper is also in charge of resizing the images to a dimension proportional to their real sizes.

To arrange objects into the virtual space, we have developed a set of heuristics to create realistic visits. For each cluster of artifacts, the *VSU collocator* will first identify and load a VSU template among the ones whose architecture corresponds to the one specified by the user. Then, it starts assigning each artifact into the largest expositive area available into the VSU. Once assigned, the value of the active ares is reduced by the passive areas occupied by the artifact. If all the objects in the cluster do not fit in the selected VSU, some relaxation techniques such as scene enlargement (within the bound fixed by the modeler) or organizing the artifacts (i.e., paintings) into two parallel lines instead of one in each wall are applied. Furthermore, there will be cases when the system will load another scene until all the objects have been assigned to a VSU. The result of this phase is a list of VSUs, each one identified by an ID and the cluster information it contains, and for each VSU a list of its active expositive areas and the artifacts assigned to each of them.

4.3 Visualization Phase

Before applying the algorithms to draw 3D scenes and objects composing the user customized virtual exhibition, the visualization phase uses some additional techniques to refine the results of the previous phases and to combine all the loaded VSUs into a single tour. More precisely, the system evaluates each scene and in case some active expositive areas are not completely filled, it compacts them by reducing VSU dimensions or by combining artifacts of different active expositive areas into a single one. In this case, the free expositive areas left with no-objects are automatically filled by autonomous objects such as windows or columns, as shown in Figure 3 (C). Finally, an absolute position is computed for each element of the exhibition, and the 3D visit generated.

5 Implementation

A prototype of CiVeDi has been developed at University of Milan. The prototype has been built using the JAVA language, whereas virtual tours are generated according to the X3D [1] format. X3D stands for eXtensible 3D Language and it is a next-generation Extensible 3D Graphics specification language which aims at extending and upgrading the geometry and behavior capabilities of the Virtual Reality Modeling Language (VRML 97) using XML to generate virtual reality environments which can also be presented via a common web-browser. VSUs and db-driven objects have been built by using the *prototype* feature of X3D which offers the possibility of creating parameterized elements generated on the fly based on some input-specification.

The system supports the user with two graphical interfaces. The first one, reserved to the museum curator, is a web-based interface running on a desktop system, through which an exhibitor can visually specify the types of queries a user can submit as well as the architectures in which the customized exhibitions could be arranged (by loading them from the scene repository). The second interface, used by the final user, is running as a web-page on a PDA and it is dynamically generated according to the requirements set in advance by the curator.

For the sake of testing, CiVeDi has been used to generate customized virtual exhibitions of paintings belonging to the Van-Gogh museum in Amsterdam [13] which have been stored into an Oracle DBMS.

Figure 3 (C) shows an example of a customized virtual gallery of Van-Gogh works of art, we have generated using our system.

Particular attention has been devoted, during both the design and implementation of CiVedi, to usability require-

[1] www.web3d.org/x3d.html

(A) (B) (C)

Figure 3: Customized Virtual Tour

ments [9], which are clearly a key aspect for the success of the system. Following the ISO 9241 standard, we used three main usability indicators: effectiveness in use, efficiency in use, and satisfaction in use. Additionally, we are interested in evaluating both *curator usability*, intended to evaluate the usability of the system from the exhibition curator point of view, and *final user* usability, which evaluates usability wrt the final user. In developing CiVedi, we try to meet usability requirements by actively involving final users and exhibition curators during each stage of the development process. In this respect, we follow an iterative approach where the feedback of the users drive the refinement of the devised solutions. Now that a complete prototype of the system has been developed, we are starting a set of empirical experiments to test usability. We plan to divide final users into a set of categories, based on their experience (on both the use of IT and VR techniques and on the kind of information on which the exhibition is based) and background (examples of users are exhibition curators, children, students of Humanities and so on), and to perform experiments to evaluate the usability of the two interfaces provided by CiVeDi, and the virtual environment in which the multimedia contents are displayed. The experiments will be conducted after a training session which is different for each class of users.We plan to adopt the observational evaluation method [9] to evaluate system usability, a well-known method which requires to observe final users when interacting with the system. Moreover, we plan to carry on some experiments according to the Walkthrough methodology [10], which requires to divide the whole system into tasks and subtasks which are separately evaluated in order to identify possible steps which could block the user from completing a task.

6 Related Work

The problem of disseminating cultural heritage information to the large public is one of the most pressing needs of all cultural institutes. Museums have today available a large amount of digital information which might be used for different purposes such as enhancing the visit of the real exhibition, attracting new users to the museum, making collections available to researchers, or simply contributing in increasing the overall level of cultural knowledge of the population. Notwithstanding most work has been done, each one of the above issues addresses goals (in terms of data domain, target, and so forth) whose solutions are not suitable for use in other contexts [11]. For instance, interactive multimedia systems such as GettyGuide [7], are intended for use only inside the museums and the goals (e.g., help users finding locations of artifacts) are different with the ones of CiVeDi. For instance, GettyGuide, which is built for disseminating the contents of the J. Paul Getty Museum (www.getty.com) collection, proposes a customizable system to access information on art objects and galleries via kiosks using standard Web browser or via a location-based wireless handheld device. They do not support customized virtual tours of the content which is organized accordingly to the actual physical exhibitions.

With CiVeDi, we focus on virtual reconstruction of tours more than a virtual guide to be used in a museum. In this context, during the last few years many museums have invested into 3D virtual tours, and some of them today available on the web [12, 13]. Those are really painstaking jobs in terms of museum architectures and level of details, but often they do not allow any user interaction and are usually costly.

With CiVeDi, we aim at developing a low-cost system usable by many museums and at the same time independent from the collections of data they possess. Instead of a highly accurate model, we are interested in using VR as a tool for accessing multimedia databases. Based on our knowledge, such an approach is quite novel and the few approaches, already proposed by other researchers, are very limited and do not address many of the issues involved in combining database and VR techniques for personalized content exploration.

One of the first systems which aims at integrating VR and MMDBs is Virgilio [14, 15]. As CiVedi, Virgilio presents cultural context through virtual worlds in which some objects, situated in a VR environment, are used as metaphors to represent information. However, to use Virgilio, curators need to describe in advance the structure of the virtual tour by applying so called *Background Knowledge* predicates. Those predicates, used to map the structure of the content database onto the dynamic virtual environment, restrict the usability of the system to IT expert curators and cannot be used for sets of data distributed over different databases.

Another interesting system is the one built within the framework of the *Arco*[2] (Augmented Representation of Cultural Objects) project. CiVeDi and Arco have many points in common. They both organize tours into scenes (i.e., *Virtual Scene Unit* for Civedi and *Exhibition Space* in case of Arco) and create virtual tour using web-based 3D languages (X3D or VRML). Notwithstanding, the goals of the two projects are different because much of the effort of the Arco project has been devoted to make available large collections of 3D objects using a graphical interface. For instance, the Arco system provides an easy-to-use graphical user interface, called *Presentation Manager*, through which curators can easily manage (insert, move, group) media objects within the environment just like managing files within a file system. Such feature is outside of the scope of CiVeDi which instead focuses on creating virtual tours independently from the data contents and in which user can customize not only the contents but also the appearance and the duration of the visit.

7 Conclusions

In this paper, we have presented CiVeDi, a system to built customized virtual reality environments for disseminating cultural heritage information stored into MMDBs. The key features of the system are the possibility for the user of fully customizing contents, appearance and duration of the visit, a set of algorithms for dynamically arranging objects into the scenes, and the model we have developed as a foundation for managing virtual objects (i.e., VSUs, Db-driven and autonomous objects). The algorithms are very flexible and can be easily ported over any type of advanced 3D language beyond the one designed for the Web. At the same time, the data model we propose for creating objects can be used by third party modellers to constantly enhance the environment. Museums can create their own architectures or simply load the ones already available, possibly developed by other institutions or parties.

As a future work, besides an extensive test of the prototype system we have developed, we plan to make it possible to change the tour at runtime in case user modifies his/her choices. Even if built specifically for disseminating cultural heritage, CiVeDi can be used for many other different domains such as Virtual Shops. As a next step, we plan to use our solution within such domains to identify possible distinguishing requirements.

[2] http://www.arco-web.org

References

[1] A. Gaitatzes, D. Christopoulos, M. Roussou. *Reviving the past: Cultural Heritage meets Virtual Reality*. in the Proceedings of "Virtual Reality, Archaeology and Cultural Heritage. Athens, Greece, 28-30 November 2001

[2] G. Scali, M.Segbert B. Morganti. *Multimedia applications for innovation in cultural heritage: 25 European trial projects and their accompanying measure TRIS*. In 68th IFLA Council and General Conference August 18-24, 2002

[3] Dr. James Jones, Mark Christal. *The Future of Virtual Museums: On-Line, Immersive, 3D Environments*. Created Realities Group (c)2002

[4] Archeoguide: http://archeoguide.intranet.gr/project.htm

[5] 3Dmurale: http://www.brunel.ac.uk/project/murale/home.html

[6] Thetron: http://www.theatron.co.uk

[7] David S. Marshak. *J. Paul Getty Museum Re-Architects Technology to Enhance Visitors. Experience.* Patricia Seybold Group Case Study, June 2003.

[8] Politou E., Pavlidis G., Chamzas C., *JPEG2000 and dissemination of cultural heritage over the Internet.* IEEE Transactions for Image Processing, accepted for publication

[9] T. Catarci. *What Happened when Database Researchers Met Usabilit*. Information Systems, 25(3): 177-212, 2000.

[10] C. Lewis, P. Polson, C. Wharton, and J. Rieman. *Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces* Proceedings of the SIGCHI conference on Human factors in computing systems, 1990

[11] Augusto Celentano *Virtual Worlds as Metaphors for Web Sites Exploration: Are They Effective?* IEEE Symposium on Visual Languages, September 13 - 16, 1999

[12] MUVA. Museum virtual the artes el pais. http://www.elpais.com.uy/muva2/

[13] Van Gogh 3D Exhibition. http://www.vangoghmuseum.nl/

[14] Massari, A., Saladini, L., Hemmje, M. and Sisinni, F. *Virgilio: A NonImmersive VR System To Browse Multimedia Databases.* Proceedings of the IEEE International Conference on Multimedia Computing and Systems, IEEE Computer Society Press, Los Alamitos, CA, 1997

[15] M. F. Costabile, D. Malerba, M. Hemmje, A. Paradiso, *Building Metaphors for Supporting User Interaction with Multimedia Database.* Proceedings of 4th IFIP 2.6 Working Conference on Visual DataBase Systems - VDB 4, L'Aquila, Italy May 27-29 1998.