3D MURALE: Multimedia Database System Architecture

Edward Grabczewski, John Cosmas, Peter Van Santen
Damian Green, Takebumi Itagaki, Fred Weimer
edward.grabczewski@brunel.ac.uk

Brunel University, Uxbridge, Middlesex UB8 3PH, United Kingdom

Abstract

Archaeological databases are required to store a wide range of data about archaeological objects. Multimedia, spatial and temporal requirements are placing new demands on these databases. Virtual models of archaeological sites require new storage and search facilities, including searching of 3D graphics for virtual and physical restoration of archaeological finds. We examine the architecture, design philosophy and proposed implementation of the 3D MURALE multimedia database, which will be used by archaeologists to construct a virtual model of the Sagalassos excavation site in Turkey.

KEYWORDS: Archaeology, Multimedia Databases, Virtual Reality, Photogrammetry.

1. Introduction

This paper describes the 3D MURALE multimedia database system architecture, design philosophy and proposed implementation. 3D MURALE is a project associated with the archaeological excavation site at Sagalassos in Turkey [5]. The goals of MURALE are to digitally record, store, restore and visualise those archaeological objects found at the Sagalassos site; this includes the site as a whole.

Our paper is structured as follows: the following section gives a brief account of the history of Sagalassos and the archaeological objects found. Section 3 outlines the process of archaeological excavation. This is followed by section 4, describing the database system requirements and design philosophy. Section 5 examines the system hardware and software. Section 6 looks at the proposed database system architecture. Section 7 outlines the methods used to digitally record a variety of architectural objects and section 8 shows a partial logical model for storing archaeological data about those objects. Section 9 discusses the query facilities required by multimedia database applications requires. Section 10 describes how virtual objects are visualised with the database. Data integrity and backup issues are explored in section 11. And finally, our concluding section discusses current work in progress.

2. History of Sagalassos

Sagalassos was an ancient city situated in southern Turkey, in what used to be the region of Pisidia – now corresponding roughly to the modern Turkish provinces of Burdur and Isparta. The city lies among the Taurus mountains and valleys of Burdur, about 109 km north of the coastal town Antalya, near the village of Aglasun. Sagalassos ranges in altitude from 1490 to 1600 m above sea level, the site lying on a steep limestone front. The earliest literary sources [11] date Sagalassos to before 333 BC. Livy describes the Sagalassians as being the best warriors of the region, with a large population and a rich, abundant territory. The next few centuries saw a spectacular growth in the city and its ceramic industry, issuing its own coins and erecting a number of monuments, buildings and fountains. Shortly after 400 AD the city fell into decline, followed by two earthquakes in 518 and 528 AD that probably damaged the aqueduct system. A plague struck the region in 541 – 543 AD killing possibly half the population. This was followed by another earthquake around 650 AD. By the middle of the seventh century Sagalassos was abandoned; some of its inhabitants resettled in Aglasun. The isolated location of the site probably saved the buildings from plunder; they were left to stand and gradually decayed or collapsed. Eventually a thick layer of erosion material covered them. In 1706 a French diplomat, Paul Lucas, saw the site but failed to identify it, leaving only a written record of his encounter. The British chaplain Francis Arundell visited the site in 1824 and realised that it was Sagalassos. Research on the site started in 1884 by the Polish count Karol Lanckoronski, who eventually published a survey in 1892. A variety of expeditions took place until 1985 when Stephen Mitchell resurveyed the site. The Belgian archaeologist Marc Waelkens joined the excavation in 1986 and is currently Director of the Sagalassos site [6].

3. Archaeological Excavation

Archaeological excavation is one means by which archaeologists answer questions about human history. Firstly an archaeologist poses some questions about a site. Next, a research design produces a project plan showing how to answer those questions. Once the plan is approved and funded, a sampling strategy is devised to collect samples that will provide the required evidence and data. A site survey draws up a plan of the site. Excavation then leads to the discovery of physical archaeological objects. These objects are recovered and recorded according to accepted archaeological practises. The compiled data on these objects is analysed in preparation for archiving, reporting and restoration. Restoration comprises two parallel processes - reconstruction and hypothesising. Reconstruction is analogous to building a jigsaw from the *available* pieces whereas restoration involves filling-in any gaps with hypothetical parts. The whole object can then be displayed in a museum or site [8].

The procedure just described is the traditional manual process or archaeological excavation. Figure 1 shows how the procedure is modified to take into account the virtual recording and modelling processes we propose to develop for MURALE. This overview shows the basic data flows necessary for our virtual archaeological excavation system. With this in mind we can create a logical model for the multimedia database.

4. System Requirements and Design Philosophy

The MURALE system has a fairly comprehensive range of multimedia database requirements. Firstly, it must store a variety of multimedia data types, including conventional text and numbers, images, video and 3D graphics. The semantic model must include both spatial and temporal dimensions, allowing time-based spatial modelling of the virtual Sagalassos site. Multimedia data can be stored against archaeological objects such as stratigraphic units (deposits and interfaces), buildings, friezes, statues, sculptures, pottery, sherds (pottery fragments) and coins. The database system must provide query by attribute and query-by-example tools for both image and spatial objects. Using annotated links, it should be possible to retrieve data associated with virtual objects highlighted in a VRML scene. Media content must be described using MPEG-7 content metadata.

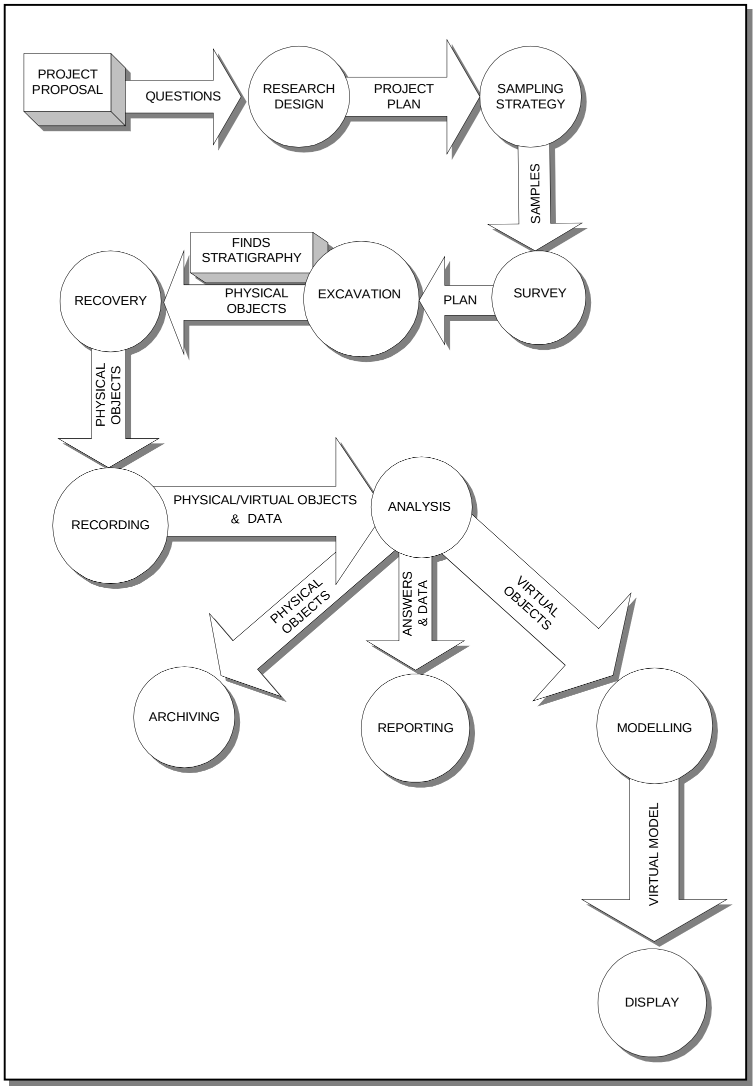

Figure 1. Virtual Archaeological Excavation

Our design philosophy is to use existing open technologies. We use modular PC hardware components, open source software and conform to international standards wherever possible. Failing this we write bespoke software and extend existing standards or define our own. Bespoke software is initially written in Java, however performance critical code may be rewritten in C++. The imposed standards to which we must conform are CIDOC, MPEG-7 and VRML. Additional standards include CORBA, Unicode, XML, PNM (Portable aNy Map) for uncompressed images and XHTML.

There are some practical constraints imposed on the MURALE system design. The first of these is to create design solutions that are generic and applicable to other archaeological sites. The second is to implement a system that other archaeologists can typically afford. The third is to investigate the possible use of distributed technologies for the MURALE database.

Each year, during an intensive two-month excavation, up to 10,000 new archaeological objects are recovered. The database must have adequate storage to record this volume of data in whatever form it appears and for many years to come. Our solution to this problem is to choose a SCSI-based, hardware RAID subsystem for the multimedia database.

By using open source software it is possible to divert most of the financial resources towards the hardware. We avoided two performance bottlenecks – disk and network access – by spending more of our budget on these subsystems. The fast network access allows us to experiment with CORBA and Jini technologies.

Our approach to conceptual modelling, whilst conforming to a variety of software standards, is to firstly create a logical model using established Entity-Relationship modelling techniques and then to modify it according to CIDOC, MPEG-7 and VRML standards. This puts the design emphasis firstly on our application and secondly on the standards. We do this partly because current CIDOC standards are incomplete, partly because we feel the application should benefit from the most appropriate design.

5. Database System Hardware and Software

The hardware for the MURALE database comprises modular PC components from established manufacturers and built by a local PC supplier. There are three bespoke servers, each comprising a large tower case with six internal fans and a dual processor ASUS motherboard holding two 1 GHz Pentium III processors. Each machine has 512 MB of onboard RAM, expandable to 4 GB.

The disk subsystem for each server comprises an Adaptec 3200S SCSI RAID controller card with 140 GB of disk space in a RAID-5 configuration, expandable to 2 TB. The card includes battery backup for the extended 128 MB cache. Two extra data channels can be added to the controller, supporting up to 4 TB of storage in our case. Note the maximum storage capacity is determined by the size of the SCSI disks; all disks should be identical in a RAID-5 configuration. Our disks are 73 GB each, however the latest have a capacity of 250 GB, giving a maximum storage potential of 15 TB. The random access bandwidth is approximately 10 MB/s.

The network subsystem for each server is a 3Com Gigabit Ethernet card running at speeds of 1000/100/10 bps. The practical bandwidth is expected to be around 50 MB/s.

The backup subsystem is a Tandberg DLT 8000 SCSI tape drive containing a 40 GB DLT tape cartridge. The bandwidth is at least 6 MB/s.

The software for the MURALE database is open source and comprises Red Hat Linux 7.1, PostgreSQL 7.1, CORBA 2.3 (JacORB and TAO ORB), Java JDK 1.3, Jini and the Apache 1.3 Web server. We also use a variety of XML based tools.

The PostgreSQL database has been chosen because it is one of the most advanced and well-supported *open source* object-relational databases. Its development and features are well documented by Stonebraker [16] and Momjian [17] respectively. Some of the rarer features for an open source database include transaction support, BLOB support, R-tree indexing, triggers, internationalisation and a wide range of programming interfaces. Some of the rather unusual features include temporary tables, table inheritance and a range of geometric data types and operators. For example, the data types POINT, CIRCLE and POLYGON, with their corresponding operators and functions. PostgreSQL allows the user to extend the SQL engine to define new data types and operators. However, it does not currently support tablespaces and there appears to be no documented case of a terabyte database - MURALE may well push it to its limits. We are currently investigating the use of table inheritance in our CIDOC implementation strategy.

6. Database Architecture

Figure 2 shows the layered architecture of the 3D MURALE multimedia database system. The underlying network layer uses CORBA (and optionally Jini) to provide distributed services. The presentation layer uses XML and XSL to encapsulate and format transmitted data to and from an application. The application layer contains the MURALE database. This layer comprises an extended CIDOC data model with support for MPEG-7 content description. All data inputs to the database are received in XML format. All data outputs are converted to XML format and optionally transformed by XSL into one of the supported output file formats, namely XHTML, PDF, PNM and VRML. Compressed images for Web pages are stored and output in JPEG format.

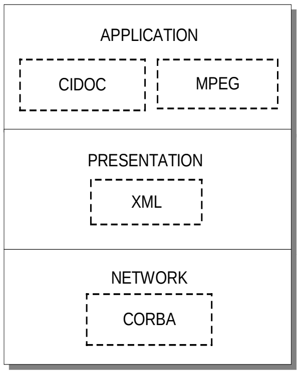

Figure 2. Layered Architecture

The CORBA *relationship* service was considered for the implementation of a distributed database design, however it is difficult to find an implementation of this service in current ORBs. In retrospect, the CORBA community finds this service too fine-grained to be of practical use. Relying on distributed processing at *row* level granularity makes database joins excessively slow compared to distributed databases working at *table* level granularity. Far higher network speeds (an order of three or more) would be required to make performance comparable.

MURALE needs a replication service to copy data in database tables between servers. This is not a standard CORBA service however. To widen our choice of services at the network level we have included Jini in our portfolio. Jini does not have a replication service but includes the *JavaSpaces* service.

There are good reasons for choosing a distributed or replicated system design. These are to do with the practical and logistical problems of taking a computer to an archaeological site for a few months each year. Firstly, there is the risk of damaging disks in transport. Secondly, there are often security concerns in transporting computers across borders, or leaving them at the excavation site when the season ends. Thirdly, there is the risk of disk failure in hot climates. Finally, there is the problem of power supply stability and potential system damage. These are all good reasons for *not* installing local servers at the site.

We considered two options: **a)** to have an off-site database server; **b)** to replicate data from an on-site database server periodically to an off-site database sever. Both options minimise the risk to stored data, however PostgreSQL does not have a built-in replication service. Also, option (a) would require the site to have a high bandwidth network connection for transport of video and 3D graphics. Currently Sagalassos does not have Internet access; this may be rectified next year but will probably be low bandwidth. We currently use a centralised database architecture.

7. Recording Archaeological Objects

Digital data on archaeological objects is captured in a variety of ways. The conventional method of capturing terrain and stratigraphic data manually is to use a Total Station - housing both a laser theodolite and Electronic Distance Measurement (EDM) equipment. This allows 3D points at the site to be recorded at a maximum rate of about one per minute. The data is stored in the memory of the Total Station and later downloaded to a computer, where it is further processed to create a Digital Terrain Model (DTM). This model is usually output in DXF format to a 3D Graphics Workstation such as the Bentley *MicroStation* or a PC running *AutoCAD*. Such models are then converted and saved to VRML format.

These conventional techniques capture relatively small amounts of data by a largely manual process. The MURALE project aims to increase both the precision and density of data capture using newer methods such as photogrammetry. Photogrammetry is used in MURALE to capture DTMs by calculating three-dimensional points from two or more images. These images must be separated by five or more degrees, their calculated perspective projection is used to find the position of any point on the surface of the subject [14]. Such calculations are done automatically by software, generating an accurate 3D model as output. VEXCEL's *FotoG* is a similar commercial product and is used in MURALE.

There are other techniques used in MURALE to generate 3D models from objects such as statues, buildings and pottery. A tool called *ShapeSnatcher* uses a structured-light principle to project a grid onto the subject, from which a 3D surface is generated from only a single digital image. This tool is typically used to record pottery. *Shape from Video/Photo* is a MURALE tool in development, capable of using digital video to generate 3D models using a structure-and-motion algorithm. This tool is typically used to capture terrain, strata and buildings.

We do not simply store these 3D models as individual BLOBs (Binary Large OBjects) in the database. This would make the spatial querying of data far too slow and complex. Our goal is to post-process all the data for these 3D models and to construct a single spatial model of the Sagalassos site in our database.

Sagalassos archaeologists currently record new deposits uncovered in excavation units by measuring the depth at each corner of the excavation unit down to the surface of the new deposit. This gives them four points on a plane surface from which they calculate the volume representing the deposit by taking the volume difference between the surface of the current deposit and that of the previous deposit.

The digital technique of photogrammetry does not record volumes; instead it records accurate surfaces. If we wish to know the volume of a deposit recorded using photogrammetry we must take the volume difference between the surface of the current deposit and that of the previous deposit – both surfaces bounded by the sides of the excavation unit.

Despite the additional data processing, the advantages of digital recording are the higher density of recorded points and the correspondingly greater accuracy of the site model.

8. Modelling Archaeological Objects

Archaeological objects form two broad categories – stratigraphic units and finds. Note that *deposits* are often called layers or strata and *finds* are often called artefacts.

According to Harris [9], *stratigraphic units* are subdivided into the following sub-categories:

Stratigraphic Units:

- **Deposit units:**
  - *Natural deposit*
    - Horizontal (e.g. mud from floodwaters)
  - *Man-made deposit*
    - Horizontal (e.g. road surface, filling in a pit)
    - Upstanding (e.g. walls)
- **Interface units:**
  - *Layer interface*
    - Horizontal (e.g. top surface of a deposit)
    - Upstanding (e.g. re-decorated wall)
  - *Feature interface*
    - Horizontal (e.g. exposed portion of a wall)
    - Vertical (e.g. pit, postholes, ditches)

Stratigraphic Units

When a site is excavated, its stratigraphic record is destroyed. It makes sense therefore to record every significant detail, including all deposits and interfaces. Geometrically speaking, a *deposit* may be viewed as a volume and an *interface* viewed as a surface. Each carries information necessary to fully model a site. The stratigraphic record of a site is modelled by storing data about the stratigraphic units in a database. This data is normally drawn as a Harris Matrix - encapsulating the four laws of stratigraphy. Every stratigraphic unit must be identified and recorded according to this classification system. For example, site terrain is classified as a *horizontal layer interface*. Gradually a Harris Matrix is built up, showing inter-relationships between the site deposits and interfaces [15]. Until recently, Sagalassos archaeologists only recorded deposits, not interfaces, on their Harris Matrix. This practice is being reconsidered as the new recording techniques provide more interfacial data about the site.

Finds are objects such as statues, buildings, walls, friezes, columns, pots, coins and possibly fragments of these. Each object has attributes as well as associated multimedia. *Systematics* is the field of organising unique objects into classes or groups. *Typology* is the process of extracting meaningful relationships from these systemic collections [1]. The MURALE database must capture all such attributes, ready for querying using textual tags. However, the requirement that all solutions be “generic” places a burden on the database designer. Each archaeological team appears to store a slightly different set of attributes. Even when the same finds are stored, different archaeological specialists may use different systemics. One solution to this problem is the standardisation of systemic categories. Another solution is to have a dynamic database create or extend its own logical model based on a supplied XML schema. This would allow data to be transported quickly between databases.

The MURALE requirement to conform to the CIDOC, MPEG-7 and VRML standards results in a complex logical model that may be time-consuming to copy to another multimedia database. Once again, the possibility of exporting the logical model to an XML schema, and vice-versa, would provide a portable means of transporting data between databases.

Although it is too early to publish a complete logical model of the MURALE database, partial models are being tested with new applications that process the first batch of 3D graphics recorded this summer at Sagalassos. These partial models will be integrated and modified to conform to the various standards.

Figure 3 shows a partial logical model for an archaeological site. We take the definitions of an *archaeological item* and *physical space* from the CIDOC standard [2]. Most of the entities are familiar to a field archaeologist. A PROJECT may have many SITEs and each site must have a GRID defined over its surface – like lines on graph paper. Each grid creates many squares or SECTORs comprising several POINTs. Every site grid has at least one site DATUM - a reference point whose co-ordinates are known. An excavated site contains many EXCAVATION UNITs - areas that the archaeologists have dug. Often, excavation units neatly correspond to the site sectors. Sometimes the excavation units cross sector boundaries – in which case one excavation unit belongs to many sectors. Sometimes a sector will contain many excavation units.

Excavation units usually contain a variety of ARCHEOLOGICAL OBJECTs. These may be grouped into COLLECTIONs, such as a collection of pottery fragments from the same pot. Each archaeological object comprises a set of points, recorded using the techniques mentioned in section 7. The NODE entity is derived from the VRML standard. We model a tree of nodes containing

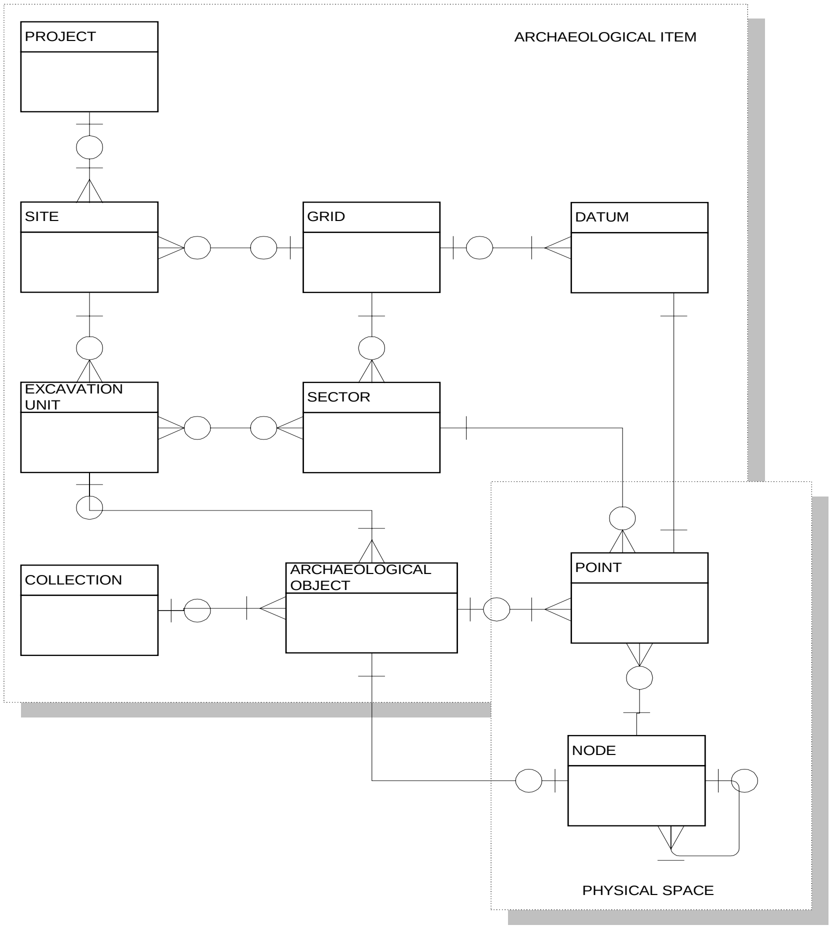

Figure 3. Logical Model

data about a virtual scene. These nodes sometimes refer to archaeological objects – whether real or hypothetical. Whereas real objects are based on recorded data, hypothetical objects are created by 3D modelling software during virtual restoration. These hypothetical objects fill gaps found in real object data. For example, a real model of the statue of Venus may have hypothetical arms.

9. Querying Archaeological Objects

MURALE requires that text queries are performed on the following object attributes: annotations, location, period, deposit and level-of-detail. Furthermore, image queries must search on colour, texture, shape and spatial relationships. Finally, 3D graphics queries must search on shape and spatial relationships. We know of no databases that permit searches of 3D graphical objects thorough the medium of a query language, however, work on spatial searching of 3D medical databases has been done by Keim [10].

Image and 3D graphics queries can be implemented using an extension of the SQL syntax by defining new functions and data types. Subrahmanian [12] describes the principles of adding functions to standard SQL, permitting queries on features, attributes and values. The syntax of queries can be simplified by creating an XML based query language. Furthermore, PostgreSQL supports R-tree indexing for spatial queries, as well as B-tree and hash indexing.

Virtual timelines of the site excavation and site history are created by performing temporal queries against the archaeological objects in the database. Each object must have an excavation date and a creation date where possible. The PostgreSQL DATE, TIMESTAMP and INTERVAL data types are sufficient for the Sagalassos site but there is a problem with creating a sufficiently generic application using the DATE type, which ranges only from 4713 BC to 32767 AD [13]. These limits are too narrow for some archaeological applications, such as the Palaeolithic site at Combe-Capelle in France [7].

10. Visualising Archaeological Objects

We pointed-out in section 7 that all recorded data for archaeological objects are stored as discrete points inside the database. This is not an optimal solution for the purposes of visualisation, which normally deals with polygons, not points. Therefore the data needs to be retrieved and pre-processed to create a VRML scene.

There are two ways of creating VRML scenes, dynamic and static. The *dynamic* method extracts live data from the database and creates a VRML formatted stream before sending it to a client program. The *static* method pre-processes a VRML scene and stores the resulting file in the database as a BLOB, ready for retrieval by a client program.

The MURALE database is unlikely to have the processing capabilities to generate *dynamic* data streams without additional technology. One MURALE partner is currently developing a VRML streaming system that supports levels-of-detail (LOD). This approach embeds different resolution 3D models in a VRML stream depending on network bandwidth. Currently we generate *static* files to visualise VRML scenes. Figure 4 shows the manner in which XHTML can be generated dynamically and displayed. This example includes a VRML hyperlink that automatically calls a VRML plug-in to display a scene. MURALE requires both high-resolution and low-resolution models to visualise virtual archaeological sites.

11. Data Integrity and Database Backup

We have some concerns about storing data in the MURALE database. Firstly, the data must be scientifically accurate. Checks must be made to confirm and maintain the accuracy of this data periodically by checking the calibration of the recording equipment. Secondly, the data must not be contaminated. Some virtual models in the database are based on real objects whereas some are hypothetical models. They must not be confused, otherwise years of excavation work and scientific data could be rendered useless. During MURALE development it is prudent to keep all site paperwork.

Of course the MURALE database needs regular backups.

Currently we have a 110 GB filesystem per server. The Tandberg DLT 8000 streams data at an uncompressed rate of 6 MB/s, backing up 110 GB in approximately five hours. Should we expand our storage capacity close to the maximum of 2 TB we would require approximately 104 hours (4.3 days); clearly a different backup system would be required in this case, especially for a busy excavation site. The practical limit for any such backup is probably 24 hours. In addition, DLT tapes must be stored at close to room temperature. This is yet another reason against on-site database servers and an argument for using client/server or distributed model solutions.

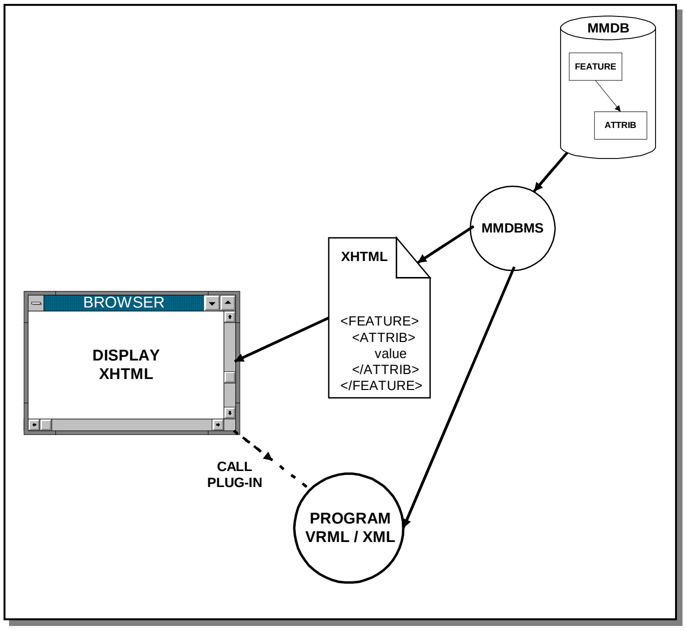

Figure 4. Dynamic XHTML

12. Conclusions

We have examined the relationship between archaeological processes and how they map to database requirements in the light of changing recording and visualisation techniques. The new types of multimedia objects stored in databases have an influence on implementation issues, in particular those of distribution, replication, recording, storage space, querying and visualisation. We note that some traditional procedures, such as the recording of only deposits, will probably change and are indeed supported by the new technologies. We saw that some potentially useful technologies, such as the CORBA relationship service, are not currently useable in our design. Work is in progress on the development of logical models - including spatial and temporal models - and conformance to CIDOC, MPEG-7 and VRML standards. Two areas of particular research interest include queries on 3D objects and logic model generation using XML schemas.

13. Acknowledgements

The authors gratefully acknowledge the support of the EU, funded under the IST program, for project IST-1999-20273, 3D MURALE (*3Dimensional Measurement and Virtual Reconstruction of Ancient Lost Worlds of Europe*).

14. References

[1] E. B. Banning. *The Archaeologist's Laboratory: The Analysis of Archaeological Data*, chapter 3, Kluwer Academic / Plenum Publishers, New York, USA, 2000. ISBN 0-306-46369-5.

[2] CIDOC. *CIDOC Core Data Standard for Archaeological Sites and Monuments*. ICOM/CIDOC, chapter 2, http://www.cidoc.icom.org/pub1.htm. 1995.

[3] CIDOC. *CIDOC Relational Data Model: A Guide*. ICOM/CIDOC, http://www.cidoc.icom.org/pub1.htm. 1995.

[4] CIDOC. *Definition of the CIDOC object-oriented Conceptual Reference Model*. ICOM/CIDOC, http://www.cidoc.icom.org/pub1.htm. 1999.

[5] John, Cosmas, Take Itagaki, Fred Weimer, Eddy Grabczewski and Damian Green. *3D MURALE: A Multimedia System for Archaeology*. VAST 2001 Conference, Athens, 2001.

[6] Roland Degeest. *The Common Wares of Sagalassos*. Brepols Publishers, pp.29-37, Belgium, 2001. ISBN 2-503-50797-2.

[7] Harold L. Dibble, Shannon P. McPherron and Barbara J.Roth. *Virtual Dig: A Simulated Archaeological Excavation of a Middle Paleolithic Site in France*, Mayfield Publishing Company, California, USA. 2000. ISBN 0-7674-0245-6.

[8] Peter L. Drewett. *Field Archaeology: An Introduction*, UCL Press, London. 1999. ISBN 1-85728-738-X.

[9] Edward Harris. *Principles of Archaeological Stratigraphy*, Academic Press Limited, chapters 6 & 7. London. 1989. ISBN 0-12-326651-3.

[10] Daniel A. Keim. *Efficient Geometry-based Similarity Search of 3D Spatial Databases*, SIGMOD 99 Philadelphia PA Conference Proceedings, pages 419-430. ACM-SIGMOD, May 1999.

[11] Livy (59 BC – 17 AD). *History of Rome*, 38.16.9.

[12] V. S. Subrahmanian. *Principles of Multimedia Database Systems*, Morgan Kaufman Publishers, Inc., San Francisco, California, USA, 1998. ISBN 1-55860-466-9.

[13] C. Tondering. *Frequently Asked Questions About Calendars: Parts 1, 2 and 3*, claus@tondering.dk, www.faqs.org/faqs/calendars/faq. 2000.

[14] M. Pollefeys, R. Koch, M. Vergauwen and L. Van Gool. *Metric 3D Surface Reconstruction from Uncalibrated Image Sequences*, Proc. SMILE Workshop, LCNS 1506, pp.138-153, Springer-Verlag. 1998.

[15] Edward Harris. *Principles of Archaeological Stratigraphy*, Academic Press Limited, pages 29-39. London. 1989. ISBN 0-12-326651-3.

[16] Michael Stonebraker. *The POSTGRES Next-Generation Database Management System*, Communications of the ACM, 34(10), pages 78-92. Association for Computing Machinery Inc. 1991.

[17] Bruce Momjian. *PostgreSQL: Introduction and Concepts*. Addison-Wesley. 2001. ISBN 0-201-70331-9.