AN e(X)TENSIBLE AND MODULAR HISTORICAL DOCUMENTATION MODEL: THE "GeoHistoryPortal"

C. Guney [a], L. Thys-Senocak [b], N. Ulugtekin [c], D. Tomlin [d], R. N. Celik [e]
[a] IGS-ISTA Satellite Observation and Processing Laboratory, in the Division of Geodesy at Istanbul Technical University, 34469, Istanbul, Turkey, guneycan@itu.edu.tr
[b] The Department of History at Koc University, Istanbul, Turkey, lsenocak@ku.edu.tr
[c] The Division of Cartography at Istanbul Technical University, 34469, Istanbul, Turkey, ulugtek@itu.edu.tr
[d] The Department of Landscape Architecture at the University of Pennsylvania, Philadelphia, PA 19104-6311, USA, tomlin.dana@verizon.net
[e] The Division of Geodesy at Istanbul Technical University, 34469, Istanbul, Turkey, celikn@itu.edu.tr

KEY WORDS: Documentation, E-Heritage, Internet/Web, Data Structure, Visualization

ABSTRACT

Building an effective information system requires a vision of business technology which involves detailed plans and an effective architecture drives the project. This paper shows the technical components of the business technology vision of GeoHistoryPortal, an exclusive cultural heritage documentation *Web portal for two 17th century Ottoman fortresses on the Dardanelles in Turkey*, and describing how GeoHistoryPortal v1.0 is being designed, developed and implemented using Web service technologies in order to share project's motto *"the life history of two Ottoman fortresses on the Dardanelles"* with users. The major purpose of this paper is to present the GeoHistoryPortal's comprehensive and sophisticated framework for data encoding, transferring and presentation on the Web. Here, what would otherwise serve primarily as a data source has been expanded into a communication hub and a collaborative analysis tool for project members and authorized users alike. GeoHistoryPortal v1.0 uses the e(X)tensible Markup Language (XML) as the basis for a new *cultural heritage documentation language* intended to facilitate the exchange of information in a wellstructured yet highly flexible manner. This study describes how XML and offshoots can be used in Spatial Information System (SIS) settings in general and cultural heritage documentation projects in particular. The paper also demonstrates that recent innovations in Internet-based information technologies hold enormous promise for the understanding, promotion, and preservation of cultural heritage.

1.INTRODUCTION

An effective Web portal must offer a broad array of information in a manner that provides for convenient user access.

GeoHistoryPortal, which seamlessly integrates tremendous content with many outstanding features, is a gateway (common user interface) to cultural heritage documentation project of two 17th century Ottoman fortresses located at the entrance to the Dardanelles in Turkey: *Seddulbahir* and *Kumkale*. It is intended that visitors participate in the project's motto "*life history of the fortresses*" and that a graphical user interface (GUI) enables project staff and other authorized users to manage the multiparticipant SIS project and make the decisions pertaining to the project's motto.

GeoHistoryPortal is not only virtual identification employed in the project for public relations (PR) purposes, but this innovative cultural heritage documentation Web portal also collects internet applications and entire information in one united environment with eXtensible, modular, and dynamic key features to enhance productivity and ease-of-use. Afterwards, it distributes instant & exhaustive information and documentation pertaining to the project back to associated user(s). eXtensible Markup Language (XML)-based this information is presented to users via a Web browser through GeoHistoryPortal on the Web. It is considered as a base of operations for the project staff on the Web.

XML, the lingua franca of Web services, is a Web standard developed by the World Wide Web Consortium (W3C) and is the fastest evolving technology for Web applications. It is a standard for describing, storing and transporting data across the Web. [URL 16] The Internet is the hot topic for delivering information and GIS applications as it is playing a central role in the exchange of information. Markup languages open up new opportunities to build a semantic context model and to specify and exchange ontologies.

GeoHistoryPortal relies on XML as a proposal for a cultural heritage documentation language to make disparate pieces of information Web-accessible in a common form. As such, it is presented as a prototype for the exchange of spatial information through a semantic context model.

GeoHistoryPortal utilizes open-source technologies (PHP, Apache with Tomcat, MySQL with phpMyAdmin) with XML in a Fedora Linux environment to establish its fat server (Web + application + database servers) and services. Using these technologies within enterprise architecture, powerful Webbased client/server computing environment was deployed to build up engineering rich internet applications.

2. BUSINESS MODEL

GeoHistoryPortal, is a multi faceted and multi dimensional cultural heritage documentation Web portal and its elemental implementation strategy is comprised of the following.

1. Web designing
   - Determining goals & target audience
   - Generating usable content
   - Developing attractive graphic design
2. Web building
   - Developing Web services/applications/modules
   - Hosting (registering)
   - Broadcasting
   - Testing
3. Web managing
   - Optimizing
   - Updating content

GeoHistoryPortal serves up its valuable content pertaining to two Ottoman fortresses through a fascinating graphic design considering that the design of a portal has to be flexible enough to meet diverse users' and visitors' needs, yet structured enough to accommodate a wide range of content and visual elements.

GeoHistoryPortal v1.0 provides two distinctive phases in terms of concept: general and intra project content. These two separate sets of capabilities are encompassed for internal and external use. Figure 1 outlines this design philosophy.

GeoHistoryPortal v1.0 has been developed as a Macromedia Flash-based website with Action Script using Macromedia Flash MX 2004 (with Action Script 2.0) and the eXtensible Hyper Text Markup Language (which extends HTML by making it XML compliant) with Cascade Style Sheet and XML (XHTML 1.0/CSS 2.0/XML). It has been designed for at least 1024x768 screen resolution, tested using Mozilla Firefox 1.0.4, and is best viewed with 1400x1050 resolution. GeoHistoryPortal v1.0 uses XML with Flash MX and CSS to define the content to be displayed, since XML is only a semistructured document that holds content. XML can be transformed at runtime into a presentation markup language, such as HTML or XHTML.

A device/platform independent e-documentation interface for GeoHistoryPortal v1.0 can be visited at www.seddulbahir-kumkale.org via any Web browser with Macromedia Flash Player 7.0.

Macromedia Flash is a fantastic tool for developing remarkable small size animations & presentations and an effective interaction tool for website visitors. [URL 7]

3. DELIVERING INFORMATION OVER THE WEB

The efficient transfer of information is as much dependent on business structure as on enabling technology. Client/server computing is emerging as the most appropriate computing technology in the business world. Similarities between this computing paradigm and the way many of today's businesses operate are so clear that a large number of corporations are migrating to client/server technology.

Recent advances in World Wide Web (WWW) and Database Management System (DBMS) technology make it possible to combine the best features of each to provide client/server DBMS applications over the Internet. The result is Web-based client/server computing, which sends and receives dynamic data over the Internet by creating HTML files on the fly.

In this project, Web-based client/server computing has been implemented using Thin Client model that enables conversion of client/server applications to the Web. Thin Client makes it possible for all business processing to be performed in a controlled environment on a managed server (fat server). It also eliminates the requirement that clients download additional software in order to use the site. As shown in Figure 2, this client/server computing architecture enables large number of users to easily and efficiently access information from all over the world.

At the core of this architecture is a language involving a modular data encoding system that enables data sets at various levels of detail (LOD) to be handled in a structured manner. A highly formalized language able to avoid ambiguities and misunderstandings in data is required in order to make use of a markup language as a tool like XML for documenting, querying, analyzing and interpreting the historical features of the fortresses in hypertext format.

3.1 Detailed Planning of Multi-Tier Client/Server System and Applications

The system architecture and physical infrastructure of this project comprise a Web-enabled multi-tiered (this may also be referred as an n-tier architecture) solution which based on process-driven computing model. Key aspects of this distributed enterprise model include the number of servers, the role each server plays in the solution, the relationship between servers, and how the multiple servers collaborate together to meet system-level operational requirements. The model can be envisioned as a set of tiers. This four-tiered distribution is listed below in process order:

1. Client tier
2. Web tier
3. Application tier
4. Data tier

Open-source and affordable products were selected based on the cost and the need to support a large community of users.

3.1.1 Tier-1: Thin/Minimal Client: Web-based client/server computing uses a standard Web browser to access and manipulate dynamic information over the Internet. This is an alternative to the traditional client/server model in which a custom-written GUI application accesses and manipulates dynamic information stored in a DBMS using proprietary communication software. Each instance of the client software Web browser communicates with Web server as a requester of services. Web-based client/server computing generates HTML pages on the fly to provide the latest information via the WWW.

3.1.2 Tier-2: Web Server System: As a provider of services, Web server exclusively handles HTTP protocol and HTTP requests in order to make Web pages viewable in Web browsers. It is the Web Server that hosts both the components of a Web page (such as HTML files, CSS files and templates) and all other essential website functions.

The current Web server software in this project is Apache 2.0.54, developed and maintained an open-source HTTP/1.1 compliant Web server by the Apache Software Foundation. It is available under "GNU's Not Unix General Public License (GNU GPL)" for non-commercial purposes.

In this project, the Apache httpd server runs on the Fedora Project Core 3 Linux operating system (OS). Fedora Project is an open source project sponsored by Red Hat and supported by the Fedora community. Its goal is to work with the Linux community to build a complete, general purpose operating system exclusively from open source software. [URL 4]

The Apache Web server is connected to a MySQL database (tier 4) by way of a common gateway interface (CGI) program called "Hypertext Preprocessor (PHP 5.0.4)". PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. [URL 10]

3.1.3 Tier-3: Application Server System: An application server (app server) is a component-based product that resides in the middle-tier of a server-centric architecture in order to provide business logic for client applications to application programs through various protocols. The software installed on an application server facilitates the serving (running) of other applications in a highly flexible and scalable Java-based Web services capability.

In this project, Java 2 Platform Enterprise Edition (J2EE) Application Server with Apache Tomcat is used as middleware to build Web applications and services, *i.e.* Web mapping, interactive mapping. Apache Jakarta Tomcat 5.5 is a Java Servlet and J2EE JavaServer Pages (JSP) Web container. Tomcat is developed in an open and participatory environment and released under the zero-cost Apache Software License. [URL 13]

In this project, Tomcat has been deployed as an out-of-process servlet container to the Apache Web server. Thus, the Tomcat engine runs in a separate process on application server, different computer than the Apache HTTP server. Communication between these two different farms is provided with TCP/IP protocols by way of TCP/IP sockets connection. Running out-of-process allows more scalability, flexibility and reliability.

3.1.4 Tier-4: Database Server System: The underlying tier is a DBMS that stores the data required by the middle tier.

In this project, the current database server software is AB's MySQL 5.0 with MySQL network (MySQL Database System Version 5.0.4-beta) running in a Fedora Linux environment. MySQL runs on the database server and can be operated over the Internet using phpMyAdmin 2.6.2-pl1 (patch level one).

MySQL relational database management system (RDBMS) is a Web-enabled, open-source, multi-threaded, data and information management software. Its architecture and easy administration make database-driven applications extremely fast and easy to customize in the decision-making processes. MySQL also includes support for a subset of SQL92 with the geometry types environment proposed by the Open GIS Consortium. This allows for efficiently storage and manipulation of spatial data. Additionally MySQL provides a number of geometric types and functions. Using these, webbased GIS capabilities are also being developed as part of this project. [URL 9]

phpMyAdmin is a tool written in PHP to establish a Web-based powerful interface for MySQL database administration and development. OS-independent phpMy Admin is an open-source application used under GNU GPL. [URL 15]

3.2 Generate Web Images on the fly

One of the Web services provided by GeoHistoryPortal v1.0 is an easy-to-use image gallery GUI that lets authorized users view all old and current visual materials of on the two Ottoman Fortresses. This photo gallery application allows users to display images on a website as thumbnails with a large image that displays over the top of the thumbnails after clicking one. Some features of photo gallery GUI include:

- ✓ Arrangement of images in categories and albums
- ✓ Automatic thumbnail
- ✓ Caption, title, description and user defined fields for each picture
- ✓ Image information stored in database
- ✓ Users can download images with Web interface
- ✓ Slideshow viewer
- ✓ Capture & Trim
- ✓ Print with associated data from database

PhotoGallery Web service runs in a LAMP (an acronym for Linux, Apache, MySQL, and PHP) environment along with a Macromedia Flash front end that loads data through the standard Macromedia Flash Remoting Components. This connects Macromedia Flash MX 2004 and application server, integrating rich Macromedia Flash content (Flash Client) with PHP (server) applications via another open source project called “Flash Remoting for PHP (AMFPHP).” According to the developers of AMFPHP, Flash Remoting for PHP enables objects in PHP to become objects in ActionScript.

Flash Remoting Components extend the Macromedia Flash MX authoring environment to develop rich Web applications, adding the ActionScript APIs needed to invoke remote services using Flash Remoting MX.

Because Macromedia didn’t provide support for PHP in Flash Remoting, an open source project was started to give PHP coders the benefits of Flash Remoting under their preferred server-side language.

AMFPHP is a class library for the PHP scripting language, which is used to allow PHP programmers to access Flash Remoting objects natively in PHP and connect Macromedia Flash to data.

AMFPHP is a free, open source replacement for the many commercial Flash Remoting packages provided by Macromedia for ColdFusion Server, .NET and Java. It is able to read the binary ActionScript Message Format (AMF) data sent from the Flash player to the server and can provide results back to the player in AMF format. AMF is a binary format that can describe any Flash variable of any data type. This format which is used by Flash to communicate with the server can be delivered over regular HTTP, is the core of Macromedia Flash Remoting. AMF is essentially a lightweight binary version of Simple Object Access Protocol (SOAP). [URL 1]

The AMF format was created by Macromedia as an attempt to obviate the need for XML objects when transferring data between a Flash movie and the server. XML objects proved to be quite inefficient; large amounts of data were required to describe simple objects, and the amount of bandwidth needed to transfer complex objects from some large Flash projects was simply not acceptable. [URL 1]

In this project, Macromedia Flash Remoting MX Components for Macromedia Flash MX 2004 Action Script 2.0 and AMFPHP 0.9.0 are used under GNU GPL. A dynamic graphics generation tool, GD Library 2.0.33, is also used under PHP on the application server. GD is an open source code library for the dynamic creation of images by programmers. GD is written in C, and “wrappers” are available for Perl, PHP and other languages. GD creates PNG, JPEG and GIF images, among other formats. GD is commonly used to generate charts, graphics, and thumbnails on the fly. Essentially, GD Library will convert the project’s uploaded images to a smaller physical size, so that each image uses less disk space and pages will load faster. [URL 2]

4. DISTRIBUTING DATA OVER THE WEB

As a base structure to provide cross communication between different systems in an n-tier environment, XML transports structured text between client and server. Since the XML encoding procedure is independent of hardware and software, it guarantees the easy transfer of files to different systems. Furthermore, the use of files in an open exchange format and the modular structure of GeoHistoryPortal, customized with PHP, have maintained the capacity for cross-platform interoperability.

XML can allow integration of data from disparate sources, with middleware applications pulling data from different databases and translating it into neutral XML for client-side processing. An XML document can contain descriptions of data that can be used in multiple applications, each application using the specific tags and delivering appropriate views for their particular tasks. [URL 16]

As a behind-the-scenes language in Web applications, XML, can readily combine together a wide variety of data types including text, graphics, audio, voice, spatial and more. This means that geometric data can readily be integrated with a wide range of semantic data types thus greatly enhancing the value and accessibility of spatial information.

GeoHistoryPortal v1.0 implements XML transfer functionality in the Web-based four-tier client/server computing of the project. It allows flexibly changing the structure of XML that the user expects.

5. CONCLUSION

Using structured methodologies and proven IT processes helps to increase flexibility, reliability and scalability and reduce complexity, risks and costs associated with client/server development.

Many Web developers rely heavily on a popular combination of open source software (such as LAMP) to develop and deploy their websites and applications.

GeoHistoryPortal and project’s client/server distribution architecture have been built to meet this project’s all business needs by relying primarily on open source products with their supportive communities at a low cost. Web applications, services and modules developed in this project scope run seamlessly on cross-platform products: Fedora, Apache, Tomcat, MySQL, PHP, phpMyAdmin, Java, AMFPHP, and XML.

In a nutshell, the on-going study GeoHistoryPortal utilizes interoperable technologies (specifications, guidelines, software, and tools) to lead this best-of-breed portal to a widespread, e(X)tensive and modular cultural heritage documentation platform and bring the power of cutting edge IT technology to the study of cultural heritages.

URLGRAPHY

URL 1: AMFPHP: Flash Remoting for PHP A Responsive Client-Server Architecture for the Web
http://www.amfphp.org/

URL 2: Boutell.com (Web Resources) - GD Graphics Library
http://www.boutell.com/gd/

URL 3: Extensible Markup Language (XML)
http://www.w3.org/XML/

URL 4: Fedora Project
http://fedora.redhat.com/

URL 5: Java.net: The Source for Java Technology Collaboration
http://community.java.net/

URL 6: LAMP Area
http://www.lamparea.org/index.22.html

URL 7: Macromedia Flash MX 2004
http://www.macromedia.com/software/flash/?promoid=BIHL
URL 8: Mozilla Foundation
http://www.mozilla.org/
URL 9: MySQL AB
http://www.mysql.com/
URL 10: PHP: Hypertext Preprocessor
http://www.php.net/
URL 11: Red Hat
http://www.redhat.com/
URL 12: Sun Developer Network: Java Technology
http://java.sun.com/
URL 13: The Apache Jakarta Project
http://jakarta.apache.org/tomcat/
URL 14: The Apache Software Foundation
http://www.apache.org/
URL 15: The phpMyAdmin Project
http://www.phpmyadmin.net/home_page/index.php
URL 16: W3C: World Wide Web Consortium
http://www.w3.org/

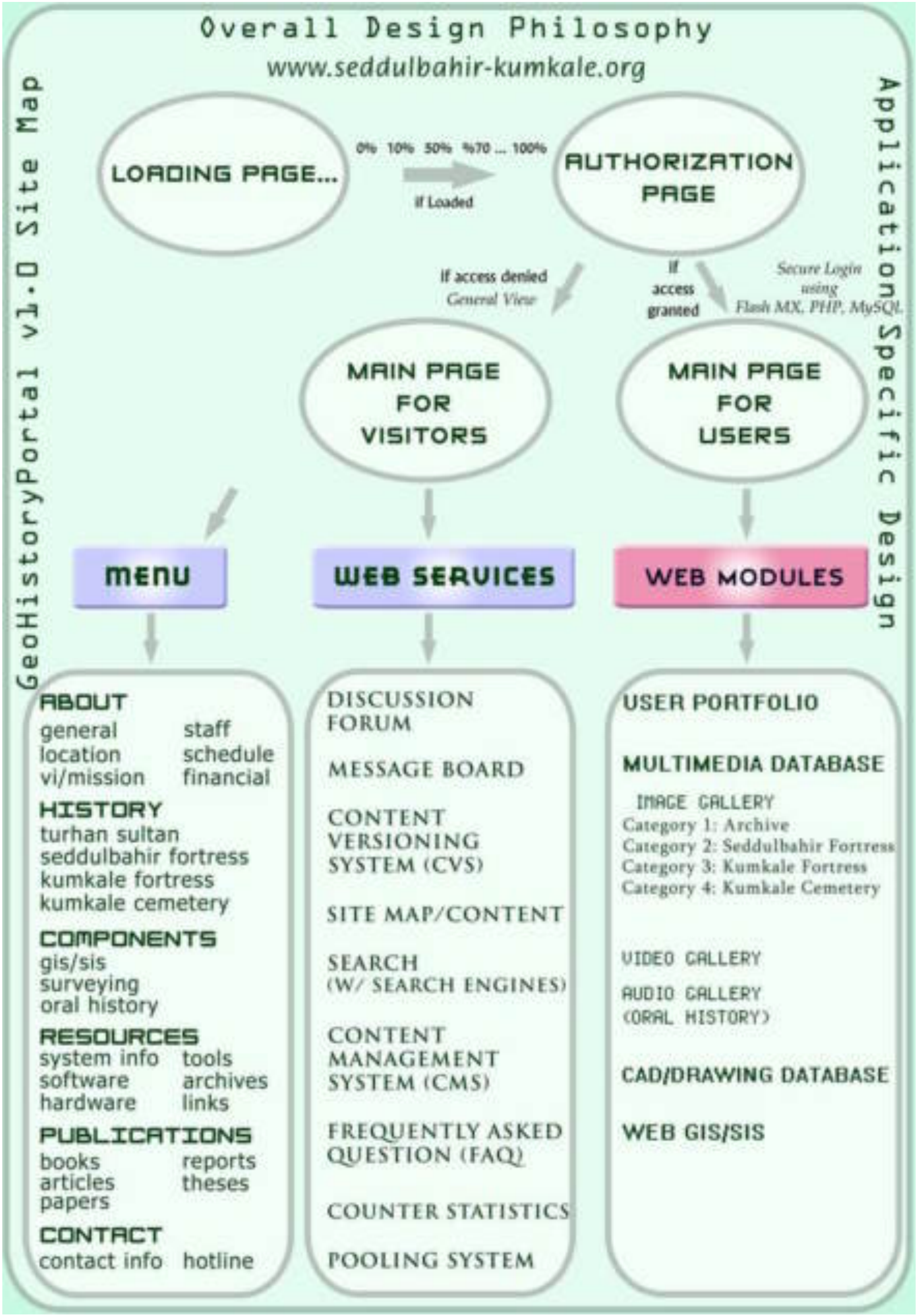
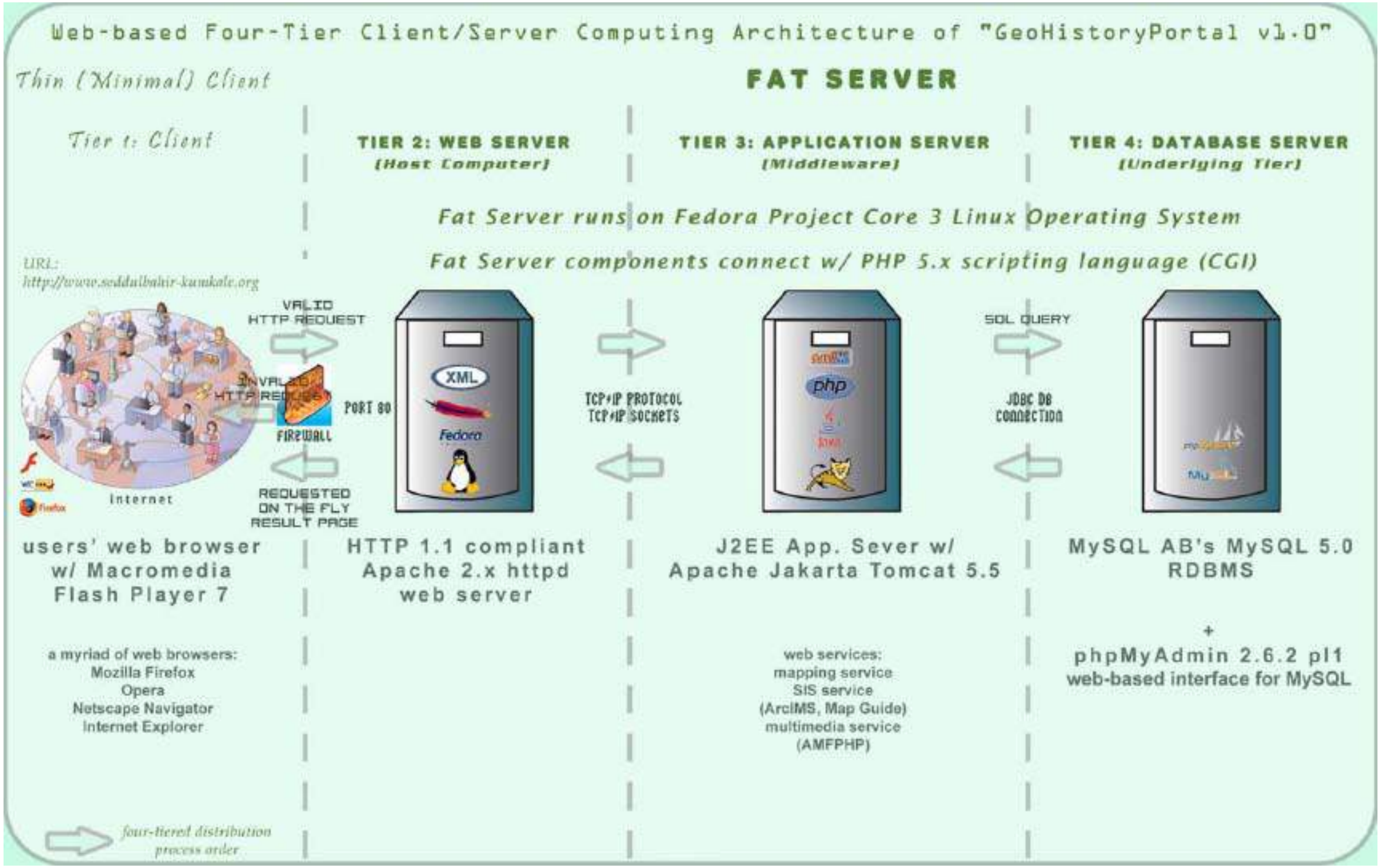


Figure 1. Overall design philosophy of *GeoHistoryPortal v1.0* Figure 2. Distributed tier client/server technology architecture of *GeoHistoryPortal v1.0*


Web-based Four-Tier Client/Server Computing Architecture of "GeoHistoryPortal v1.0"
Thin (Minimal) Client
FAT SERVER
Tier 1: Client
TIER 2: WEB SERVER
(Host Computer)
TIER 3: APPLICATION SERVER
(Middleware)
TIER 4: DATABASE SERVER
(Underlying Tier)
Fat Server runs on Fedora Project Core 3 Linux Operating System
Fat Server components connect w/ PHP 5.x scripting language (CGI)
URL:
http://www.seddulbahir-kumkale.org
VALID HTTP REQUEST
INVALID HTTP REQUEST
FIREWALL
PORT 80
REQUESTED ON THE FLY RESULT PAGE
Internet
XML
Fedora
TCP/IP PROTOCOL
TCP/IP SOCKETS
php
java
SQL QUERY
JDBC DB CONNECTION
MySQL
users' web browser w/ Macromedia Flash Player 7
HTTP 1.1 compliant Apache 2.x httpd web server
J2EE App. Sever w/ Apache Jakarta Tomcat 5.5
MySQL AB's MySQL 5.0 RDBMS
+
phpMyAdmin 2.6.2 pl1
web-based interface for MySQL
a myriad of web browsers:
Mozilla Firefox
Opera
Netscape Navigator
Internet Explorer
web services:
mapping service
SIS service
(ArcIMS, Map Guide)
multimedia service
(AMFPHP)
four-tiered distribution process order