



TECHNICAL REPORT
LHNCBC-TR-2003-001

Anatomic Images for the Public:

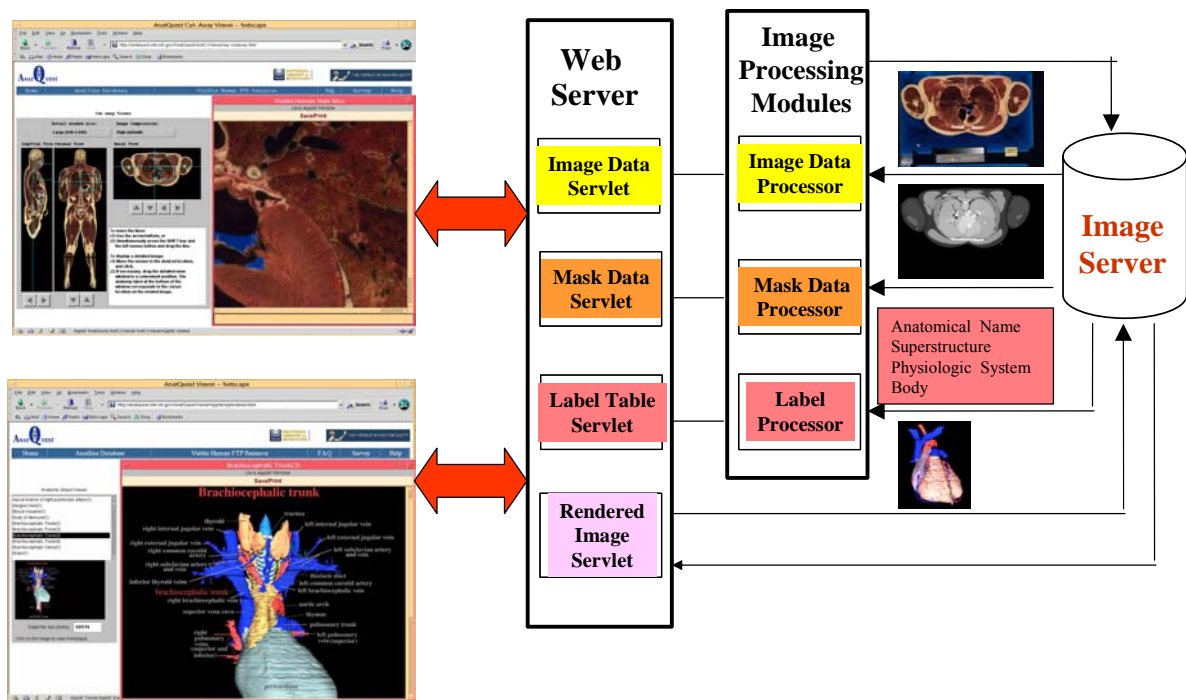
A report to the Board of Scientific Counselors
May 2003

Communications Engineering Branch

U.S. National Library of Medicine, LHNCBC
8600 Rockville Pike, Building 38A
Bethesda, MD 20894



Anatomic Images for the Public



A report to the Board of Scientific Counselors
May 2003

Communications Engineering Branch
Lister Hill National Center for Biomedical Communications

Anatomic Images for the Public: Contents

1 Introduction	3
2 Project Objectives	3
3 Project Significance	4
4 AnatQuest Project	4
4.1 Background	4
4.2 Prologue: AnatLine design	7
4.3 AnatQuest: Design considerations	11
4.3.1 AnatQuest for onsite visitors	16
4.4 Next steps	17
4.4.1 Increasing content	18
4.4.2 Linking text resources to image database	18
4.4.3 Implemented prototype: MEDLINEplus proxy server	27
5 Summary	28
References	29
Curriculum Vitae	

Anatomic Images for the Public

1 Introduction

In recent years the NLM has considered the provision of biomedical information to the lay public a high priority, as seen by one of the four goals of the NLM Long-Range Plan 2000-2005 to be to “Encourage use of high quality information by health professionals and *the public*.” In addition, among the high priority new initiatives identified by NLM’s Board of Regents in 2001 is Health Information for the Public.¹ In implementing this vision, NLM has created such services as MEDLINEplus, ClinicalTrials.gov and NIHSeniorHealth, all primarily for the public rather than for the biomedical clinical or research communities, our traditional constituencies.

It is in this same spirit that we have organized the AnatQuest project discussed here. While this project inherits and builds upon previous inhouse work that focused on the creation of a database for the Visible Human (VH) image set and a facility for the bulk distribution of the images over the Internet, its focus is the delivery of anatomic images to the lay public.

This report is organized as follows. In sections 2 and 3 we outline the objectives and significance of the AnatQuest project. In section 4 we briefly describe earlier inhouse projects as background to this one, the design tradeoffs that underlie the development of the AnatQuest system, and the design considerations in creating a kiosk version of AnatQuest for onsite exhibits. Also in section 4, we propose future work that should provide the lay public greater access to anatomic images.

2 Project Objectives

The overall goal of the AnatQuest project is to explore and implement new and visually compelling ways to bring anatomic images from the Visible Human dataset to the general public. Specific objectives are to:

- a. Develop systems to let users query the VH image database, via visual and textual navigation, and retrieve and display high resolution images with minimal bandwidth requirements.
- b. Investigate techniques to extend information services for the lay public (e.g., MEDLINEplus) with access to anatomic images.
- c. Explore options to segment and label the *high resolution* VH cross-section images beyond the thorax, the only region currently segmented and labeled, to enable the creation of images of 3D structures from all anatomic regions of the Visible Male and Female datasets.

3 Project Significance

The increasingly important mission of the NLM to “universalize” access to biomedical information, including biomedical images, require investigation into appropriate techniques and technologies, e.g., multi-tier system architectures, database design, design of suitable image viewers, image compression and others.

4 AnatQuest Project

4.1 Background

Previous inhouse work, viz., the 3DSystems and VHSystems projects, has been reviewed in past years by the Board. The VHSystems project had the goal of establishing a means for the bulk transfer of Visible Human data, mainly the color cross-sections, over the Internet. In addition to the original cross-sections (resolution 2048 x 1216 pixels) captured by a CCD camera, the 70mm film frames exposed during cryosectioning were digitized to a resolution of 4K x 6K, and cropped to 4K x 2.7K. These high resolution images as well as the original CCD-captured files were disseminated by an FTP server that continues to deliver an increasing amount of data, as seen in Figure 4.1. Demand for these images is widespread, as evident from the geographic and domain distribution of the recipients (Figure 4.2).

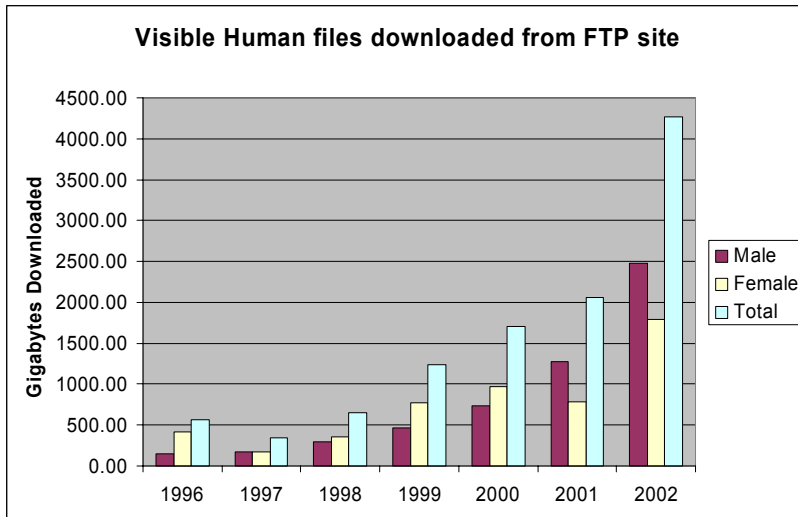


Figure 4.1.

Educational (.edu)	1,513,066	Australia	51,032	Norway	6,539
Unresolved	1,295,728	Belgium	41,143	Venezuela	5,292
Commercial (.com)	396,861	Italy	38,510	Greece	4,231
Japan	297,028	Singapore	36,364	Portugal	3,132
Germany	182,289	France	35,278	Chile	2,696
Network (.net)	171,468	Spain	34,740	United States (.us)	2,383
United Kingdom	140,303	Switzerland	29,602	Malaysia	1,037
Canada	128,874	Organization (.org)	25,863	Slovenia	500
Hong Kong	100,588	Austria	23,315	China	379
Taiwan	96,660	Israel	18,691	Mexico	227
Netherlands	84,293	Ireland	18,522	Hungary	212
Government (.gov)	79,964	Denmark	18,132	Egypt	129
Brazil	75,260	Military (.mil)	15,369	New Zealand	15
Czech Republic	58,356	Finland	13,924	Estonia	10
Korea (South)	56,276	Iceland	13,588	India	4
Poland	52,111	Sweden	12,599		

Figure 4.2. Domain distribution of Visible Human files

The main goals of the 3DSystems project were to create a database out of the VH data set, and to create suitable data and file structures so that access would be provided to the original images as well as to derivative products. A non-proprietary file format, VHI², was created to accommodate a wide range of image types: volume or surface rendered images, color cross-sections, MR, CT, segmentation masks, and volume of interest (VOI) image stacks. The target uses for these images were: rendered images for education, e.g., for curriculum development; the segment masks and VOI stacks for product development, e.g., to create surface and volume rendered images of organs; and the color cross-sections for research into the design of algorithms for segmentation, registration and rendering.

To achieve the goals of the 3Dsystems project, a prototype image management system, AnatLine,³ was developed to import and store the images, and to retrieve and export them. As a means for validating the design and implementation of the database and image management system, example 3D rendered images and VOI stacks were required. To create these, the thorax region of the male (consisting of 411 slices out of the total 1,878) was segmented and labeled, and a tool VHVis⁴ was developed to use these labeled segments to surface-render selected anatomic objects.

Also developed were tools needed to use AnatLine: VHParser and VHDisplay. The first is for unpacking the VHI data files into its individual components (cross-section images, byte masks, coordinate and label tables, etc.). VHDisplay is for displaying both cross-sectional and rendered images. Also, VHDisplay is augmented to audibly voice the names of anatomic structures as the images are displayed.

The principal users of AnatLine were expected to be scientists interested in testing algorithms for image processing (e.g., registration, segmentation, feature extraction) using the cross-sectional images and segmentation maps, for instance. Another application was the use of the segmentation maps for the construction of 3D anatomical objects, and thereby the testing of such rendering tools. For such expert users it was considered reasonable to have them download software for disassembling the retrieved files and displaying them. However, this approach posed significant barriers for the lay public. First, the requirement to download and install VHParse and VHDisplay was neither desired nor easily done by novice users. The second barrier was the lack of immediate visual feedback, a consequence of the large size of the VHI files and the difficulty of unbundling them quickly. As a consequence, AnatLine found relatively low use overall, and almost none at all by the general public.

Building on the AnatLine system, we created AnatQuest with the goal of providing widespread access to the VH images for lay users with special attention to those with low speed connections as well. AnatQuest offers users thumbnails of the cross-section, sagittal and coronal images of the Visible Male, from which detailed (full-resolution) views may be accessed. Low bandwidth connections are accommodated by a combination of user-adjustable viewing areas and image compression done on the fly as images are requested. Users may zoom and navigate through the images. Since its release in June 2002, it has averaged more than 80,000 hits per month, over 4 times the number of hits for AnatLine, as shown in Figure 4.3.

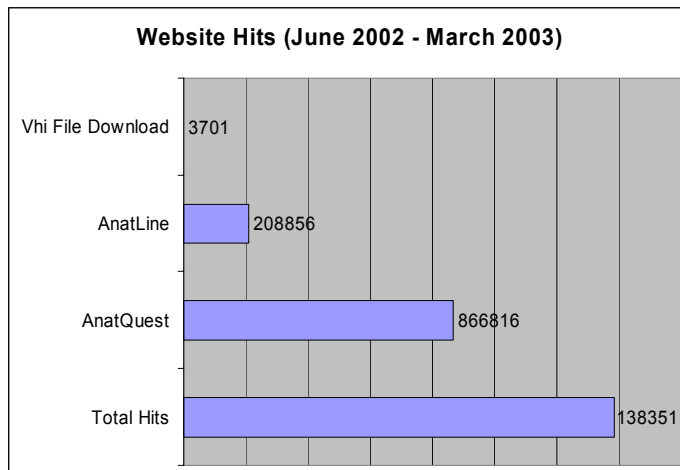


Figure 4.3.

In addition to its main purpose, AnatQuest serves as an entry point for both the FTP server for bulk downloading of VH files, and AnatLine that allows access through anatomic terms to high resolution cross-sectional images and segment masks (useful for rendering anatomic objects). Through AnatQuest the user may also retrieve more than 200 surface-rendered objects created at the Lister Hill Center as well as a few samples from outside sources (e.g., VoxelMan).

In a modification to the Web-mediated version of AnatQuest, a ‘kiosk’ version equipped with a touchscreen monitor was developed for the Dream Anatomy exhibit at NLM. This is designed as a Java application suitable for onsite patrons encouraged to navigate through the anatomy by touch.

The large size of the VH images, both in totality and as individual files (7.5 MB for the CCD files; 33.2 MB for the scanned 70mm photographs), motivated parallel inhouse research in the compression and transmission of these images.⁵⁻¹⁰ Studies were conducted in both lossy and lossless compression techniques. Among the lossy techniques investigated were JPEG and Digital Wavelet Transform (DWT) followed by scalar and vector quantization. For equivalent compression ratios (CR), DWT was found to yield better quality, artifact-free, decompressed images, but was computationally too expensive for real-time operation. Moreover, it would require client-side plug-ins, while JPEG is accommodated by most Web browsers.

Lossless techniques (Unix compress and its variations) yield low CR, on the order of 2 to 3). We combined background removal with a lossless method (arithmetic coding) and achieved a CR of about 9. Though a considerable improvement, this figure was deemed too low for practical use in transmitting VH images to the AnatQuest client. Our final choice was to remove the background (i.e., convert the blue background of all the slice images to uniform black), followed by JPEG compression. Other details on improving transmission rate are given in Section 4.3.

4.2 Prologue: AnatLine design

Since the AnatQuest system inherits the image and data repository developed for the AnatLine system as part of the 3Dsystems project, here we present the principal design considerations leading to AnatLine. We focus on: the choice of the object oriented framework for the design of the database, the file format to package images and metadata, and the visualization and rendering tools. Each of these is discussed next.

Database design. In this section we provide brief descriptions of the data and the conceptual data model. We also discuss the reasons for choosing an object oriented framework, including object oriented programming and DBMS, for the design and implementation of the database.

a. Data

In addition to the raw color cryosection images and the CT and MRI images from the data collection process, the data to be stored also includes annotated rendered images, as well as segmented and labeled images from the Visible Male’s thorax region. UMLS Metathesaurus concepts and relations were used to assign labels to the segments, and anatomical relationships among the segmented structures were identified. The x-y-z coordinates of the segmented structures were used to compute their spatial relationships. The anatomical and spatial relationships transform the database of structures to a form essential for navigating through the body. These entities and their relationships are described next in the data model.

b. Data model

The data model, as shown in the object relationship diagram in Figure 4.4, is organized around a number of objects representing: the body (i.e., the male or female cadaver), body regions (e.g., thorax, abdomen, etc.), the anatomical structures (e.g., organs and their parts), and the images. Note that the data model is general enough to accommodate future data from other cadavers or their parts.

The relationships among these objects may be either hierarchical or non-hierarchical. Hierarchical relationships allow for inheritance of shared properties. For example, the male and female body objects inherit generic descriptions of a heart and cardiovascular system but have different reproductive systems. Non-hierarchical relationships (that do not exhibit inheritance properties) represent the anatomical relationships among anatomical objects. These are modeled in terms of part-of and contains (or has-part) relationships as shown in the conceptual model in Figure 4.4.

Each of the objects is described in terms of its attributes. Simple attributes are brief, such as a name. More complex are the relational ones that contain instances of other objects which define a part-of/contains relationship between two objects.

As shown in Figure 4.4, the anatomical structure object consists of attributes for its name, superstructure, the region of which it is a part, and the physiologic systems to which it belongs. The "superstructure" attribute of an anatomical structure object refers to a second anatomical structure object which contains the first object. For example the *right-ventricle's* superstructure is *heart*, which means that the *right ventricle* structure is part-of the *heart* structure. The "body region" attribute points to a body-region object which contains the anatomical structure object. For example, *heart* belongs to the *thoracic* region. A physiologic system is a function of the body in which a structure is a component. For example, *heart* is a component of the *cardiovascular* system.

In addition, each of these database objects points to image metadata objects in the database representing data descriptive of an available image (whether MRI, CT, color slices, rendered, segment masks, etc.), including its size and the name of the image file in the Visible Human file server where the images actually reside.

The male and female body objects are represented in terms of their basic characteristics such as age, gender and race.

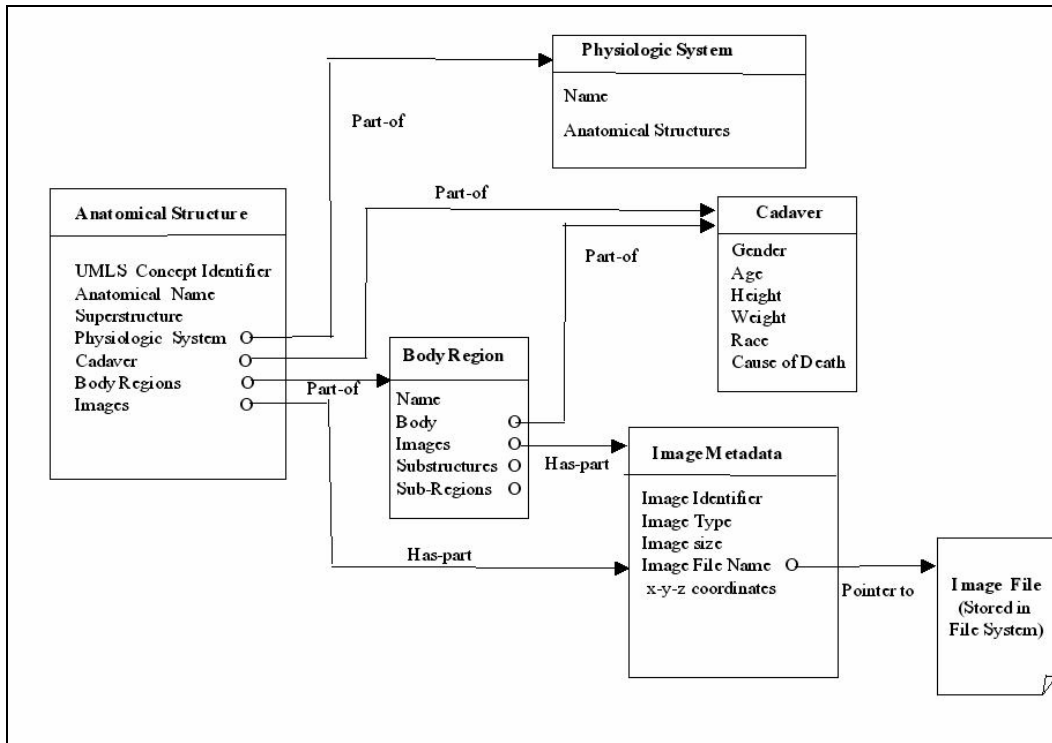


Figure 4.4. Database conceptual model

c. Object oriented framework

The choice of the object oriented framework for programming and data modeling was influenced by a number of factors including the structure of the data, and the requirement for efficient data representation and retrieval, as discussed below.

☞ Data structure

Anatomical data consists of both spatial and structural information, which in turn are represented by other complex objects. For example, a *heart* object consists of four other objects, each describing one of its chambers in terms of other substructures, e.g., *right atrium* containing the *right auricular appendage*. Such nested structures form graphs of objects that are represented more readily in object oriented environments.

☞ Efficient data representation

An object oriented programming framework provides efficient data representation through instantiation and inheritance. Male and female hearts, for example, are instances of the same structure which can be described once but *instantiated* twice. To give another example, a degenerative heart *is a* normal heart with extra conditions associated with it. This *is-a* relationship forms a class hierarchy which allows specialized objects to *inherit* characteristics of their exemplar objects, thereby providing a more efficient representation for reuse and maintenance. Since the "healthy" heart object is described by the common characteristics of heart, the "degenerative" heart only needs to add the extra conditions, because it inherits the generic information from its exemplar object (the healthy heart). These examples demonstrate the benefits of the instantiation and inheritance of objects in the object

oriented framework allowing efficient data representation, modeling, and maintenance.

✍ Efficient data and image retrieval

A goal of the 3Dsystem project was to enable the development of human atlases in which users may explore the body by navigating through structures, sub-structures, and sub-sub-structures by using the spatial and structural relationships among anatomical parts. This requirement is best supported with the graph navigation property of the object oriented framework.

d. Object oriented database

The advantages of the object oriented framework for the design of the system determined the selection of an object oriented database over a relational alternative. Object oriented databases offer the following advantages over their relational counterparts in the object oriented programming environment used in developing AnatLine:

✍ Transparent persistence: The objects within the programming environment are automatically saved in, and retrieved from, the database. This makes the database an extension of the computer memory and hence transparent to the programmer. For example when navigating through a *heart* object, its *right ventricle* is retrieved from the database transparently when needed, with no additional programming. For relational databases additional modules would be required to provide the necessary mapping between the database and the object in memory.

✍ Unified model: A single model for representation of data in the object oriented programming language as well as the object oriented database is preferable from the point of view of reduced effort in development and maintenance, and eliminating mapping between two disparate models.

✍ Ease of navigation: The navigation through objects corresponds to a graph or tree, thereby fitting the requirement for a navigable human image atlas.

In light of the factors listed above, we selected ObjectStore® as the DBMS. It was found to provide efficient data management, good performance, concurrency and multithreading. Unlike relational or object-relational DBMS which retrieve related rows of data by executing joins at runtime, ObjectStore stores and manages data components and objects with their relationships intact. Also, as in any DBMS, ObjectStore offers concurrency (allowing multiple users and applications to simultaneously access and update the database) and a multithreading feature (allowing the use of kernel threads, asynchronous I/O and shared memory.)

VHI File Format.² This non-proprietary file format was created inhouse for several reasons. Metadata, for instance, are necessary not only for database searches for desired images, but also for viewing and subsequently processing them. Also, images could consist of many image components that have to be kept together. For example, a

segmented heart image consists of a stack of image slices that make up the heart volume, a stack of bitmask overlays describing the components of the heart, the labels, and the coordinates. The VHI file format was created to retain all these elements in a single file to avoid processing problems associated with missing elements if they were transferred in pieces.

Other requirements for a file format were that it should allow the storage of different visual types of interest in the future, i.e., any standard image file type, QuickTime®, volume 3D data, streaming video, etc. A study of existing file format standards, including DICOM, showed the need to develop a format inhouse.

Tools. In the 3DSystems project, a number of tools were developed, both to enable users to access, retrieve and use the data, as well as to validate the design and implementation of the file format and the database. Among these are VHParser, VHDisplay, and VHVis. All of these tools were created as standalone applications using the Java® programming language. VHParser and VHDisplay may be downloaded via the AnatLine website. (<http://AnatQuest.nlm.nih.gov/Anatline/>).

VHParser unbundles the VHI file and saves its components to a directory on the user's local disk. It was also designed to be compatible with third-party visualization software. For example, it has the capability to map the byte masks used in describing the anatomical structures to a format that can be used by the Mayo Clinic's Analyze® imaging software.

VHDisplay was created as a proof-of-concept application to demonstrate the flexibility of the VHI format. VHDisplay has the capability of categorizing these VHI files, and presenting this information to the user as a selectable list. VHDisplay uses the Java Advanced Imaging® APIs (described further in section 4.3) to allow the manipulation of the displayed image data in useful ways. For example, VHDisplay converts the retrieved byte mask overlay by a Sobel edge filter operation to create an outline of the mask that defines the shape of a segmented structure in the image. This helps the user visualize such structures more precisely.

To further validate the database and the VHI format, VHVis, a rendering tool, was created as a Java standalone application implemented with interfaces to the VTK® visualization toolkit.^{11,12} With the database entries serving as a selection guide for structures, the user (renderer) can employ VHVis functions to create surface rendered models of these structures in 3D visualization space. VHVis enabled the selection of multiple anatomical structures, color them for best contrast, and render them in correct spatial 3D alignment.

4.3 AnatQuest: Design considerations

As already stated, AnatLine was designed with researchers in mind rather than the lay public. Its utilization by lay users was low due to a number of factors. A major reason was the excessive transmission time due to the size of the VHI files. Some files are on the

order of 1 GB when segmented slice images and bitmap overlays for some of the larger organs are packaged. Moreover, to preserve the complete and accurate content of these files, no lossy compression (which would have yielded significant compression) was introduced. To transfer such large files to a user site in a reasonable time requires connections that are not only high speed links, but need to be reliable to avoid resending because of intermittent connections. Such robust high speed links are often not available to the lay user.

Further, the AnatLine Web client that serves as the interface for searching the database was not designed to display slice images, again due to the large size of these images and the significant delays experienced in transmission. The lack of such an online display capability limited the usage of the system to an audience of imaging researchers familiar with the image content and able to render and display images offline with their own tools or the ones provided.

We undertook the design of AnatQuest to offset these drawbacks, and thereby accommodate a wider range of users. The main effort in this development focused on suitable image viewers and server-side image processing modules, each dictating different sets of development tools.

Two image viewers are provided in the AnatQuest GUI: A Rendered Image Viewer (RIV) to display rendered images in 2D projection, and a Cut-away Viewer (CAV) to display thumbnail as well as detailed-view images of two-dimensional slices of the front (coronal), side (sagittal), and top (axial) views of the body.

The Cut-away Viewer allows the user to dynamically navigate the Visible Human body via a Web browser in three dimensions: along the x, y, and z axes. A view port is provided for each of the three views to allow navigation through the corresponding high-resolution images. The user selects the region of interest encompassed by a green field-of-view rectangle for which a detailed view can be downloaded and displayed. Two parameters determine the amount of data transferred to the client interface and thereby accommodate users with varying bandwidth connections: the size of the field-of-view which is user-adjustable, and the degree of image compression, also selectable by the user. Both image viewers are viewed through a standard Web browser and do not require the installation of any plug-ins.

The image viewers were designed to serve as the clients in a three-tier client-server architecture (Figure 4.5), in which a tier is a logical partitioning of an application across client and server. The image viewers have only the graphical user interface, while the middle tier contains the application logic, and the third tier consists of the image server.

between applets at the first tier and servlets at the middle tier provides the fast response necessary for a wide user community equipped with low speed connections. (Running the Web server and the servlet engine in this middle tier are Apache and Java Jakarta-Tomcat, respectively).

As shown in Figure 4.6, the user sees the Cut-away Viewer as three side-by-side view ports that occupy the left part of the Web page. Each view port contains a thumbnail representation of a sagittal, coronal, or axial slice. In each view port, a user-movable hairline controls the location of the x-, y-, and z-axes (planes). To dynamically display a different cut-away view of the body in three directions, the user moves a hairline vertically or horizontally across any one of the view ports, resulting in updating the corresponding spatially related one. In this way, the first view port (control source) and the second view port (control target) form a coordinated pair.

When the user clicks on the rectangular field-of-view box in any of the view ports, a detailed view of the section of the image encompassed by the box is displayed. The user may select the size of this detailed view (and therefore the number of bytes received from the server) from a pull-down menu in the GUI. The three field-of-view sizes are: 448 x 320 pixels (for low bandwidth connections); 640 x 448 pixels and 640 x 640 pixels, for higher bandwidth connections. At the user's click, the coordinates and size of the selected field-of-view are sent to ImageDataServlet on the server.

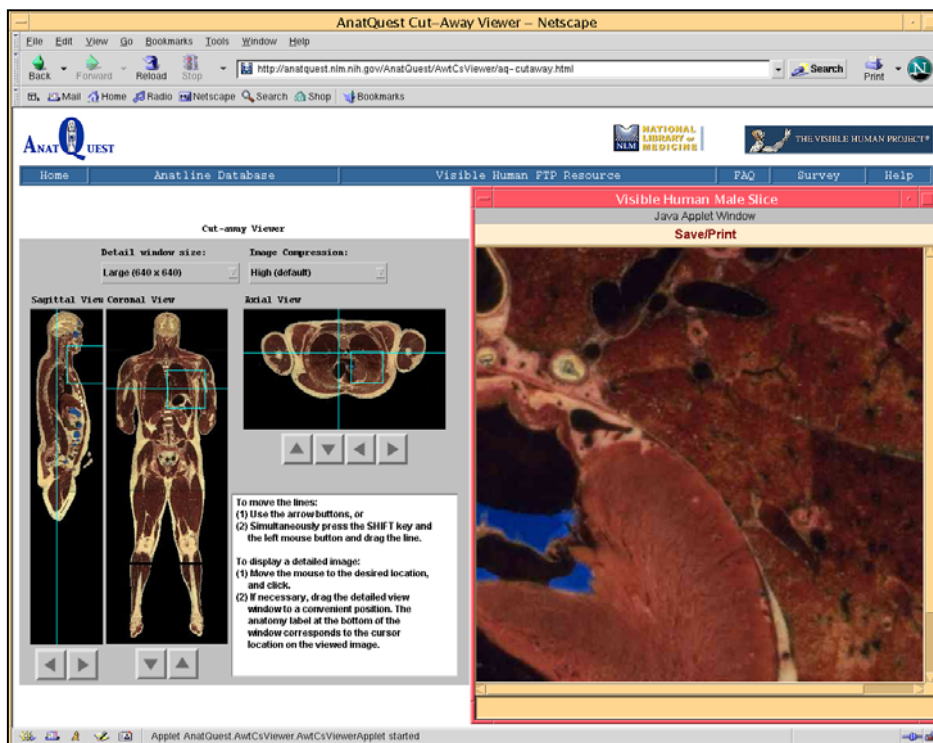


Figure 4.6. AnatQuest Cut-away Viewer

This servlet validates the parameters, passes the information to another server-side program called ImageDataProcessor which extracts the cryosection image file that contains the requested detailed image frame according to the specified position and dimensions. It then uses the JPEG compression scheme in Java Advanced Imaging to compress the image data at a user-specified compression ratio. The generated image data is then returned to the ImageDataServlet, which sends the data to the applet at the client to be displayed. Similarly, mask information is retrieved through MaskDataServlet and MaskDataProcessor, and label table data (a table of anatomical terms) through the LabelTableServlet and LabelTableProcessor. The label table and mask information enable the detailed view window to display the label (name) of an anatomic structure under the mouse cursor.

In order to have the region of interest efficiently encompass the detailed image, i.e., at their boundaries, we picked view port sizes that are divisible by 64. Also, since the files are stored as tiled TIFF images, and the average user's screen resolution is expected to be 1024 x 768, we set the maximum window size to 640 x 640 pixels to enable users to display an image portion that is large enough to include a significant part of the image, without taking up the entire screen.

To increase the transmission speed of the detailed image portions from the AnatQuest server to the user's browser, the images are compressed on the fly using the JPEG compression scheme, as mentioned earlier. The user may select one of five compression levels from a pull-down menu in the GUI: high (the default value), medium/high, medium, low/medium, or low. The approximate compression ratios (sample averages) range from 60 at the high level to 11 at the low. To achieve a reasonable response time for users with low speed modems, we set the default window size to 448 x 320 pixels and the default image compression option to high. For example, assuming an average transfer speed of 33.2 kbps commonly available through such modems, these default values allow a JPEG image of approximately 7,168 bytes to be retrieved and displayed in the user's browser in 1.7 seconds.

The GUI of the Rendered Image Viewer applet consists of a user-selectable list of available rendered images. Selecting an item in this list results in the display of its thumbnail image together with an indication of the full image size. When the user clicks on the thumbnail, the full-size rendered image is displayed in a separate window. Figure 4.7 shows the components of the RIV.

As noted earlier and summarized in the following, several aspects of the AnatQuest design exploited the functionality offered by Java Advanced Imaging. First, we exploited JAI's file handling capabilities to locally store very large image files in the Tiled TIFF format that would be too resource intensive for most users to download. JAI was also used to extract only those tiles required by the user (as defined by the coordinates of the selected region of interest), and to compress the image data to a JPEG file, as mentioned earlier.

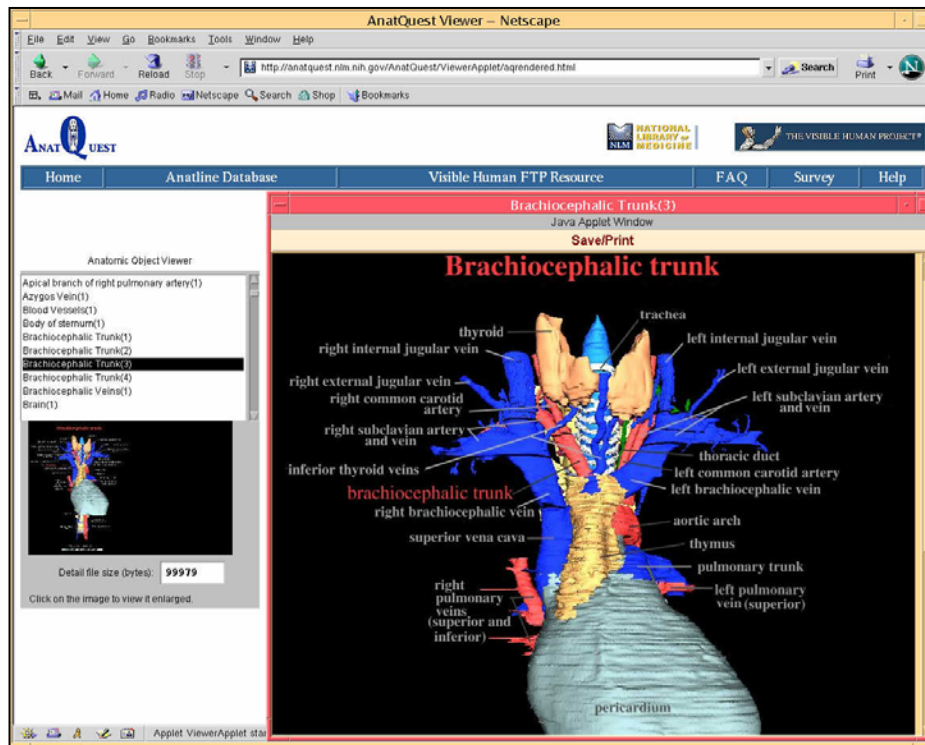


Figure 4.7. AnatQuest Rendered Image Viewer

As a class library, JAI supports generalized image processing functionality built as an extension to the Java programming language. In a simple programming model, JAI provides a rich set of imaging capabilities that can be readily used in applications without undue programming overhead. JAI encapsulates image data formats and remote method invocations within reusable image data objects, so that an image file would be processed the same way, whether in local storage or across networks. JAI provides a rich set of cross-platform imaging APIs, allows distributed imaging, and comes as an object-oriented API that is flexible and extensible. Also, it is device independent and provides high quality performance on various platforms.

4.3.1 AnatQuest for onsite visitors

While reliable and rapid online access promotes the use of anatomic images by the public, onsite displays in exhibits are further opportunities to reach another public constituency: visitors to the library. One such exhibit is Dream Anatomy¹³ currently installed at the library, and expected to remain for at least a year. To serve as an onsite display in this exhibit, AnatQuest was modified to take advantage of the particular characteristics of this environment. Different issues came into play in the design of this modified system we call AnatQuestKiosk.

First, as an *onsite* system, AnatQuestKiosk did not need to rely on a Web browser, and could be designed as a standalone application. Second, an exhibit visitor in close

proximity to the screen is inclined to navigate by touch; hence, a touchscreen monitor is provided. The implications of these factors are discussed below.

Eliminating the need for a Web browser allowed both the Cut-away Viewer and the Rendered Image Viewer in AnatQuestKiosk to be designed as Java applications rather than as applets. Compared to applets, Java applications allow the user interface to be built with a greater variety of Java visual control class libraries (as in Java Swing), giving the interface a more polished look than possible with the Abstract Windowing Toolkit (AWT) libraries used in AnatQuest. Using AWT in the online AnatQuest system is necessary since these libraries are built into Web browsers, while Swing libraries are not. Swing libraries have to be downloaded into the Web browser as plug-ins, adding to the download time for an online system. Since this is not a problem with a standalone onsite system, the use of Swing in AnatQuestKiosk is an advantage.

While touchscreen monitors are attractive for onsite applications, in designing applications to run on them one must take into account the size of the GUI controls. For example, buttons and sliders must be large enough to respond to the touch of a finger. To support this, we replaced the thin vertical and horizontal user-movable hairlines that control the thumbnail images in AnatQuest with large sliders. This required the modification of the Java library slider controls to achieve a tailor-made look-and-feel.

To ensure that AnatQuestKiosk runs continuously, and covers the entire screen, it is necessary to prevent users from inadvertently stopping or starting the application, or from coming in contact with the operating system. We met these requirements by using Java's Fullscreen Exclusive Mode API, a new feature in JDK 1.4. This API supports high-performance graphics by suspending the operating system's windowing function so that the application takes full control of the contents of video memory and draws directly to the screen.

Finally, as an exhibit display, it is desirable for the AnatQuestKiosk application to be unaffected by network outages. To guarantee uninterrupted use, the image files are locally stored in the system to avoid continuous fetching from a remote site. However, the application is designed to fetch images remotely as well by enabling the parameters passed to its startup script to specify the image site as a URL for the image database server. This feature is useful in the event that new images, particularly newly rendered 3D structures, are added to the database.

4.4 Next steps

There are a number of interesting directions that the AnatQuest project could take to further serve the lay public. Among these are: (1) Increase the contents of the image database; (2) Extend information systems designed for the public, e.g., MEDLINEplus, to provide anatomic images. The full realization of the second direction depends on adequately addressing the first, as discussed below.

4.4.1 Increasing content

In the 3DSystems project, a limited number of 3D surface-rendered structures were produced, mainly as exemplar images to evaluate the design of the database and the processing tools. These images were created from the segmented and labeled *high resolution* cross-sections of the Visible Male thorax region, with the assumption that this dataset would allow rendering of more detailed structures than possible with the lower resolution CCD-captured data. Currently, AnatQuest provides access to these images, numbering about 200, as well as some produced by other organizations.

However, the fact remains that other regions of the high resolution Visible Male, and all of the Visible Female, have not been segmented or labeled. If one is to make structures in all regions of the human anatomy available, this would be a necessary task.

User feedback suggests a demand for rendered images of structures in the head and abdomen regions. That these regions are of highest interest is evident not only from such anecdotal evidence, but also from the cumulative demand statistics shown in Figure 4.8.

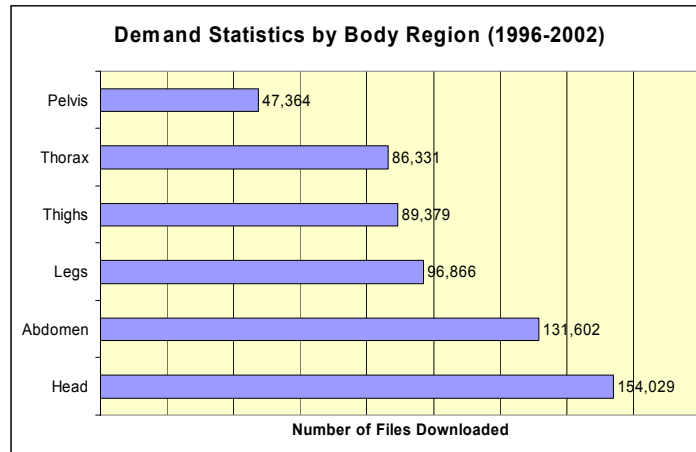


Figure 4.8.

Meeting this demand would require these regions to be segmented and labeled. The approach taken in segmenting and labeling the high resolution slice images of the thorax region (obtained by scanning the 70mm photographs taken during cryosectioning), though computer-assisted had a manual component and was therefore labor intensive. Since there appears to be work done in segmenting the low resolution (CCD-captured) images (e.g., Gold Standard), it is possible to conceive of more efficient techniques based on pixel statistics and interpolation to use the low resolution data to segment the high resolution images. The availability of a complete set of segmented and labeled high resolution slice images for both the Visible Male and Female would offer developers material for creating and testing tools for surface and volume rendering, and would consequently supply the lay user with rendered images of objects in any region of these image sets.

We additionally note that successful rendering requires the correct registration (alignment) of adjacent slice images. The high resolution images are found to be misaligned to some degree, whereas the low resolution slices have been shown to be correctly aligned. It is possible to conceive of techniques to use the low resolution set as reference images to correct the registration of the high resolution slices.

4.4.2 Linking text resources to image database

One of the long term goals of the Visible Human Project is to transparently link the print library of functional-physiological knowledge with the image library of structural-anatomic knowledge into a single, unified resource for health information. Indeed this has been echoed several times in the past, including the 1989 Board of Regents Planning Panel whose recommendations included: "The NLM should encourage... research into methods for representing and linking spatial and textual information..."

In this section we present our early research in this area. We explore the steps required to accomplish linking text from a biomedical document to the relevant images, and apply the concepts to linking a search of MEDLINEplus, NLM's information system for the lay public, to anatomic images in our database. The functions to implement linkage are:

- 1 Document Analyzer: Identifying biomedical terms in a document.
- 2 Term Mapper: Identifying the relevant anatomical terms.
- 3 Image Locator: Identifying the images in the image database.
- 4 Link Assembler: Linking the identified terms to the images.

Figure 4.9 shows the top level architecture of the system and its components. Actual implementations might utilize the components differently. For example, the document analyzer and the term mapper can be provided as independent services for indexing documents outside of AnatQuest. For simplicity, all the above text and image services can be made available through a single API interface.

In this section we discuss a number of alternative solutions to each of these steps, and place them in the context of a prototype system that links MEDLINEplus to the anatomic images in our database (described in section 4.4.3). First, we briefly describe the AnatQuest API.

The AnatQuest API is a URL-based parametric API in which a desired image can be described in terms of its metadata in the URL statement. In addition, the user can specify: (a) whether mapping the terms through a vocabulary (e.g., UMLS) should be utilized, and (b) the form of the output desired (e.g., XML file, image file). The following is an example of the API for the metadata in XML for a heart image:

<http://image1.nlm.nih.gov/pm/servlet/ImageLogic?name=heart&resultFormat=xml>

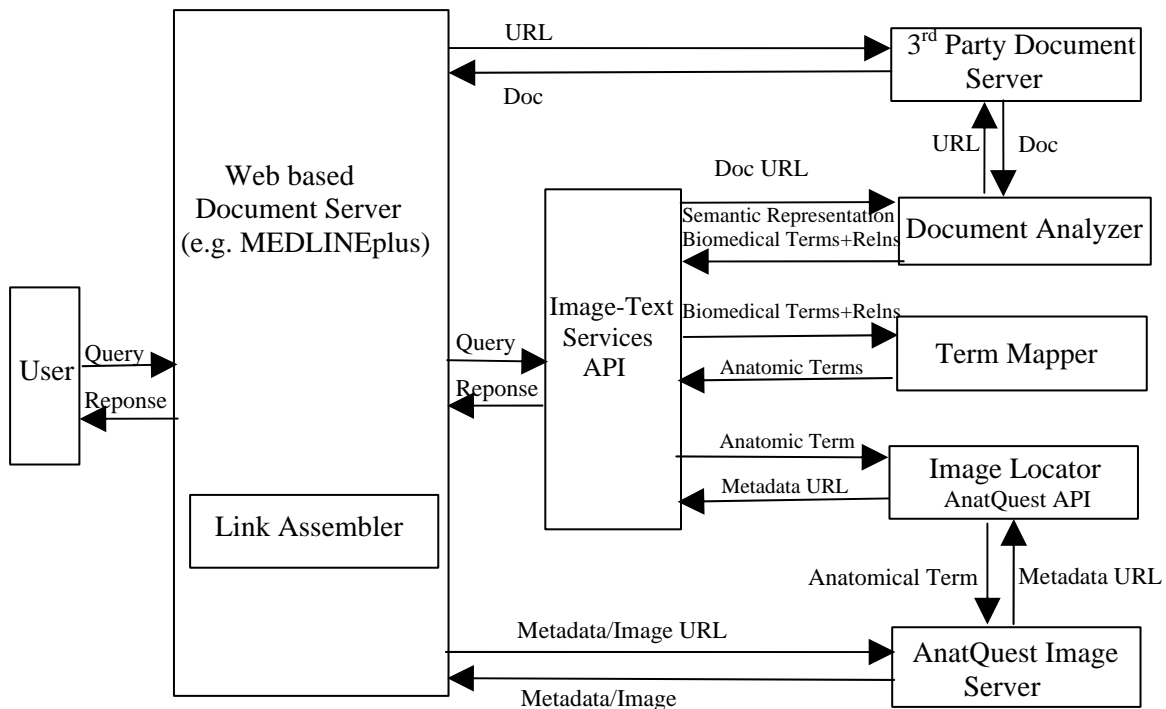


Figure 4.9. Architectural components for linking text and images

Document Analyzer: Identifying biomedical terms in a document

The document analyzer creates a representation of document content. This may be as simple as a list of keywords, or more complex as a semantic network representation of the contents.

Alternative approaches are:

- ✍ Word frequency vector
- ✍ Pre-assigned vocabulary terms
- ✍ Vocabulary-based term identification
- ✍ Vocabulary-based term and relation identification
- ✍ Text understanding

The document analyzer can be as simple as a word-based analyzer or as complex as a full fledged text understanding system. Such a system would be able to infer from the context, for example, whether the term "brain" should point to a male or female brain.

The vocabulary-based term identification approach would process the text to identify the occurrences of the vocabulary terms in it. This could be an elaborate process taking into consideration such factors as lexical variances, word order variances and synonymy, as implemented in the MetaMap program.¹⁴ The accuracy of this approach would depend on the level of sophistication of the analyzer. Another alternative would be to use domain

specific semantic processing to identify the relationships as well as the anatomical terms in the document, thereby generating a richer representation of content.^{14a}

The parsing of documents can be initiated either by the Web-based document servers (e.g., MEDLINEplus) or provided as a service by AnatQuest. No parsing is needed when the documents already possess vocabulary based indexing terms (e.g., MeSH terms assigned by an indexer) as part of their metadata.

Term Mapper: Identifying the relevant anatomical terms

Since our objective is to link text resources to anatomic images, the question is whether the biomedical terms identified in a document are anatomical. In fact, the chances are that they are not explicitly anatomical terms. However, using the concept relationships in the UMLS Metathesaurus we can map a biomedical term in the document to a related anatomical term. For example, the term *pneumonia* (a disease) could be mapped to *lung*, the underlying organ with the disease. Other mappings are also possible, e.g., mapping a particular anatomical term, for which there is no image in the AnatQuest image database, to a more general anatomical term for which there is an image in the database. But to demonstrate the concept we only focus on the location-of relationship.

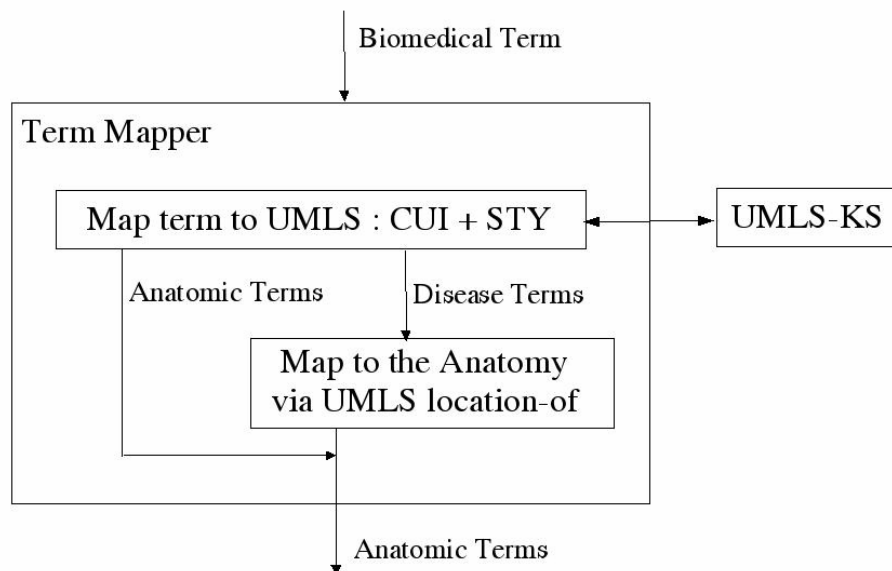


Figure 4.10. Term mapper components

Figure 4.10 shows the mapping processes implemented in our prototype system linking a search of MEDLINEplus to images. The steps are:

- 1 The term mapper maps biomedical terms to UMLS concepts through UMLS-KS, identifying their concept unique identifiers (CUI) and their semantic types (STY). The mapping takes into consideration lexical variants and synonymy information.

- 2 For a term identified as anatomic based on its semantic type, the mapper returns the same term.
- 3 For terms that are not anatomical, the server uses the UMLS Metathesaurus concept relations to obtain the related anatomical term. This is done by using the location-of relationship that maps disease terms to corresponding anatomical terms. This appears to be the most suitable mapping for MEDLINEplus since most health topic pages to which it points possess MeSH designators that are disease terms.
- 4 For a related term identified as anatomic based on its semantic type, the mapper returns the anatomical term.

The application builders can request the term mapping via the AnatQuest API. If they choose otherwise, the mapper is not applied and the user's term is queried against the image database.

Image Locator: Identifying the images in the image database

There are three possible approaches to identifying and accessing the images, each influenced by the design of the image database:

- ✍ Database-brokered access via a URL-based parametric API
- ✍ XML file-based access with searchable metadata
- ✍ Integrated image and metadata file format

The first approach is based on a private database system which brokers the access to the images within a controlled environment, and custom-made interfaces designed for particular systems. The AnatQuest Web browser falls in this category. Here users can access the images through the GUI menus, the database is local to the system, and the database query language is not accessible to the users. In order to relax this constraint, as mentioned earlier, we have defined a URL-based parametric API in which a desired image can be described in terms of its metadata in the URL expression.

The second approach is to eliminate the database and store the metadata for each image in separate files that are made public for search engines to crawl and index. These metadata files, in addition to the information about the image, contain the URL to the actual image file. The metadata encoding can be as simple as unstructured text included in the ALT text of the IMG tag of an HTML document. Alternatively, the metadata can be represented in a more formal XML structure with data elements that can be searched individually. Search engines can index specific XML data elements, in order to provide results with high precision and recall. Alternatively, the search engines may ignore the XML structure of the data and index all the metadata page contents. We have defined a detailed XML schema describing the structure of the image files which can be crawled and indexed by search engines. Figure 4.11 shows a fragment of the XML structure for the image metadata file.

In this example the web crawlers can be instructed by their owners to index the contents of the VHI-image.term.name nested field when it comes across an XML file with the root element <VHI-image>. Here the value of the name field is *Heart* allowing the crawler to add the URL to this example XML file to its list of index entries for the term *Heart*. Thereafter, when the search engine is queried for *Heart* the URL for this XML file will be returned. The GUI can then display desired elements of the XML file such as the name and the thumbnail for the image.

```

<?xml version="1.0"?>
<VHI-image xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:VHI="http://anatquest.nlm.nih.gov/vhi">
  <term anatomicalType="Structure">
    <cul>C0018787</cul>
    <name>Heart</name>
  </term>
  <specimen>
    <id>1</id>
    <name>Visible Human Male</name>
    <owner>National Library of Medicine (NLM)</owner>
    <sex>Male</sex>
    <race>Caucasian</race>
    <age>38</age>
  </specimen>
  <image size="59008" format="jpg" modality="70mm" capturedBy="NLM">
    <rendered segmentedBy="EAI" renderedBy="NLM">
      <url>...</url>
      <title>Anterior View of Heart</title>
      <modifier>1</modifier>
      <thumbnail size="5073" format="jpg" dimensions="">
        <url>...</url>
      </thumbnail>
    </rendered>
  </image>
</VHI-image>

```

Figure 4.11. XML structure of image data

The third alternative solution for the image locator is to combine an image with its metadata in a single file. This has the advantage of preventing dangling links between the image and its metadata when they are stored in separate files and moved individually. This approach requires a standard file format that encompasses both pieces, as well as a standard set of metadata. A number of groups are attempting this. For example, the PNG2000¹⁵ and extensions to the Dublin Core-based metadata activities¹⁶ seem to be moving towards this objective. This approach also requires modifications to search engine crawlers to open a compressed image for retrieving and indexing its metadata.

A PNG-enabled search engine should be able to decompress a PNG file format, as well as know about the metadata in order to extract and index the data elements. An example of such an implementation is the PNG-enabled version of HotMeta.¹⁷

Link Assembler: Linking the identified terms to the images

Ways to present the links between the identified terms and their corresponding images depend on the design of the GUI of the Web based document server. Possible approaches to this problem are:

- ✍ Hot linking anatomical terms in a document
- ✍ Related-Images button or menu option, based on document similarity criteria
- ✍ Portal approach

In the hot-linking approach, every identified term from the parsing phase is converted to a hot link that indirectly points to a relevant image in the Visible Human Image database through the AnatQuest API. The degree to which the links point to a specific image depends on the ability of a text analysis system (parser) to resolve ambiguities. For example, by default the links would point to images of the Visible Male unless the parser can understand from the document's context that the topic is about the female. Needless to say, this is a difficult task with a high probability of error. Figure 4.12 provides the flow diagram for the hot-linking approach, with numbers representing the sequence of actions following a user query.

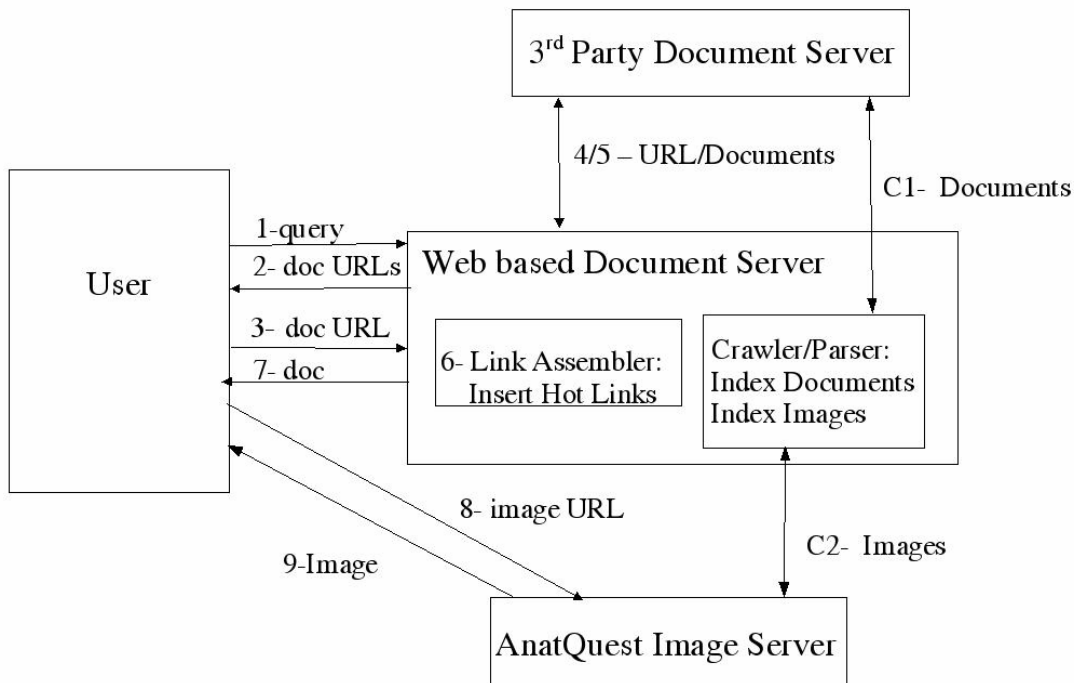


Figure 4.12. Link Assembler – hot linking anatomical terms

As shown in the diagram, steps C1 and C2 require the Web based document server to compute and save the offsets of the anatomical terms within the documents at the indexing phase so that hotlinks may be inserted in the third-party documents at time of retrieval. The Web based document server would not store these documents but would act as a proxy server for them in order to insert the hotlinks.

An advantage of this approach is that the anatomical terms are turned into hotlinks which the user can easily click on while reading the document. A drawback is the possibility of mismatch between the text and the image due to parsing inaccuracies. Further, the server needs to maintain the offsets to all the anatomical terms within the document, requiring additional storage and processing to include the hyperlinks for each retrieved document.

In the Related-Images approach to displaying the links the original text of the document is displayed for viewing by the user. The GUI provides a "related anatomical images" button in proximity to the document. This button, when clicked, will send the set of vocabulary terms, identified earlier in the parsing phase and already associated and stored with the document metadata, as query terms to the image database. Thumbnails of the returning (matched) images will then be displayed on the sidebar next to the main document for the user to view. Selection of each thumbnail will open up a detailed view of the image.

A variation of this approach is to display the thumbnails of related images on a side bar when the document is first displayed instead of providing the related-images button.

The advantage of this approach is that it reduces possible confusion caused by a mismatch between terms and images, since the links or thumbnails are offered as "related images" rather than "exact images" suggested by hotlinked terms in the text (as in the previous approach). A drawback is that the Web based document server has to act as a proxy server with control over user interaction with the 3rd party document including the related-images button.

In the portal approach to displaying the links we do not need to analyze text at all, and nor are changes made to the documents. We simply make the Visible Human image database website open to the web crawlers of the Web based document servers (e.g., MEDLINEplus). These web crawlers can then build an index to the images in the database, and simply treat the images as individual documents. Through metadata, the image database would provide the anatomical terms and possibly their synonyms to be indexed by the crawlers. Advantages of this approach are that: (a) no changes need be made to existing Web based document servers; (b) since the documents do not need to be parsed, parser accuracy would not be an issue. The flow diagram for the portal approach is given in Figure 4.13. The interaction consists of:

- 1 The user types in a term at the query prompt of the document server.
- 2 The browser displays a list of documents grouped under a number of categories. For each document, a one-line summary is displayed together with the hotlinked URL to

access the full document. One of the categories is "Images", under which appears a list of anatomical terms the crawler has found to be relevant to the user query.

- 3 A document URL points directly to a 3rd party document server.
- 4 A returned document from this server is displayed for the user.
- 5 An image URL points to the AnatQuest Image server.
- 6 An image or its metadata file is returned from this server.

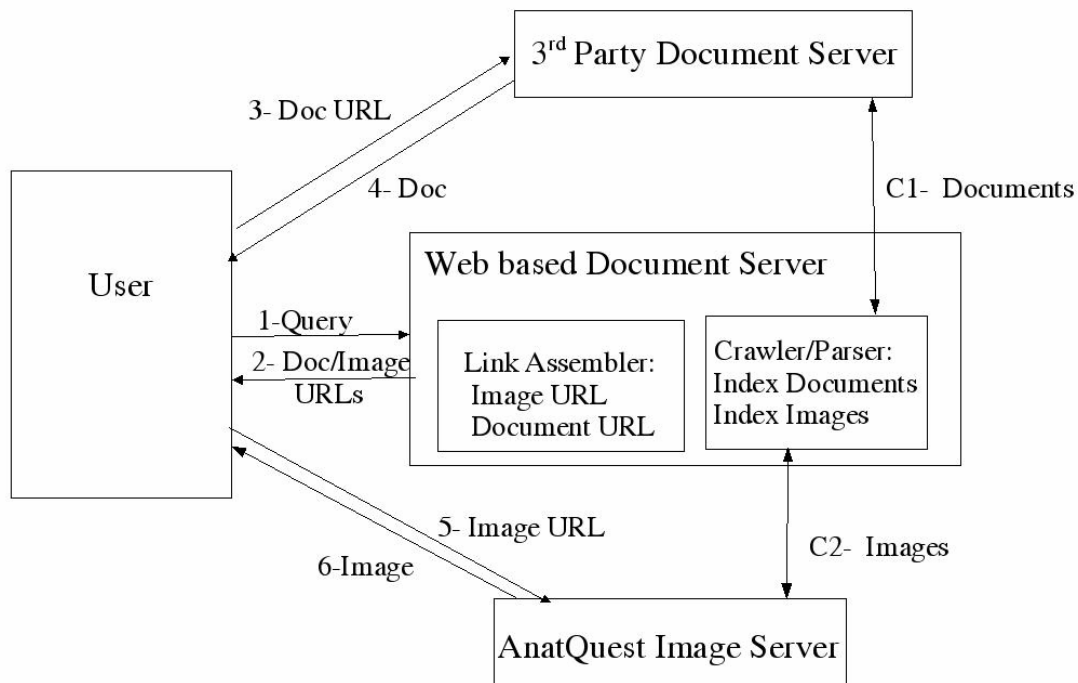


Figure 4.13. Link Assembler – portal approach

As shown in the diagram, steps C1 and C2 consist of crawling the document and the image files and generating indexes for the images and documents. The indexes are used by the link assembler component.

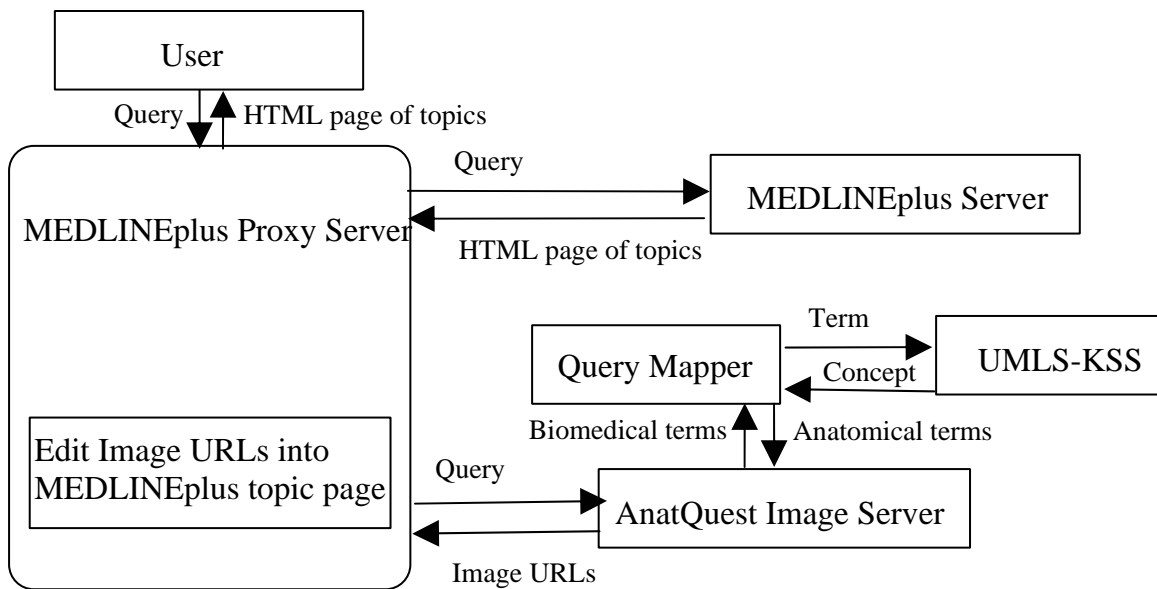
The advantage of this approach is that it requires no changes to Web based document servers such as MEDLINEplus. Its drawback is that it only provides images related to the user query and not those related to the contents of the documents.

A comparison of the advantages and drawbacks of each of the three alternatives for linking terms to images suggests that the best approach would be to combine the portal and the Related-Images alternatives. This approach not only provides the images matching the user queries (portal approach) but it also allows the user to view the images related to the contents of the selected documents. Further, when the metadata in the relevant documents include MeSH headings, this approach avoids the parsing requirement of the hotlinking approach.

4.4.3 Implemented prototype: MEDLINEplus proxy server

We have implemented a prototype of a text-to-image linking system based on the MEDLINEplus health information Web server. As shown in Figure 4.14, a proxy server has been developed to intercept the user request to MEDLINEplus. The proxy server first retrieves the MEDLINEplus page that satisfies the user query. In parallel, the proxy server sends the user query to the AnatQuest image server which uses the UMLS Knowledge Source Server and a query mapper module to map the query terms (mostly disease terms) to the corresponding anatomical structures. The links to these images are then inserted by the proxy server as hotlinks in the image section of the MEDLINEplus page, which is then returned to the user.

In a variation of the prototype, instead of parsing the text, we could utilize the MeSH terms metadata attached to the document, if these exist, since these terms may be assumed to represent the focus or topic of the document. The documents are then linked to the images through their MeSH terms.



MEDLINEplus Proxy Server Architecture

Figure 4.14.

Our initial prototype serves to demonstrate the feasibility of three of the four functions necessary for text-to-image linkage: viz., the term mapper, image locator and link assembler. It does not address the document analyzer stage. Instead, the user query, rather than the contents of the returned document, is mapped to the anatomic term which then is linked to the image. Even the functions included are not addressed comprehensively, as for example, the term mapper uses only the location-of relationship. Nevertheless, this

prototype provides a platform on which a more extensive testbed may be designed to address research issues related to the larger problem of linking text in documents to images. Some of these issues are:

- 1 Relevance of images to the topics in a document. This is influenced by the types of mappings used by the term mapper. For example, would the location-of relationship used in this prototype result in the display of appropriate images (rather than those of a healthy lung) for a document about pneumonia? This measure is subjective, but may be quantifiable by means of user studies.
- 2 Precision and recall of images. These well-defined and quantifiable measures reflect the performance of the image retriever searching the metadata in order to locate desired images.
- 3 Accuracy and precision of the links. This is of particular importance to the hotlinking approach for the display of links where the document terms are hotlinked to the images. This approach might raise user expectations for an "exact image" rather than a "relevant image". A link may be seen as relevant but not necessarily "accurate".
- 4 Term mapper performance. Considering additional types of anatomical relationship mappings (e.g., through part-of relationships) should increase recall. In addition, a ranking of the individual mappings should provide a measure for ranking the resulting anatomical terms.
- 5 Parser performance. In addition to extracting the anatomical terms, identifying the explicit anatomical relations in the document enables formulation of more refined search queries against the image database, which should result in a higher relevancy measure.
- 6 System performance. Some approaches require more resources than others, which could translate into higher turnaround time. For example, the hotlinking approach requires modifying each document before it is sent to the user. Turnaround time, excluding the network delay, should provide a measure of system performance.

Analyzing the collected data should provide guidance for the best combination of the alternative solutions in each of the four functions described in section 4.4.2 for an optimum solution for linking document text to images.

5 Summary

The AnatQuest Project aims to facilitate the access to, and the retrieval and display of, Visible Human images from our image database. We have developed the AnatQuest system for Web access by resource-limited end users, the AnatQuestKiosk system for onsite visitors, and a prototype system demonstrating the linking of user queries to MEDLINEplus to anatomic images. We have also outlined further investigation required to enable the linking of the text content of documents retrieved by MEDLINEplus to anatomic images in our database.

References

1. Programs and Services, Fiscal Year 2001, National Library of Medicine, NIH, Bethesda, MD, p. 1.
2. Henderson E, Seamans J, Strupp-Adams A. VHIF: A prototype file format for anatomical images. Proc. Visible Human Conference, 1998. (CD-ROM from NLM).
3. Strupp-Adams A, Henderson E. Retrieving high resolution images over the Internet from an anatomical image database. Proc. SPIE: Internet Imaging, Jan. 2000, Vol. 3964, 259-65.
4. Zhou R, Henderson E, Seamans J. Visualization of Visible Human anatomic images. Proc. Visible Human Conference, 1998. (CD-ROM from NLM).
5. Thoma GR, Long LR. Compressing and transmitting visible human images. IEEE MultiMedia 1997 April-June; 4(2): 36-45.
6. Meadows S, Thoma GR, Long LR, Mitra S. Entropy encoding of difference images from adjacent Visible Human digital color photographic slices for lossless compression. In: Kim, Yongmin, editor. Medical Imaging 1997: Image Display. SPIE Vol. 3031, 1997; 749-55.
7. Long LR. Transmission of medical images over wide area networks. Proc. Visible Human Conference, 1996. (CD-ROM from NLM).
8. Mitra S, Long LR, Pemmaraju S, Muyschondt R, Thoma GR. Color image coding using wavelet pyramid coders. Proc. SSIAP'96, San Antonio, TX, April 1996; 52-63.
9. Pemmaraju S, Mitra S, Long LR, Shieh Y-Y, Roberson G. An adaptive vector quantization with fuzzy distortion measure for image coding. Proc. SPIE Medical Imaging '96, Newport Beach, CA, February 1996; 112-5.
10. Long LR, Berman LE, Neve L, Thoma GR. An applications-level technique for transmission of large images on the Internet. Proc. SPIE: Multimedia Computing and Networking 1995, Vol. 2417, San Jose, CA, February 1995, 116-29.
11. <http://www.vtk.org/index.php>
12. Schroeder W, Martin K, Lorensen B. The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics. 2nd edition. Prentice Hall, 1998, 645 pp.
13. http://www.nlm.nih.gov/exhibition/dreamanatomy/da_technology.html
14. Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: The MetaMap program. Proc AMIA Symp 2001:17-21.

14a. Rindflesch TC, Aronson AR. Semantic processing for enhanced access to biomedical knowledge. Kashyap V, Shklar L (eds.). Real World Semantic Web Applications, 2000, IOS Press, 157-72.

15. Portable Network Graphics: <http://www.w3.org/Graphics/PNG>

16. Dublin Core Metadata Initiative: <http://dublincore.org/>

17. HotMeta: <http://archive.dstc.edu.au/RDU/HotMeta/png/index.html>.

Curriculum Vitae

George R. Thoma

Chief, Communications Engineering Branch, LHNCBC, NLM

Education and Training:

Swarthmore College, Swarthmore PA	B.S. 1965	Electrical Engineering
University of Pennsylvania	M.S. 1967	Electrical Engineering
University of Pennsylvania	Ph.D. 1971	Electrical Engineering

Research and Professional Experience:

1984-present, Chief, Communications Engineering Branch, LHNCBC, NLM

Directs and conducts R&D in projects involving image processing and communications engineering. Project descriptions and results appear in the published literature (see Publications), and some are synopsized in the branch's Web page <http://archive.nlm.nih.gov>. Projects include the following:

*MARS (Medical Article Records System): a system to automate the entry of bibliographic records from journal articles. This system based on document image analysis and understanding techniques is in production; current research into expanding image analysis to process online journals.

*DXPNET, WebMIRS: an Internet-accessible multimedia database of digitized x-rays and associated text; content-based image indexing; online digital x-ray atlas.

*Image compression techniques applied to the highly data intensive 24 b/p color cryosection Visible Human image collection.

*AnatQuest: end-user access to a database of Visible Human cryosection and rendered images.

*DocView: research in document imaging for end-user Internet access to electronic documents.

1974-84, Senior Electronics Engineer, Communications Engineering Branch, LHNCBC, NLM

Conducted R&D in communications systems and signal processing techniques applied to electronic document storage, retrieval and display; biomedical image data compression, text recognition and image enhancement, satellite communications, echo suppression, videodisk mastering, motion-adaptive video compression.

1973-74, Systems Engineer, General Electric Co., Space Division, Valley Forge PA.

1971-73, Systems Engineer, AII Systems, Moorestown NJ.

1968-71, Ford Foundation Fellow, University Research Fellow and Post-doctoral Research Associate, University of Pennsylvania, Philadelphia

Honors:

* General Chair, 14th IEEE Symposium on Computer-Based Medical Systems, 2001

* Fellow of the SPIE (elected 1999)

* Awards: NIH Merit Award 1998, NLM Regents Award 1998, Federal Computer Week's *Federal 100 Award* in 1995, NLM Staff Recognition Awards (1997-2002); On-the-spot Award (1998); Special Act/ Service Award (1997); Special Act Group Award (1992); Merit Awards (1984-97).

- * Appointed to Internet2 Applications Strategy Council (2002).
- * Appointed to Maryland Governor's Task Force on High Speed Networks (1998- present).
- * Keynote speaker at SPIE Electronic Imaging, San Jose CA, February 1996; and at 8th IEEE Symposium on Computer-based Medical Systems, Lubbock TX, June 1995.
- * Appointed to Blue Ribbon Panel at National Institute for Standards and Technology for document file format standards (1994-95).
- * Invited referee for: IEEE Journal on Special Areas in Communications; IEEE Computer Journal; IEEE Parallel and Distributed Technology; Information Processing and Management Journal; Journal of Clinical Engineering; SPIE and IEEE conferences.
- * Association of Image Information Management Certificate of Service (1995)
- * American College of Physicians Certificate of Accomplishment (1993)
- * NASA Certificate of Appreciation (1993)

Publications (most recent only):

Ford G, Thoma GR. Ground truth data for document image analysis. Proc. 2003 Symposium on Document Image Understanding. College Park, MD: Institute for Advanced Computer Studies, University of Maryland. April 2003; 199-205.

Hauser SE, Sabir TF, Thoma GR. OCR correction using historical relationships from verified text in biomedical citations. Proc. 2003 Symposium on Document Image Understanding. College Park, MD: Institute for Advanced Computer Studies, University of Maryland. April 2003; 171-7.

Kim J, Le DX, Thoma GR. Automated labeling of bibliographic data extracted from biomedical online journals. Proc. SPIE Electronic Imaging, Jan 2003. SPIE Vol. 5010: 47-56.

Hauser SE, Schlaifer J, Sabir TF, Demner-Fushman D, Thoma GR. Correcting OCR text by association with historic datasets. Proc. SPIE Electronic Imaging, Jan 2003. Vol. 5010: 84-93.

Antani S, Long LR, Thoma GR, Lee DJ. Evaluation of shape indexing methods for content-based retrieval of x-ray images. Proc. SPIE Electronic Imaging: Storage and Retrieval for Media Databases, Jan 2003. Vol. 5021:405-16.

Thoma GR, Ford G, Le DX, Li Z. Text verification in an automated system for the extraction of bibliographic data. Proc. 5th International Workshop on Document Analysis Systems, Springer-Verlag: Berlin, Aug 2002, 423-32.

Le DX, Straughan SR, Thoma GR. Greek alphabet recognition technique for biomedical documents. Proc. 6th World Multiconference on Systemics, Cybernetics and Informatics, Vol. III, eds: Callaos N, et al; July 2002, 86-91.

Walker FL, Thoma GR. A SOAP-enabled system for an online library service. Proc. InfoToday 2002, Medford NJ: Information Today, May 2002, 320-9.

Long LR, Krainak DM, Thoma GR. Identifying image structures for content-based retrieval of digitized spine x-rays. Proc. SPIE Medical Imaging 2002: Image Processing, Vol. 4684, 118-26.

Curriculum Vitae

Amir Razi

Computer Scientist

Education and Training:

School of Planning and Computer Applications	B.S. 1977	Computer Science
Pennsylvania State University	M.S. 1980	Computer Science
University of Delaware	Ph.D. 1985	Computer and Information Sciences

Research and Professional Experience:

January 1994- present, Computer Scientist, Lister Hill National Center for Biomedical Communications (LHNCBC-NLM-NIH): A lead designer for the Digital Imaging Database System of the Visible Human Project. The system provides capabilities for the retrieval and the display of human anatomical images from a 3D image management system. The system utilizes an object-oriented database system with a home grown file format for efficient packaging and retrieval of segmented and labeled images.

March 2001-March 2002, Computer Scientist, National Cancer Institute (NCI-NIH): Provided design and implementation support for the terminology component of the CancerNet, a web site for dissemination of cancer information. The terminology module facilitates search and categorization of documents.

December 2001-June 2002, Computer Scientist, Center for Information Technology (CIT-NIH): Performed a feasibility study for bringing all of the NIH diverse administrative web resources under the umbrella of a single NIH web portal.

January 1991-March 2001, Computer Scientist, LHNCBC-NLM-NIH: A lead designer and implementer of the UMLS Knowledge Source Server for concept-oriented retrieval of biomedical information. The system provided access to each of the Unified Medical Language System (UMLS) Knowledge Sources: the Metathesaurus, the Semantic Network, the Information Sources Map, and the Specialist Lexicon. The system's API supported an object model with methods for accessing XML encoded data using Java's Remote Method Invocation (RMI).

1988-Present, Adjunct Faculty, Johns Hopkins University, School of Engineering: Teaching graduate courses in Artificial Intelligence.

1984-1990, Senior Computer Scientist, Software A&E, Virginia. The principal investigator for projects in Natural Language Processing and Knowledge-Based Systems. The major projects included a concept-based text analysis system and a software specification environment for building intelligent systems.

1980-1984, Research Programmer, University of Delaware. Designed and built a Natural Language Understanding system which adapted to the idiosyncrasies of the user's language.

1978-1980, Application Programmer, Department of Business Logistics, Pennsylvania State University. Built and maintained a statistical database system.

Professional Societies: ACM, IEEE, AAAI

Curriculum Vitae

James R. Seamans

Senior Systems Scientist

Education and Training:

Ricker College	BS	1968	Mathematics
DeVry Institute of Technology	AS	1978	Electronics & Computer Controls

Research and Professional Experience:

Mr. Seamans is the senior developer on the National Library of Medicine's (NLM) R&D project, the Visible Human Project. The project design incorporates the world's first computer-based images of human beings. Two cadavers, one male and one female, are being imaged from head to toe with Cyrosections, CT scan, and MRI imaging. The bodies were frozen and sliced cross-sectionally at one millimeter intervals, and images captured using CCD, 35mm, and 70mm film cameras. The project is re-creating the cadavers' computer images for research with 3-D Volume Rendering techniques. Mr. Seamans is directly responsible for developing the 3-D imaging software that researchers use to view the Visible Human Project Dataset. Java language and Java Advanced Imaging APIs are being employed for cross-platform development. He provides expertise in Object Oriented Database Design and Implementation utilizing C++. Additionally, he provides systems administration support for all systems and development teams for the project. Mr. Seamans has utilized XML, Perl, and Unix Shell Scripts; and he has supported a number of diverse environments including Sun Solaris, Windows/NT, Windows, Macintosh, Mac OS, Unix and Windows workstations. Mr. Seamans also has experience with disk arrays and DLT tape backup jukeboxes.

Mr. Seamans was the senior technical developer on the X-Windows Color Medical Imaging System (XCMIS). This system was a prototype medical imaging system for capturing and displaying microscopic slides. Using digital techniques, after capturing the slides, the user is presented a computer-driven representation of a microscope with full-function capabilities. XCMIS was used by the Centers for Disease Control and Prevention (CDCP) to develop their own CytoView, *A Prototype Computer Image-Base Papanicolaou Smear Proficiency Test*, which was published in *The International Academy of Cytology*, November–December 1999.

Publications:

Earl Henderson, James Seamans, Diane Soloman, MD, "Electronic Capture and Conversion of Microscopy Images from Cytologic Glass Slides", *American Society of Clinical Pathologists, "Laboratory Medicine"*, Vol. 25, No. 4, April 1994.

Earl Henderson, James Seamans, "Image Acquisition of Microscopic Slides", *The International Society for Optical Engineering, Symposium on Electronic Imaging Science & Technology*, February 1994.

James Seamans, Earl Henderson, "Network Distribution of the Visible Human Data Set", *The Second Visible Human Project Conference*, October 1998, CDROM from NLM.

Earl Henderson, James Seamans, Annette Strupp-Adams, "VHIF: A Prototype File Format for Anatomical Images", The Second Visible Human Project Conference, October 1998, CDROM from NLM.

Ruixia Zhou, Earl Henderson, James Seamans, "Visualization of Visible Human Anatomical Images", The Second Visible Human Project Conference, October 1998, CDROM from NLM.

Curriculum Vitae

Annette Strupp-Adams

Senior Systems Analyst, MSD Inc., Vienna, VA

Education and Training:

University of Maryland,

University College, College Park, MD

B.S. 1994

Computer Science

Johns Hopkins University, Baltimore, MD

M.S. Expected 2004

Computer Science

Research and Professional Experience:

1996 to present, Senior Systems Analyst, MSD Inc., Vienna, VA.

Designed and implemented 3-tier Internet client/server user-interface, network, and application programs in Java (Servlets, SAX, XML, JFC Swing, RMI, Java Advanced Imaging API), Perl, and C++ for the implementation of the Visible Human 3DSystems Imaging Project to submit user queries via an Apache Web Server/Tomcat servlet engine to an Image Database. Configured Apache Web Server/Tomcat servlet engine by writing shell scripts, cron jobs. Implemented Web Server statistics program.

*Linking text resources to an image database: Implemented prototype: MEDLINEplus proxy server.

*AnatQuest: Implemented image viewers to navigate and display the Visible Human body via a Web browser.

*AnatQuest for the Kiosk: An onsite version of the AnatQuest Image Viewer version for the Dream Anatomy exhibit at the NLM.

January 1996-August 1996, Software Developer, Dow Jones & Co., Inc., Federal Filings, Washington, DC

Wrote C,C++, and Perl programs on a Unix Sun SPARC station platform to access a Sybase SQL database.

1995-1996, Programmer/Analyst, Federal Information Exchange, Inc., Gaithersburg, MD

Wrote 3-tier Internet user-interface CGI programs in Perl/C++ to communicate with full-text retrieval search engines to access a GUPTA SQL database.

1994-1995, Intern Software Developer, Personal Library Software, Rockville, MD

Wrote Internet client/server CGI programs in Perl and C to use in-house full-text search engine API's to embed queries to Web server documents in HTML tags, and to perform client/server database administration functions via Internet GUIs. Platform: HP-UX under Solaris 3. Software tools used: Clearcase 2.0, vi, make, dbx, tcsh shell.

Curriculum Vitae

Linglan Edwards

Senior Software Engineer, Management Systems Designers, Inc.

Education and Training:

Southeast University, P.R. China	B.S.	1989	Computer Science
University of Maryland, Baltimore County	M.S.	1994	Computer Science
University of Maryland, Baltimore County	Ph.D.	1998	Computer Science

Research and Professional Experience:

1999-present, Senior software engineer, MSD, Inc.

Working on research and development projects in image data processing and visualization, including: Visible Human Database, AnatLine and AnatQuest.

1999, Post-doctoral research fellow, LHCBC, NLM.

Researched on large scale image data processing and storage. Designed and developed software for the 3D System Project, such as VhiBuilder, VhParser.

1997-1998 Research assistant, University of Maryland, Baltimore County.

Designed and developed scientific and information visualization software. Collaborated with the Johns Hopkins University Medical Center on VOLREN, a medical image rendering and visualization system. Designed and developed intelligent software agents. Developed computational models for biological simulation.

1995-1997 Teaching assistant, University of Maryland, Baltimore County.

1995 Software engineer, SHL Kee Systems, Inc.

1993-1994 Teaching assistant, University of Maryland, Baltimore County.

1992-1994 Research assistant, University of Maryland, Baltimore County.

1989-1992 Software Engineer, Jianghai Inc., Nanjing, P.R.China

Honors:

Upsilon P Epsilon Honor Society for Computing Sciences.

Publications:

Edwards, L., Peng, Y., "Computer Simulation of Lipid Aggregates", accepted for oral presentation at The Third German Workshop on Artificial Life", 1998.

Edwards, L., Peng, Y., "Computational Models for the Formation of Protocell Structures", presented as a plenary talk at The Sixth International Conference on Artificial Life, June 1998.

Edwards, L., Peng, Y., "Computational Models for the Formation of Protocell Structures", An extended version is published in a special issue of the Journal of Artificial Life, 1998.

Ebert, D., Heath, D., Kuszyk, B., Edwards, L., Shaw, C., Kukla, J., Bedwell, E., Fishman, E., "Evaluating the Potential and Problems of 3D Measurements of Arterial Stenosis" accepted for publication, Journal of Digital Imaging, 1998.

Edwards, L., Peng, Y., "Computational Models for the Formation of Protocell Structures", poster presentation at The Second Annual International Conference on Computational Molecular Biology, March 1998.

Heath, D., Ebert, D., Kuszyk, B., Zhang (Edwards), L., Shaw, C., Kukla, J., Bedwell, E., Fishman, "Estimation of stenotic degree using volume rendering " (abstract) 83rd Scientific Assembly and Annual Meeting of the Radiological Society of North America (RSNA), November 30 to December 5, 1997, Chicago, IL.

Curriculum Vitae

Jinghong Gao

Senior System Analyst, Management System Designers, Inc

Education and Training

Wuhan Technical University,

Wuhan, China

B.S. 1992

Photogrammetry/Remote Sensing

Bowie State University, Bowie, MD

M.S. 1998

Computer Science

Research and Professional Experience:

1998-present, Senior System Analyst, Management System Designers, Inc

Ms. Gao supports the Visible Human Project of the National Library of Medicine's (NLM) Lister Hill National Center for Biomedical Communications (LHNCBC) as a Senior Systems Analyst. Ms. Gao developed and now maintains a medical image database for the storage of large Visible Human image dataset along with the corresponding anatomical structure information. The database is built by using ObjectStore Object Oriented Database Management Engine and implemented using Java JDK1.4. Ms. Gao also designed and developed a data retrieval module to retrieve medical images from the database through the Internet. The module includes a set of image processors to extract and compress large Visible Human images for online viewing using Java Advanced Imaging API and an image database server to retrieve the required image from the medical image database using Java Servlet and/or Java RMI. She also participated in design and implementation of web based Visible Human image viewers using Java Applet and HTML and the server side applications using Java Server Page, Enterprise Java Bean and XML.

1997-1998, Research Assistant

Center for Research in Distributed Computing, Bowie State University, Bowie, Maryland

Agent based distributed intelligent system design and development for NASA's USIA project. Projects include: Natural Language Processor: a natural language parser to parse plain English commands and transfer it to a machine understandable format using Java JDK1.1.

1992-1995, Software Engineer,

National Information Center of Surveying and Mapping, Beijing, China

Geographic Information System and image processing software design and development. Projects include:

Remote Sensing Image Orientation Toolkit using FORTRAN.

Cancer Cell Detection System using C.

Honors:

Academic Achievement Award, Bowie State University, 1998.

Outstanding Research Project Award (1st place), Bowie State University, 1997.

Curriculum Vitae

Miyoung T. Ledoux

Biomedical Imaging Researcher, LHNCBC, NLM

Education and Training:

Johns Hopkins University	PhD candidate	present	Public Health
Johns Hopkins University	MS	1996	Info. Sys. and Telecom
Johns Hopkins University	MA	1992	Medical Illustration
McGill University	BS	1989	Human Physiology

Research and Professional Experience:

The Visible Human Project (1992 - present):

- Image data analysis
- Imaging technology: digitization of the 70 mm film
- 3D visualization of the anatomic structures

Anatline and AnatQuest: a database of Visible Human images and 3D rendered anatomic structures, 1998 – present.

3D visualization of the connection between neck muscles and the dura in the Visible Humans by Mi Young Toh, Gary D. Hack, DDS, and Gwendolyn Dunn, DDS, University of Maryland College of Dental Surgery and Dental School, 1996 - 1998.

Interactive Brain Atlas Using the Visible Human Project Data, produced by Mi Young Toh Mark E. Molliver, MD, and James Main, 1995 - 1998.

An Application of Visible Human Data: Hip Joint Replacement Surgery (video), produced by APDB, Lister Hill National Center for Biomedical Communications, National Library of Medicine, National Institutes of Health, Bethesda, Md, 1994 - 1995.

Structure of the Human Brain (interactive digital atlas), produced by Mi Young Toh and Mark E. Molliver, MD. Departments of Neuroscience, Johns Hopkins University, School of Medicine, Baltimore Md, 1992 - 1993.

The Reconstruction of 3D Ventricular System on Macintosh, produced by produced by Mi Young Toh and Mark E. Molliver, MD. Departments of Neuroscience, Johns Hopkins University, School of Medicine, Baltimore Md, 1991 – 1992.

Publications:

Hack GD, Dunn GF, Toh M, The Anatomist's New Tools, Encyclopedia Britannica Medical 1997.

Toh M, Falk RB, Main JS, Interactive Brain Atlas Using the Visible Human Project Data: Development Methods and Techniques, Radiographics, 1996;16:1201-1206.

Martin JG, Xu LJ, Toh MY, Olivenstein R, Powell WS. Leukotrienes in Bile during the Early and the Late Airway Responses after Allergen Challenge of Sensitized Rats. *Am Rev Respir Dis* 1993; 147(1):104-110.

Abstracts:

Toh M, Banvard R, Ackerman M, Rodgers R, Main JS, The Visible Human Project Web Page, Radiological Society of North America, 1996.

Toh M, Falk RB, Horowitz S, Main JS, Digital Brain Atlas Using the “Visible Human Project” Data., Visible Human Project Conference, 1996.

Hack GD, Raitu P, Kerr JP, Dunn GF, Toh M, Wilson DR, Visualization of the Muscle-Dural Bridge in the Visible Human Dataset, Radiological Society of North America 1996

Toh M, Falk RB, Horowitz S, Main JS, Three-Dimensional Visualization of the “Visible Human Project” Data, Radiological Society of North America Annual Conference, 1995.

Toh M, Ackerman MJ, Main JS, Interactive Brain Atlas with the “Visible Human Project” Data, Radiological Society of North America Annual Conference, 1995.

Xu LJ, Toh M, Olivenstein R, Martin JG, Powell WS. Leukotriene Metabolite in Bile of Sensitized Rats Following Tracheal Instillation of Antigen. *American Thoracic Society and American Lung Association* 1991.