MultiVIS™: A Web-based interactive remote visualization environment and navigable volume imagemap system

M. Doyle, G. Klein, F. Hussaini, and M. Pescitelli
Eolas Technologies Incorporated (www.eolas.net)

This work represents the convergent evolution of a number of technologies and research “threads.” A project called MetaMAP, which developed early hypermedia imagemap technology, dates back to 1986. Work on creating a new paradigm for doing client-server visualization over the Internet began in 1992. Another major project began in 1993 to turn the Web into a platform for interactive aplications. A project to develop multidimensional imagemap technology began in 1995. Finally, work on a scalable computational server architecture called "Dark Iron" began in 1997. The MultiVIS project represents the intersection of these various research efforts to create a new kind of navigable knowledgespace that leverages the advantages of each of its constituent technologies.

The seed of the idea began as early as 1984, during discussions between Mike Doyle and Maurice Pescitelli. Dr. Doyle was then a graduate student at the University of Illinois, enrolled in Dr. Pescitelli's course on embryological development. The two were computer hobbyists and wondered if it would be possible to aid in the teaching of the difficult 3-D concepts of embryology by making a computer system to build virtual models of embryos that students could interact with on the computer. Little did they know that this question was the beginning of a long and winding road.

The two collaborated on a grant application to fund a project, hoping to use the powerful new IBM AT computers that had just come out on the market to build such a system. The proposal was funded, but, luckily, the project fell apart for political reasons, before the team could discover that they had gotten in way over their heads, and that the complexity of the task would have swamped such a small computer system.

The seed had been planted, however.

MetaMAP®

Shortly afterward, Doyle was working as a teaching assistant in a Medical Histology class. He noticed that most of the student's questions in lab were a matter of putting the microscope's pointer on a particular view of a microscopic view and asking a TA "what is this?" He thought it might be possible to create a system that could act as a “virtual TA” to automate answering such questions via a computer screen. He set out to build such a system as a “hobby” project using his early IBM PC.

He had a third-party graphics board for the that could display images that were 512x480 resolution, and able to handle 256 colors from a 16 million color palette. Doyle had gained experience in working with limited palettes due to his training as a medical illustrator. It seemed to him that 256 colors was much more than he would need to render a realistic image of a microscope slide. He began to think about using excess capacity in the 8-bit pixel to store object information, and then to use this as a basis for a program to automatically identify image features in response to users' mouse clicks. He made a small “browser program” that allowed a user to navigate from image to image and to retreive text information about image features merely by clicking on "htspots" in the images. His new technology was awarded a U.S. patent (4,847,604) in 1989. [1, 8]

A unique aspect of this program was that this was an “open” hypermedia system, in that it allowed “webs” of information to be created and navigated that could be arbitrarily large, since it used image files and "secondary maps" that were external to the navigation program. Unlike the previously-used “closed” hypermedia systems that required the hypermedia application to store a complete representation of the network of hypertext links within the program’s address space, and resulted in large unweildy programs needed to handle large databases, Doyle's system could use a small browser program to allow a user to navigate from image to image throughout a potentially infinitely-large web of interconnected data. As far as the authors are aware, MetaMAP was the first open distributed hypermedia system, predating the World Wide Web by three years. It was certainly the first Web-like hypermedia imagemap system, employing a true client-side imagemap architecture in the mid-1980s.

Since then, the Web has provided a simple mechanism, called an imagemap or ISMAP, for linking 2-dimensional spatial data (images) to related symbolic information (URLs). This is a simple technology that links simple polygonal regions within images to the locations of data objects on the Internet. Web imagemaps are currently the standard mechanism that is used for creating graphically attractive user interfaces to Web pages. Unfortunately, the standard polygon-based ISMAP technology that is used in most Web imagemap systems can only work with simple polygon maps, and becomes either intolerably slow or totally unusable for mapping high-resolution images with large numbers of irregularly shaped objects, such as for complex biomedical images. [6, 9]

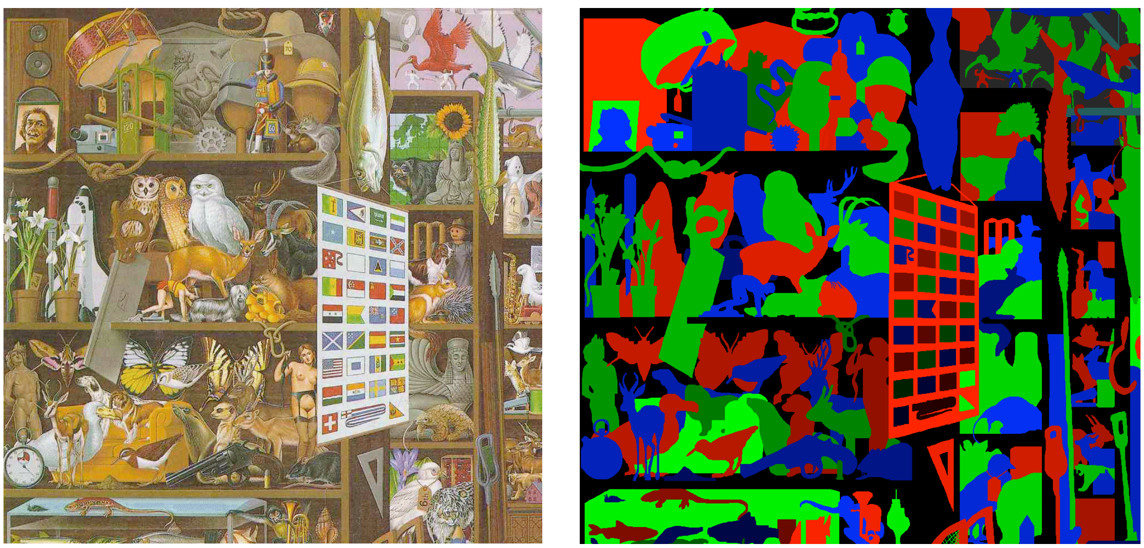

Figure 1: A very complex Web-based imagemap made possible by Doyle's MetaMAP technology

For this reason, Doyle's MetaMAP system was adapted to the current Web imagemap architecture. The efficiencies that allowed hypermedia imagemaps on 4.7 MHz IBM PCs in 1986 allowed the creation of incredibly-complex imagemaps on newer hardware platforms. Current implementations of MetaMAP allow as many as 16 million different classes of hotspots within a single image, where even each pixel can be an independently-adressable hotspot (for images of fewer than 16 million pixels). Figure 1, above, is an example of what is probably the most complex 2-D imagemap on the Web today (viewable at http://demo.eolas.com/metamap/bigdemo.htm). The left frame shows a very complex scene full of many objects. Figure 1 actually shows only one-fourth of the full imagemap that is viewable online. The right frame shows the MetaMAP-based secondary map which correlates each image object to a unique 24-bit object index, where each object index appears in the map as a unique 24-bit color.

Visible Human Nameserver

The MetaMAP system was recently usedby MuriTech Inc. to create an anatomical object table which corresponded to the 2000 anatomical features represented in the Visible Human Project image dataset. [10] This table correlated the 24-bit object index with an appropriate server response. The MetaMAP server allows script-based programs to be associated with image objects. Initially, a simple html script is associated with each object. In the future, however, more sophisticated programmed responses can be associated with each anatomical object merely by setting a flag in the object's table record and inserting an appropriate Tcl script in each anatomical object record's "script" field. Using the *z*Map technology described below, the anatomical object MetaMAP images and the related object data table were then used to set up a Visible-Human-specific "anatomical nameserver." This server can be employed as a general-purpose information server by anyone on the Internet who has the ability to determine the x,y,z coordinate of any voxel within the VHP male dataset.

A simple HTTP GET request issued in the form of http://209.19.23.149/cgi-dos/mapper2.cmd/vhm@vh1-b@z.tif?x,y will result in the nameserver returning the anatomical object name corresponding to that voxel location in the VHP male dataset. It should be remembered that the first z coordinate in the VHP male data is 1001, so the range of valid z values for the male would be 1001 to 2878. Further, the x and y origin for each section image is offset from the original NLM data by a few pixels in order to correspond to the most commonly used set of VHP image data, which has had excess image data outside of the body cropped. This system can be queried by any conventional Web browser.

For example, issuing the above request with x,y,z values of 764,554,1400 will return "Pulmonary Vein, Right, System: Circulatory Region: Thorax" The complete HTTP request to receive such a response would be http://209.19.23.149/cgi-dos/mapper2.cmd/vhm@vh1-b@1400.tif?764,554. A simple test of the system would be to enter the above query string into a Web browser's URL window and hitting the ENTER key. This should result in the display of the associated anatomical structure name (Pulmonary Vein, Right System: Circulatory Region: Thorax) in the browser's document window.

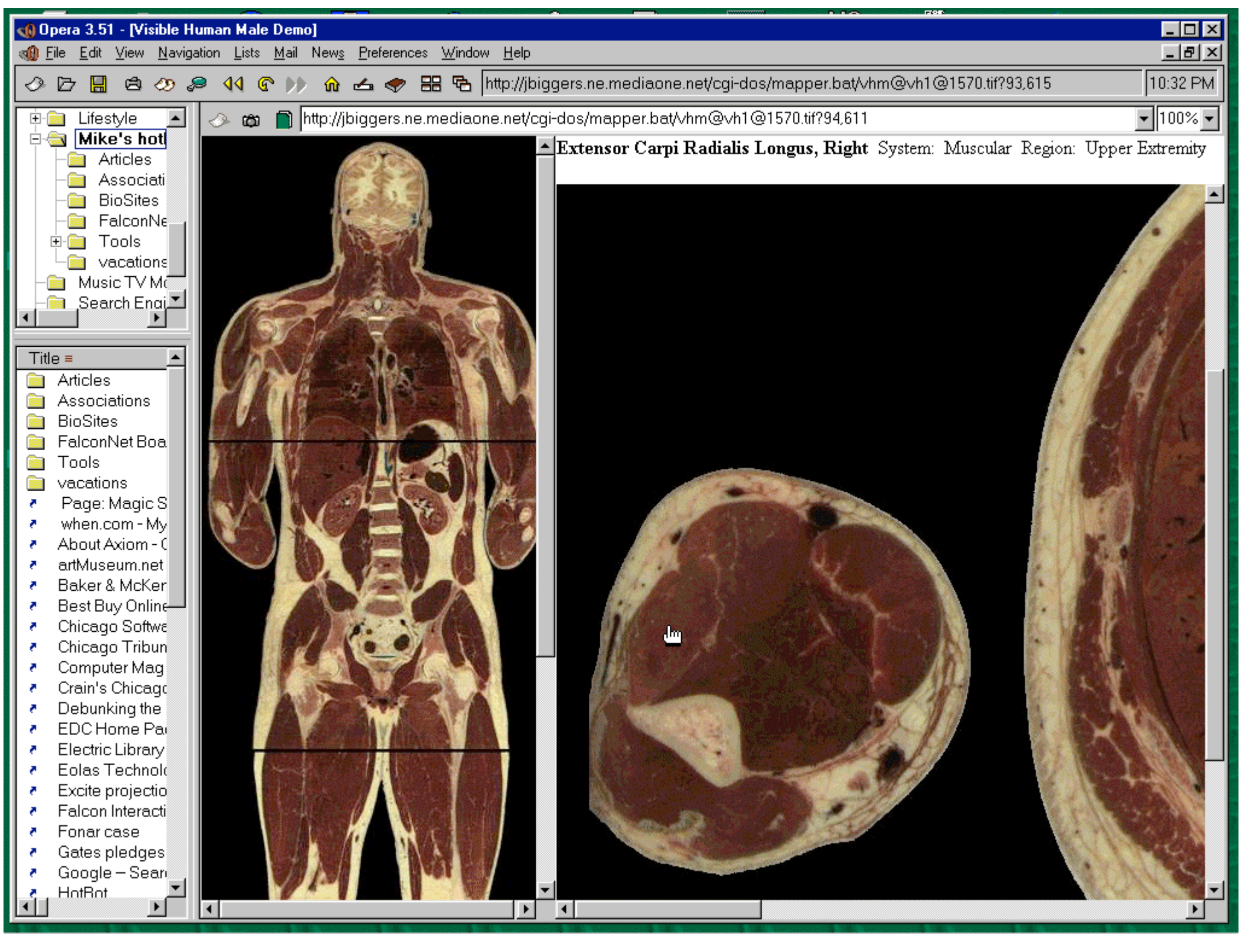

Figure 2: The MuriTech Anatomy Nameserver used to create an atlas of the Visible Human dataset

VIS

In 1989, Dr. Doyle was inspired by the announcement of the National Library of Medicine's Visible Human Project. An intruiging aspect of the project was that it was going to create a single dataset of over 15 GB of 3-D human anatomical volume data. He wondered, "how could anyone work with such a large dataset? Even if one could create a supercomputer application to deal with such data, what good would that do for the average scientist out there?"

To answer this question, his research group set out to create a client-server technology to allow a small "thin" visualization client program that could allow someone to remotely-control a powerful visualization "engine." The group later ame to call this a Visualization Internet Server (VIS) system. An early version of such a system was developed and first publicly demonstrated at SIGGRAPH '93 in Chicago (Figure 3)

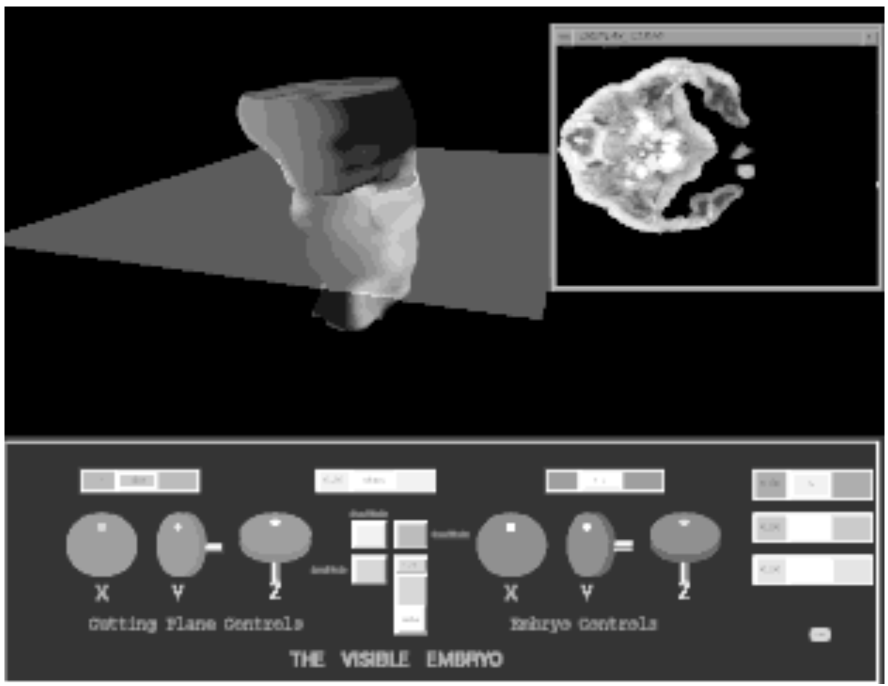

Figure 3: A screenshot of a demonstration of VIS technology at SIGGRAPH '92

The application shown in Figure 3 demonstrated a small client program running on an SGI workstation communicating with a remote visualization supercomputer that could compute an arbitrary slice through a large 3D voxel reconstruction of an embryo. The user could manipulate the controls on the client GUI to cause the visualization server application to compute, in real time, a slice through the dataset at any arbitrary angle. [3, 4]

The UCSF Enhanced Mosaic Application Platform

The group's next task was to devise a way to provide low-cost widespread access to the kind of visualization server "resource" described above. There didn't appear to be any existing way to make rich media such as VIS server-generated visualizations available to a wide population and to make it easy to use for a non-technical audience. NCSA Mosaic had just been developed and demonstrated to a National Science Foundation site visit at the University of Illinois at Chicago in early 1993. Mosaic presented an interesting model : A single web "document" could be put in one place on one server and would immediately become available for viewing by anyone in the world with a copy of Mosaic running on his or her machine. The Mosaic system, however, was designed for navigating through sets of static documents.

After moving to the University of California, San Francisco (UCSF), Dr. Doyle began to investigate extending the capabilities of Mosaic to allow more interactive capabilities. Doyle and his staff realized that by extending Mosaic, they could create a new kind of platform for interactive applications, the kind of platform that they could embed the VIS client into, as well as any other kind of program one would like. He worked with two of his staff members, Cheong Ang and David Martin, to develop a technology which allowed external programs to be "plugged-in" to Mosaic via the UCSF group's new <EMBED...> Hypertext Markup Language "tag,"and run within interactive "live windows" within Web documents.

The VIS client module was then adapted to run as a Mosaic plug-in (Figure 4), allowing a user with a low-end computer running Mosaic to go to a Web page which had a 3D embryo dataset embedded within it, and to interactively rotate the data and adjust such viewing characteristics as tissue translucency and slicing plane. [2] The document and the browser handled the technical parts of getting the data and running the visualization software. All the user needed to do was to fetch a Web page containing an embedded VIS dataset and the user would automatically find himself or herself within a live remote-visualization application, able to interactively rotate and slice through the data at any point. [2, 5, 6, 7]

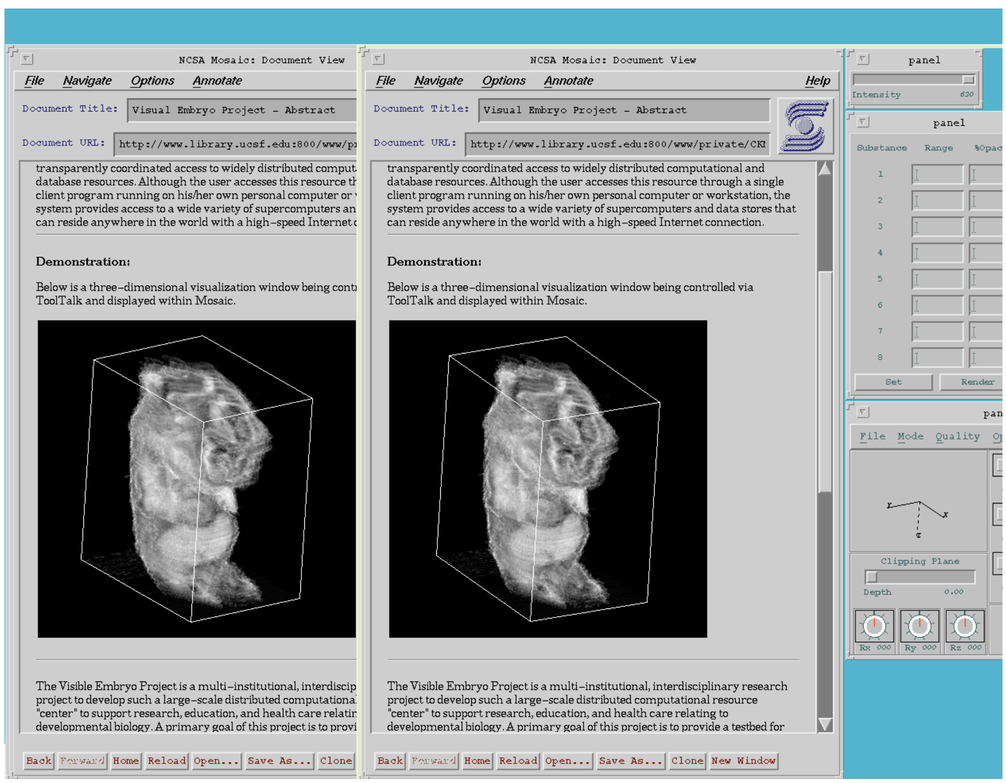

Figure 4: A stereo-pair illustration of the first Web plug-in application, embedded within an NCSA MOSAIC document, showing a 3-dimensional volume reconstruction of human embryonic anatomy. The visualizations were generated in real time by a distributed parallel heterogeneous array of remotely-networked high-end graphics workstations. This technology was developed by Dr. Doyle's research group at the University of California, San Francisco and publicly demonstrated nationwide in 1993 and 1994.

This system was also used to embed 3D MRI data within Web pages for a prototype hypermedia medical records system. Other plug-ins were quickly developed by the UCSF group for molecular modeling and playing MPEG movies within Web pages, as well as a plug-in that was an interpreter for small program code files (Web applets or "weblets") that were downloaded to the Web browser and run within the Web page, allowing interactive "inline" rotation, zooming, and rendering of computer aided design (CAD) models. [7] This new plug-in/applet technology was then followed by the development of similar products by others, such as Sun's Java applet platform, Microsoft's ActiveX Web controls and Navigator plug-ins, which have since become worldwide standards for the creation of interactive Web applications.   The University of California was awarded a U.S. patent (5,838,906) for this technology in 1998, which had been licensed exclusively to Eolas Technologies Inc. (see http://www.eolas.net/press.htm for details). [1]

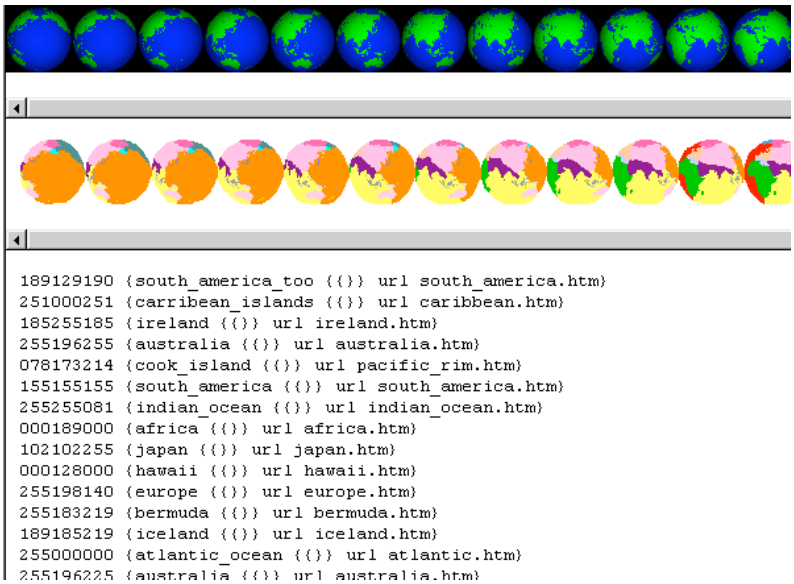

Figure 5: The three datafiles needed to make a client-side *z*Map multidimensional imagemap of a movie of the rotating earth.

The *z*Map™ System

In 1995, Dr. Doyle became interested in the advantages that MetaMAP-based systems might provide with respect to increasing the "dimensionality" of Web imagemap-based media. He realized that the spatial indexing efficiencies of MetaMAP might allow it to be used as a basis for 3-dimensional imagespaces, such as used in virtual reality environments. He began work on an enhancement to the basic MetaMAP system that would allow 3-dimensional imagemaps to be created and used in suport of rich interactive media viewed through Web browsers. This technology, called the *z*Map system, was first designed for implementing moving hotspots for animations and videos, where time is the 3rd dimension. Figure 5 shows an example of the three files needed to deploy a simple *z*Map interactive animation of a rotating earth. This can be deployed as either a client-side or server-side system.

Both server-side and client-side systems use secondary maps of 24-bit color “indices,” where a given x,y,z coordinate in the original image data corresponds to an homologous x,y,z location in the secondary “map” of 24-bit voxels. The RGB triplet found at that location in the secondary map then acts as a unique 24-bit object index for the image feature in the original data. The system then performs a database table lookup for that 24-bit index in order to find the related descriptive text information, which may be in the form of HTML code, a URL pointer to a remote Web resource, a Tcl-based server-side script or Tcl-based applet code.

A good illustration of the *z*Map System in action is the MuriTech 3-D Atlas of Human Anatomy, which was created as a collaboration between Eolas Technologies (www.eolas.net) and MuriTech (www.muritech.com). This system is based on the Eolas MultiVIS platform, described below.

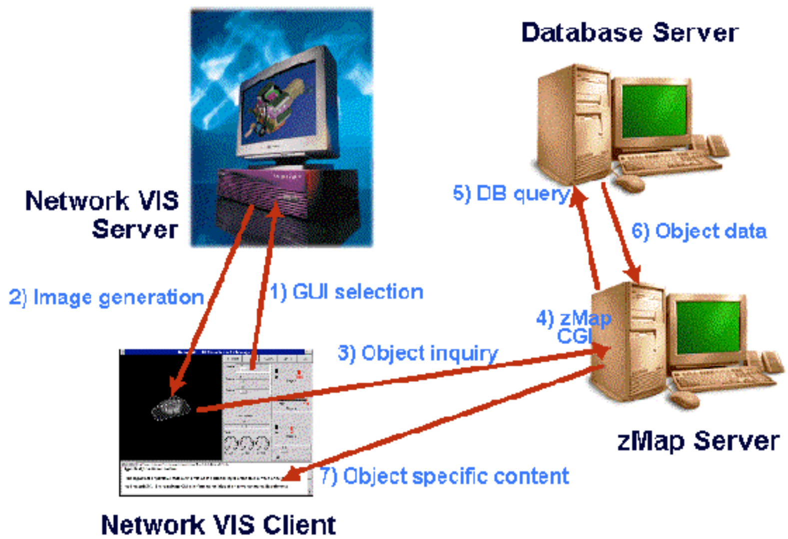

Figure 6: The MultiVIS server architecture integrating visualization server technology with interactive *z*Map indexing of 3-D object information

MultiVIS™

The MultiVIS system represents the intersection of the various research threads described above. It combines the efficient spatial indexing of MetaMAP and *z*Map and the client-server visualization capabilities of VIS technology with the interactive plug-in/applet technology of the Web platform. This system was used by Eolas and MuriTech to create an interactive navigable annotated 3-D atlas of human anatomy, based upon the Visible Human Project male dataset. The "Multi" portion of the MultiVIS name indicates that the VIS platform was enhanced to enable multile simultaneous users, overcoming a single-user limitation of the original VIS system.

The Visible Human 3-D atlas system uses a small (30K) Tcl-based client applet which downloads and runs within the user's Web browser page. This applet opens a socket communications channel to a remote visualization server, which stores and manipulates the 3-D anatomical image data. When the user makes a change to the controls in the applet GUI, such as to specify rotation around an axis, the server then performs that computation on the data and transfers the resultant data display back to the user for viewing the result of the manipulation within the display frame of the client applet. As Figure 2 shows, the user can slice through the data at .any angle and click upon any voxel on any slice surface to cause the associated descriptive text to be shown in the lower widow of the applet. The efficiency of the system is such that object identification and response occurs in nearly "real time" with identification speeds of approximately ½ second over 56KB Internet connections. This system was successfully demonstrated by the applicants at an NIH workshop in August of 1997 [presentation slides viewable at http://www.muritech.com].

A simple demonstration of this kind of 3-D imagemap capability is available in the form of a *z*Map movie-loop "fly through" of a blastocyst dataset [http://www.eolas.com/muritech/nichd/blasto.htm] where the user can click on any portion of the movie while it is playing and the system will identify the anatomical object. A more sophisticated demonstration of the multiuser "Network VIS" system described above can be seen at http://demo.eolas.com/vhm/demo/. Use of this demonstration requires the downloading and installation of the Eolas T-Machine™ Tcl plug-in, which is also available at the www.eolas.net Web site.

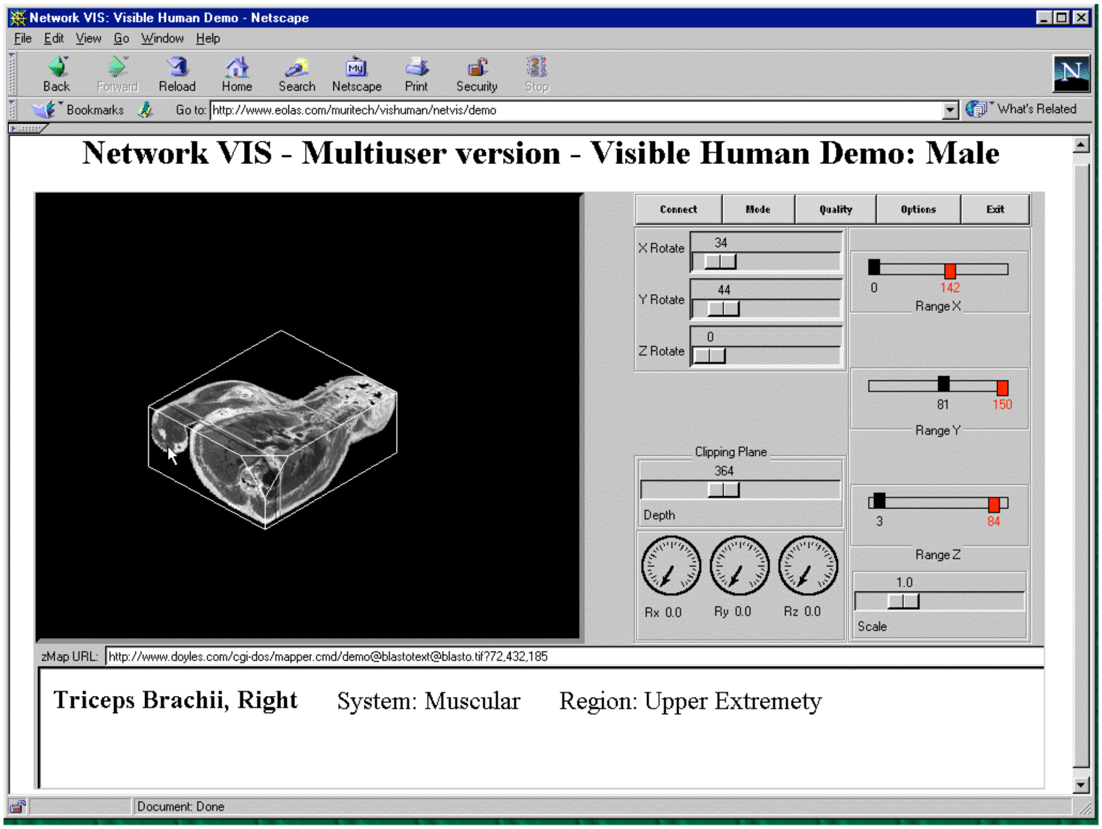

Figure 7: The MuriTech Online 3-D Atlas of Human Anatomy, allowing a user to rotate and slice through the Visible Human male dataset and to click on any structure to have the system identify the anatomcal object being clicked-upon.

Next Steps

Our next goal is to build a full-scale MultiVIS system capable of allowing many simultaneous users to interactively control 3-D visualizations of the voxel-based anatomy data from the VHP male dataset. Since this system is destined to be the basis for an online service, users will demand consistent levels of performance. Delivering such performance consistency is not a trivial matter when considering a system based upon World Wide Web architecture and usage models. A popular Web site can ramp up from just a handful of users to thousands of visitors in very short order. This makes it very difficult to plan computational requirements for mass-market Web-based services. Predicting system requirements based upon preconceptions of potential popularity is a very risky approach. Common mistakes are to overestimate requirements, and wind up with wasted investment in capital equipment, or to underestimate needs, only to find that a larger-than-expected user population experiences uniformly dismal performance.

The solution to this problem will be to use Eolas' Dark Iron™ technology, which represents a server architecture that can grow in capacity as system usage grows, while taking advantage of the visualization-object subscription model in order to anticipate future needs in a consistent manner. In order to handle the variable loads on such a system, a novel auto-scaling architecture will be designed that allows both realtime adjustment of processing capacity and easy expansion of server hardware capabilities.

This system will exploit both distributed parallel processing and server-push technologies to allow the server to automatically increase its capacity by spawning additional processes on other processors and to maximally take advantage of a distributed heterogeneous pool of computing resources. Each distributed server node will be hereafter referred to as a “servelet” to connote a piece of code and associated data that can travel across the network to a remote machine and be automatically launched on that machine to add capacity to the computational network.

The auto-scaling system will be integrated with an online authoring server with integrated content authoring tools so that when a user subscribes to and “checks out” a visualization object applet from the system, the auto-scaling system will automatically employ a server-push system to propagate (push) a “servelet” and the associated volume data to a pre-determined number of additional machines, after which each servelet will be automatically launched and begin waiting to be invoked when needed to increase system capacity.

An automatic benchmarking system will be embedded into each MultiVIS servelet. This code will run continuously in the background. It will essential be a small user session running a scaled-down version of the Phase I test suite on a very small volume dataset, probably in the range of 16x16x16 voxels. The benchmarking system will track the execution time of the benchmarking test suite. This will be compared with a set of threshold values to determine whether the server performance is in the average, below average, or above average ranges. Once performance has degraded below a certain level, then additional distributed servelets will be invoked by the processor coordinator module until performance is improved to the point where it falls within a safe-range.

A zooming capability will be provided to allow the user to view and interact with the data at various levels of magnification, from a 1:10 resolution "volume icon" view, through various intermediate levels, to full 1:1 resolution, merely by right-clicking on an anatomical object and selecting the desired zoom ratio. Specialized zoom-processor software modules will be developed to allow fast integer-ratio zooming at each parallel processor node, with each node’s zoom processor working on only that node’s assigned subset of the entire body dataset.

The earlier MultiVIS system did not expose the volume rendering capabilities of Eolas' VIS technology to the user. The server development to be done in the next phase of the project will add this capability to the client applet. This will add a “Mode” switch that will determine whether the server operates in texture mapping mode or volume rendering mode. For example, if the “volume rendering” mode switch had been activated rather than the texture mapping switch in the user’s client applet, then the system would perform a ray casting operation for every voxel in the contained space at each rendering event. This involves computing the viewed intensity of each voxel by looking up the stored transparency/translucency parameter for each voxel, and the calculating a virtual “ray” that passes through the data to determine how visible the voxel would be when viewed “through” the surrounding voxels. Volume rendering allows the selective visualization of various tissue types by setting the surrounding voxels to be completely transparent. Volume rendering produces an image where the internal organs of an organism can be viewed, while still seeing the external surface, such as is shown by the embryo stereo pair in Figure 4. Since all of the voxels are projected to the viewing plane with varying degrees of translucency, it is impossible for the user to click upon an individual voxel, unless some sort of 3-D-stereo cursor control device is used. Future development may focus in this area.

The original VIS system employed a distributed parallel rendering engine to support the volume rendering computations. This parallelization was under user control, and could be invoked at any time by the user to speed up server response time merely by adding servers to the rendering group. This architecture allowed use of an heterogeneous collection of remote computers to act as servers for the system. The next MultiVIS system will likely take a similar approach.

Te MultiVIS project represents an ongoing effort to create a technology infrastructure to support advanced knowledgespaces for biomedical informatics. As the flood of new scientific information continues to grow, it is only though such integrated software approaches that scientists will be able to fully explot the rapidly expanding capabilities of new computational technologies to explore previously-unknown patterns in large information bases, to make new discoveries which elucidate our understanding of the fundamental mechanisms of biological growth and development.

Literature Cited:

1) Both the MetaMAP patent (4,847,604) and the Web platform patent (5,838,906) are viewable at http://164.195.100.11/netahtml/srchnum.htm

2) Ang, C.S., D.C. Martin and M.D. Doyle, *Integrated Control of Distributed Volume Visualization Through the World Wide Web,* Proc. Visualization '94, IEEE Press (1994)

3) Doyle, M.D., R. Raju, C. Ang, A. Goshtasby, and T. DeFanti (1992) *The Virtual Embryo: real-time viewer-centric 4-D visualization of human embryonic anatomy,* presented in "The Cave", Virtual Reality Theatre, at the High Performance Computing and Communication Showcase, SIGGRAPH '92, the annual meeting of the Association for Computing Machinery's Special Interest Group for Graphics, Chicago, August,1992

4) Doyle, M.D., R. Raju, C. Ang, G. Klein, A. Goshtasby, and T. DeFanti (1992) *The Visible Embryo: distributed (workstation/supercomputer) interactive visualization of high-density 3D image data of embryonic anatomy*, presented at the High Performance Computing and Communication Showcase, SIGGRAPH '92, the annual meeting of the Association for Computing Machinery's Special Interest Group for Graphics, Chicago, August, 1992

5) Doyle, M.D., C.S. Ang, and D.C. Martin, *Embedding interactive external program objects within open-distributed-hypermedia documents,* High Speed Networking and Multimedia ‘94, SPIE Press (1995)

6) Doyle, M.D., *The Visible Embryo Project: Embedded program objects for knowledge access, creation and management through the World Wide Web,* Computerized Medical Imaging and Graphics, In Press (1996)

7) Doyle, M.D, C.S. Ang and D.C. Martin, *Proposing a Standard Web API*, lead article in Dr. Dobb’s Journal, February, 1996

8) McConathy, D.A., and M.D. Doyle, *Interactive Displays in Medical Art,* "Spatial Displays and Spatial Instruments", NASA Conference Publication #10032 , (1989)

9) Williams, B.S., and M.D. Doyle, *An Internet Atlas of Mouse Development*, Computerized Medical Imaging and Graphics, In Press (1996)

10) The Visible Human Project (http://www.nlm.nih.gov/research/visible/visible_human.html);