

NSF/ARPA/NASA Digital Libraries RFP Response (draft)

Title: Embedded Visualization Objects for Knowledge Access, Creation and Management through the World Wide Web

Principal Investigator: Michael D. Doyle, Ph.D., UCSF Library and Center for Knowledge Management

Table of Contents:

Project Summary:

Executive Summary:

**Plaintiffs' Trial
Exhibit 690**

E 006776

Project Description:

Objectives of Proposed Work:

1) To develop a prototype knowledge management environment for the biomedical sciences which integrates access to online representations of the scientific literature, bibliographic databases, high-performance visualization technologies, large-scale scientific databases, and tools for authoring new-generation scientific publications;

1.a) To explore and evaluate the applicability of these tools in the areas of radiology and developmental & molecular biology;

1.b) To provide a means for relating digital forms of spatial, functional, and conceptual information as a basis for linking the biomedical scientific literature, through the Red Sage electronic journals project, to data resources provided through the Visible Human Project, The Human Brain Project, The Visible Embryo Project, The Human Genome Project, The Protein Database, and other large-scale biomolecular and biostructural databases;

1.b.1) To exploit these linking strategies in the creation of a set of integrated semi-automatic front ends to varied scientific databases accessible through the Internet;

1.b.2) To incorporate these linking methodologies into interactive authoring and editorial tools, allowing the creation of online publications that can embed visualizations and simulations which draw data from these Internet-accessible scientific databases;

1.c) To develop tools which provide access to interactive visualization and analysis of massive biomedical datasets through the Internet's World Wide Web distributed hypermedia network;

1.c.1) To refine and extend our existing algorithms enabling distributed visualization and analysis software "engines" which can be efficiently accessed by remote users through the Internet;

1.c.2) To refine and extend our existing algorithms to allow the display and real-time interactive control of three- and four-dimensional data visualization and analysis tools within hypermedia documents viewed using NCSA's MOSAIC graphical browser to the World Wide Web;

1.c.3) To develop algorithms which use novel compression technologies for the optimized interactive remote control of computationally-intensive graphical applications through the Internet;

1.c.4) To integrate a, b & c into a system which allows real-time remote access to distributed parallel computational applications for visualization and analysis resources within a distributed hypermedia environment;

1.d) To explore extensions of the paradigm of scientific publishing which are made possible through use of current multimedia technologies in a networked environment, including:

1.d.1) publishing multidimensional datasets integrated with articles, eg: MRI and molecular data, preferred views, animations, interactive visualizations, interactive mathematical models, and

1.d.2) development of scientific authoring tools for publications which exist only in the networked environment.

1.d.2.a) This will include integration of HTML+ WYSIWYG authorial and editorial tools, multidimensional data visualization applications, molecular modelling and database management tools into an interactive scientific publishing environment.

Background and Significance

Knowledge Management

Biomedical professionals today face an imposing problem. The amount of information which they must assimilate just to keep current in their fields has created a situation for them which has often been described as being like "trying to drink from a firehose." This results in an environment where improvements in an individual professional's information management tools can result in drastic improvements in their effectiveness in such areas as quality of health care, significance of research discoveries, or commercial potential for new inventions. This problem is especially significant for industrial researchers, who often have problems even in accessing the information they need, which is often found in scientific literature collections at major academic institutions. Computer-based information technology shows promise for solving these problems eventually, but limitations in computer capacity, cost and networking speed pose formidable barriers to reaching solutions to these problems. The presently-proposed project will attempt to exploit state-of-the-art computer, software, and networking technology to begin to tackle these problems.

In 1990, Richard Lucier, University Librarian at the University of California, San Francisco, charted out his view of the future of academic libraries in an information technology environment. The following is adapted from his paper published in *EDUCOM Review* (Nov.-Dec., 1992, vol 27, (6):24-31).

Since the 1980s, network access to knowledge through online catalogs and indices, along with a new emphasis on service and education, has assumed major importance at academic institutions across the nation. However, during that time, the respective roles and functions of scientists, publishers, and librarians have remained fundamentally the same.

Scientists routinely discover new knowledge and then communicate it through both writing and teaching. Publishers then disseminate that information through a variety of primary and secondary information products. In their traditional storage-and-retrieval role, research libraries use these resources to build large collections and thereby make available to users the world's published literature.

The limitations of this prevailing model for scientific communication were becoming apparent by the late 1980s. The length of the hard-copy publishing process makes it increasingly difficult for scholars and scientists to communicate their findings in a timely fashion. Faced with a long-term financial crisis evidenced in libraries by the rising cost of information, universities could no longer afford to maintain comprehensive collections. As a result, the presentation of knowledge in static form, whether in print or as part of the emerging electronic library, has begun to appear grossly inadequate.

Due in part to these factors, scientists, often on their own and with inadequate support, are beginning to augment the passive presentation of knowledge by adding interactive, discipline-based knowledge bases. These are routinely developed, maintained, and shared across computer networks. Problems resulting from the heterogeneity of data formats and user interfaces can often be handled only in an ad hoc manner, however, because there is usually a lack of widespread and organized support for information exchange activities. Knowledge management has emerged from this environment as a creative response to managing the world's knowledge base.

What is remarkably different about the knowledge management model is that it insinuates the library at the beginning of the information transfer cycle, rather than at the end, and focuses on information capture rather than access and use.

Encompassing the entire information life cycle, from creation of new knowledge to its dissemination and use, knowledge management represents a collaborative enterprise in which scientists and the library work together to develop and maintain knowledge bases and derivative information products. Knowledge management constitutes a new model for scientific communication in which faculty and a multidisciplinary team of computer scientists, software engineers, and research librarians share responsibility for the collection, structuring, representation, dissemination, and use of knowledge through the use of electronic information technologies. Knowledge bases are developed and maintained through knowledge management processes, which ensure content integrity and usefulness.

The collaborative nature of knowledge management, embodied in techniques of shared development of functional specifications, rapid prototyping, and user acceptance testing, fosters an interdependency among the collaborators. The ongoing management of the knowledge base also requires funding and administrative strategies that cut across traditional departmental, disciplinary, and institutional boundaries.

The long-term implications for building and maintaining a portion of a library's collection in this manner are enormous. Knowledge management transforms the various roles in the scientific communications process and potentially places ownership and control back into the hands of the scientific community. It also has tremendous potential for closing the gap between research faculty and students and for integrating the library into research and education programs in a significant way.

The knowledge management view that Mr. Lucier has proposed outlines many of the fundamental principals behind UCSF's Center for Knowledge Management.

...more on CKM...

The Red Sage Project

The Red Sage project was created as a first step towards achieving a functional knowledge management environment at UCSF. This project arose from the belief that university libraries must thoroughly investigate the applications and consequences of state-of-the-art information technology to their mission of serving the information needs of their campuses. The university library must become less of a storage facility and more of a center for information management, for learning, and for consultation.

It's important to note that this kind of work cannot be successful in a vacuum. Many factors in the areas of both publishing and information technologies are of critical importance. So about one and one half years ago The UCSF University Librarian, Richard Lucier, met with a major scientific publisher, Springer-Verlag, and a major player in the information technology industry, AT&T Bell Labs, to decide that it would be feasible to begin a project looking at the problem of designing, implementing and evaluating a fully digital academic library. The system would use AT&T's innovative RightPages™ electronic journal software as the archiving and delivery platform. This agreement was struck over dinner in the Red Sage restaurant in Washington D.C., and so the Red Sage project was born. Over the last year, many additional publishers have joined the team. These include John Wiley & Sons, the American Medical Association, the Massachusetts Medical Society, publishers of the New England Journal of Medicine, and the Nature Publishing Company, publishers of Nature, as well as many others.

The primary goal of the Red Sage Project is to explore the technical, legal, economic, business, and social issues surrounding the delivery of scientific, technical, and medical information in a

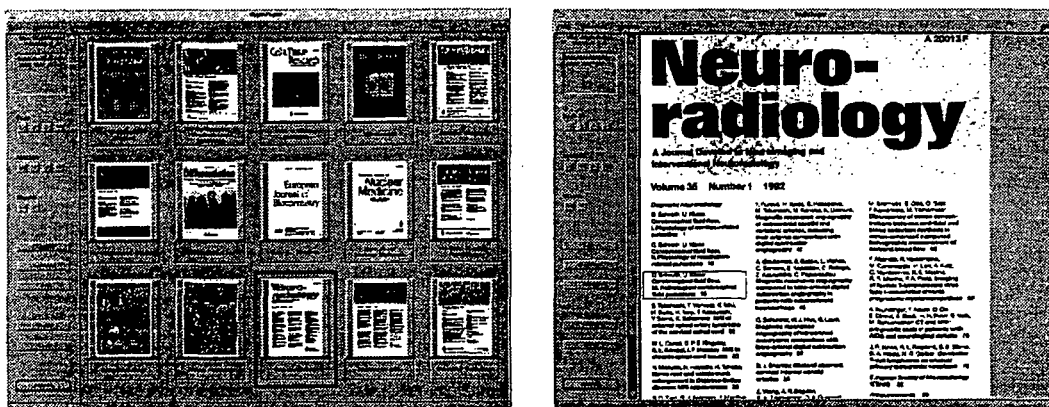


Figure 1: AT&T's Right Pages client-server system presents users with a traditional library metaphor.

networked environment. An additional objective is to establish a critical mass of content that will make this system a "must-use" resource for the faculty, staff and students on the UCSF campus. Therefore it was decided to limit the initial database to a few key content areas, so that we could achieve good coverage of content critical to a large number of research and clinical faculty on campus. These areas included both radiology and molecular biology, as well as an area called "general high impact", which included a few publications which appeal to a general audience, but which have a very high readership on campus, such as the New England Journal of Medicine. The project will have over 40 titles available online when the system is first opened to the public in the first week of January, 1994.

There are some main features of this system which are of note: First of all the project team decided that, in order to get the system up as soon as possible, they would try to use an existing software package that was stable enough to be used as a "production system" in the Library. It needed to allow the user to follow a traditional library metaphor (Fig. 1), organizing the information in the form of stacks, journals, years, issues, and articles, just as these things are organized in the Library. It required a graphical user interface that allowed onscreen display of journal pages, providing the same "look and feel" as their paper equivalents. There had to be a useable search engine, and it was desirable for that engine to be able to record an individual's preferences and to alert the user to pertinent articles as soon as they were loaded onto the system. To provide this functionality, and to insure reliability and system integrity across a wide variety of client platforms, a client-server architecture was preferable. To allow facile analysis of human factors issues, the system needed to allow retroactive analysis of the users' practices and patterns of use. And, finally, the team was aware that, at least initially, most users will use the system for finding and screening the articles they need, but will want to print out the papers before studying them. Therefore, high quality printer output to a local machine was required. The X Window-based RightPages electronic journal system from AT&T met all of these needs well.

Image quality was a primary consideration in this project. The Red Sage team realized that, in order for the users to accept the system, they must not be asked to sacrifice legibility of either text or images in the online versions of the articles. The onscreen display first shows the article so that the entire page fits on the screen. Depending upon the size of the screen, this may be adequate for easy viewing or, alternatively, the image can be zoomed to allow easier reading, as well as detailed examination of tables and figures. Photographs are shown on the pages in a binary, or black and white, mode, but if the user wants to see a higher quality view, a button on the control panel will bring up a grayscale, or eventually color, display of that photo (Fig. 2).

In addition to the bitmapped images of the pages, the system provides an underlying full text ASCII database that can be searched with conventional approaches. This searching capability can be automated so that the system stores user profiles of search terms, and new content is

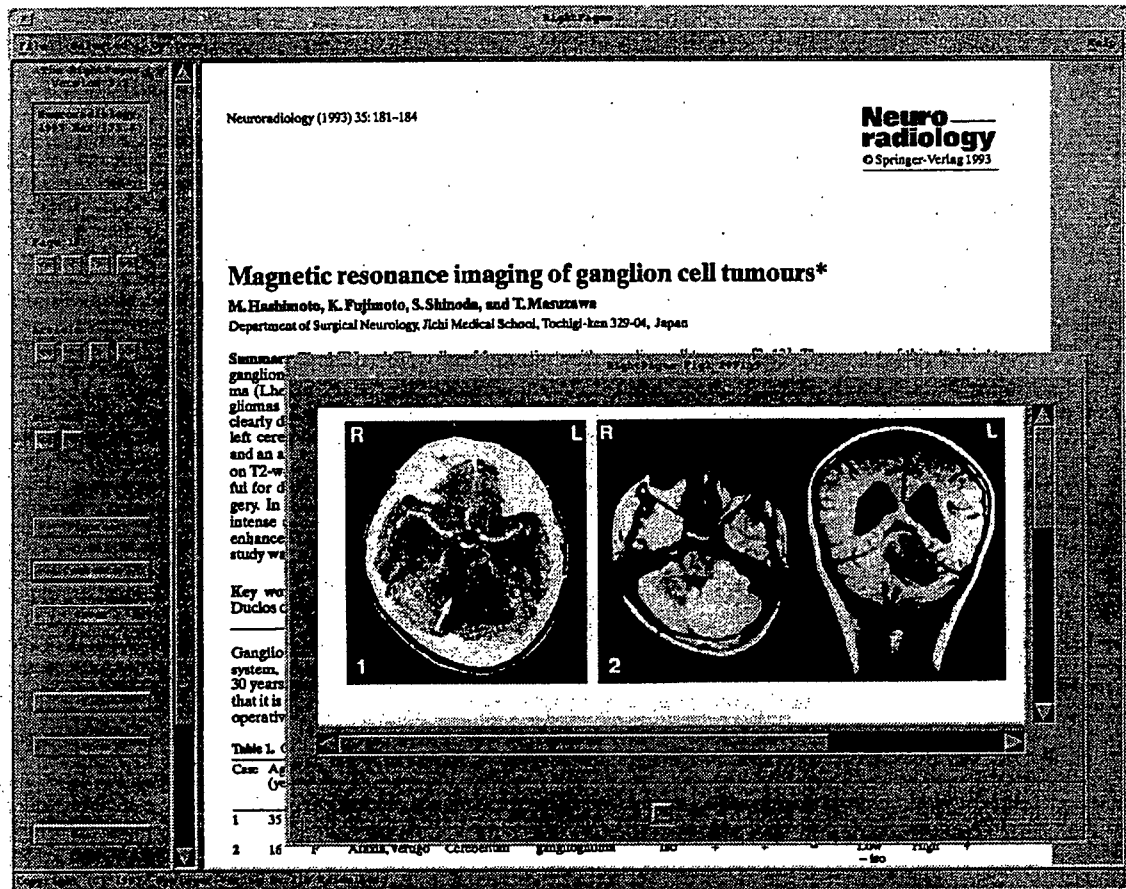


Figure 2: The Red Sage system provides high-quality grayscale images for viewing at the click of a button.

searched automatically as it is being loaded onto the system. If newly-loaded content matches those parameters, the next time the user logs on he or she will see the relevant issues highlighted on the stacks display.

This alerting feature represents perhaps the most important aspect of online literature systems in comparison to conventional printed journals. With the deluge of scientific and technical information that is printed every day, it is both physically and economically impossible for researchers to keep up to date without the aid of computer-based indexing and retrieval systems. MEDLINE, from the National Library of Medicine, provides this capability for much of the biomedical research community. The articles covered by MEDLINE, however, routinely appear 2-3 months after the date of publication. The articles included in the Red Sage database routinely appear from 1 to 8 weeks *before* their printed equivalents are available. This is very important to both academic and industrial researchers, where a piece of key information in a published article can drastically influence the course of an important experiment or clinical care protocol.

The World Wide Web

The delivery of information via electronic means has grown over the past several years to include text, images, audio and video [REF]. A significant obstacle to overcome in delivering this information in a ubiquitous manner has been a combination of format and accessibility. Although significant steps have been taken to alleviate these problems, it was not until the acceptance of the World-Wide-Web (W^3) and the availability of a portable and redistributable client, in the form of Mosaic from the National Center for Supercomputer Applications (NCSA) at the University of Illinois at Urbana-Champaign (UIUC), that a method has become available to distribute a wide variety of multi-media information in an inter-networked environment.

1.1 The World-Wide-Web

The World-Wide-Web (W^3) provides a means for accomplishing information access, retrieval and structure support over the Internet through the definition of a network communications protocol for transmitting hyper-text documents (**Hyper-Text Transfer Protocol - HTTP**) and a SGML document type definition (**Hyper-Text Mark-up Language - HTML**). These two pieces of technology, combined with a client capable of presenting multi-media data, and a server maintaining a repository of information, provide the basic mechanisms for delivering multi-media information in a client/server fashion over the Internet. In addition to the protocol and DTD, the W^3 also provides connectivity to other inter-network services, such as Telnet, FTP, Archie, WAIS, electronic mail (SMTP/MIME) and network news (USENET). This additional connectivity allows a WWW client to be used as an *inter-network information integration agent* (INIIA).

MOSAIC-based Red Sage Browser

We have utilized content from the RedSage Project [REF], technologies from NCSA/UIUC, the Center for Network Information Discovery and Retrieval (CNIDR), the European Center for Particle Physics (CERN), the University of California at Berkeley (UCB), and other individuals and institutions to provide an on-line digital library of medical journals in the areas of radiology and molecular biology. This repository is not only accessible via the Internet, but may be utilized as both a retrospective and prospective information source, assisting users in the identification and dissemination of relevant articles. Our project is an experimental evolution of the RightPages™ software, utilizing the same content base of journal titles, but extending the scope of information discovery and retrieval to include not only the journal literature, but also the bibliographic information available from on-line indices and catalogs as well as the breadth of services offered by other Internet providers (e.g. GDB, GenBank, OMIM).

...DIAGRAM...

File Options Navigate Annotate Help

Document Title: [Redacted]

Document URL: [Redacted]

[View Article](#)
[Previous Article](#)
[Next Article](#)
[Last Article](#)
[View Page](#)
[Previous Page](#)
[Next Page](#)
[Last Page](#)
[Index](#)
[Figures](#)

Neuroradiology (1993) 35: 181-184

**Neuro
radiology**
© Springer-Verlag 1993

Magnetic resonance imaging of ganglion cell tumours*

M. Hashimoto, K. Fujimoto, S. Shimoda, and T. Masuzawa
Department of Surgical Neurology, Jichi Medical School, Tochigi-ken 329-04, Japan

Summary. The MRI and CT studies of four patients with ganglion cell tumours, one with a cerebellar gangliocytoma (Lhermitte-Duclos disease), and three with gangliogliomas are reported. MRI in Lhermitte-Duclos disease clearly demonstrated a mass of low signal intensity in the left cerebellum on T1-weighted spin-echo (SE) images and an area of high signal intensity with a blurred margin on T2-weighted SE images. These MRI studies were useful for delineating the lesion, which was verified at surgery. In the ganglioglioma, MRI demonstrated two iso-

intense solid masses on T1-weighted SE images which

glion cell tumours [3-13]. The purpose of this study is to review the CT and MRI characteristics of these tumours.

Materials and methods

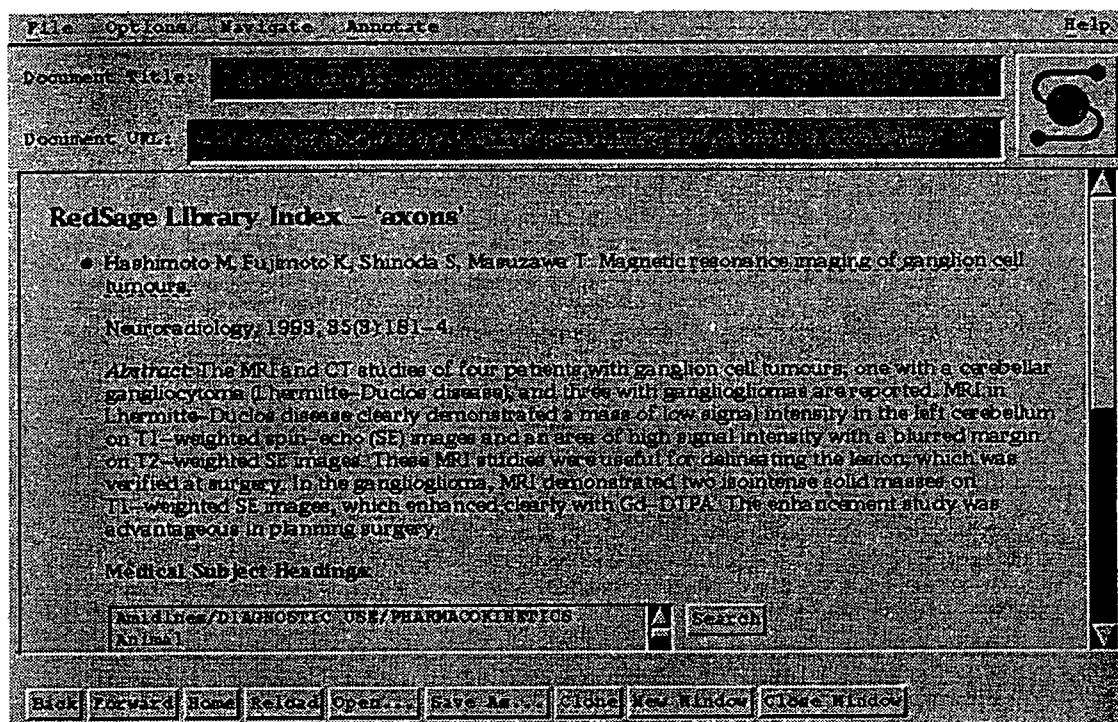
Four patients with ganglion cell tumours were examined by MRI and CT (Table 1). One had a cerebellar gangliocytoma (Lhermitte-Duclos disease), and the other three had gangliogliomas. All diagnoses were histologically verified.

CT examinations were performed with and without contrast

Back Forward Home Reload Open Save As... Clone New Window Close Window

Medline

...DIAGRAM...



Integrated Vis

Advance scanning devices such as Magnetic Resonance Imaging (MRI), Computer Tomography (CT), and Satellite Imaging have been widely used in areas like medicine, defect detection on the production lines, and meteorology. The need to visualize the sampled data quickly spurred the development of various volume visualization techniques, which in turn, provided the 3D imaging related research groups a revolutionary tool that allowed noninvasive examination of the object of interest. As a result, visualization systems such as AVS, ApE, Sunvision Voxel, and 3D Viewnix have been subsequently introduced to the user communities. These high-end commercial software systems, however, are not cost-effective solutions for the majority of the users who are interested in visualization of the published data and results. Accessibility of the tremendous datasets, and availability of a suitable graphics workstation further hinder most people's possibility of experiencing the authors' findings.

Our project attempted to integrate volume visualization with W^3 , and thus take advantage of the client-server architecture of W^3 to allow centralized data storage and on-demand dissemination. We enhanced the capability of the W^3 server, PLEXUS, from Knowledge Systems and the Computer Science Department of Carnegie Mellon University, and the W^3 client, Mosaic, from the National Center for Supercomputer Applications (NCSA) to support the 3D data object. In addition, we united our distributed volume visualization system, VIS, with Mosaic for 3D volume manipulation and image computation. The distributed nature of VIS allows the users to utilize remote computation servers, and therefore, enables visualization on low-cost workstations.

The data storage sites run W^3 servers which communicate with W^3 clients such as Mosaic via Hyper-Text Transfer Protocol (HTTP), the same way the banal FTP servers serve a file transfer session. Upon receiving a service request, the W^3 server will establish a connection with the client, and transmit the requested digital objects (documents, sound clips, images etc.) with their respective tags in the headers. The client will read the headers, and based on the types of data, decide the appropriate actions. For example, a W^3 server will reply to an image transfer request from a client by sending a header

containing /img? as the data type followed by the image data. The client will peek into the header and invoke its image handling function(s).

1.2 About the W³ client Mosaic

Mosaic is one of the major W³ clients. It has versions for different platforms, which include X-Window Systems, Microsoft Windows 3.1, and Macintosh. Because of its fast development and wide availability, it is one of the most prevalent web browsers. The current version (2.2) supports documents written in Hyper-Text Mark-up Language (HTML). It enables not only page formatting, type-face formatting, and image displaying, but also specification of hypertext links and creation of a variety of Graphical User Interface (GUI) widgets.

1.3 About the volume visualization tool VIS

VIS is a simple but complete volume visualizer that reads the Hierarchical Data Format (HDF) volume data files, and performs fast texture mapping of the volume data onto the geometry formed by six axial clipping planes (cube), and one arbitrary clipping plane. It also has a distributed volume rendering routine that supports run-time GUI based selection and deselection of computation servers, and an isosurfacing function with software Gouraud shading (thus special graphics hardware is not required) for surface extraction and rendering.

1.4 Goal

Through the integration, the W³ client (Mosaic) supports 3D image data visualization by spawning VIS when the data type description passed from the W³ server is a 3D volume image.

The rest of the paper is divided into five parts. Section 2 will describe VIS and how it fulfills the role as a W³ visualization tool. Section 3 is devoted to Mosaic, and will cover the implementation of its interface with VIS. The last section is about our conclusion and possible future development.

2. VIS as a W³ visualization tool

VIS was first developed to be a portable visualization tool among X-Window system workstations. It is designed to carry out most of its functions as fast as possible without using any graphics hardware. We traded the possibility of real time rendering with the ability to run on most workstations: SGI's, DEC Alpha's, IBM RS 6000's, Sun SPARCStations etc. In order to provide the best interactive performance, we modified VIS in two different directions: (1) utilization of graphics hardware, and (2) distributed rendering on powerful workstations. The results of rendering are sent into Mosaic with a set of predefined network protocols.

2.1 Real time VIS

With the help of an SGI Onyx with RealityEngine² (RE²) graphics, and the OpenGL implementation for the RE², we are able to implement volumetric rendering in real time. The rendering routine defines the input volume dataset as a 3D texture map. This setup allows not only interactive transformations (rotations, translations, and scaling), but also provides the GUI for interactively defining the axial and arbitrary clipping planes, and specifying the intensity-to-density and density-to-opacity functions.

2.2 Distributed VIS

The VIS implementation has great potentials to run in parallel. We distributed the loads of volume rendering among workstations with a greedy algorithm that chooses to give bigger portions of the picture to faster machines. The client that VIS is running on will break the load into a relatively small portions, which are still big enough such that the overhead for network transmission of the data is substantially smaller than the time to transmit the actual data. Each of the computation servers will then fetch a portion for rendering via Remote Procedure Calls (RPC). A fast server will return the results earlier, and fetch another portion. The servers will compete with each other for data, and this ensures that most servers are busy most of the time. A report of analysis of this algorithm from Giertsen and Petersen (Giertsen, 93) shows that the performance improvement is a function of the number of load portions, and the number of computation servers. The test results suggest that the performance improvement max out when the number of sections of 512x512 and 1024x1024 pictures are around 10-20 on several volume datasets of various sizes on up to four servers. They also show that a larger number of workstations servers will subsequently enlarge the optimal number of sections. This approach may not be the best distributing algorithm, but it achieves the close-to-optimal results most of the time.

2.3 Interface with Mosaic

The VIS/Mosaic software system consists of 3 elements: VIS, Mosaic, and Panel (which provides GUI for data manipulation). Currently, both of the above VIS implementation communicate with Mosaic through the standard X interclient communication protocol (ICP), but the same algorithm will theoretically work in various windowing systems that support ICP. Upon receiving the notification to load a 3D image dataset, Mosaic creates a DrawingArea widget in its document page, and a pixmap as a storage of rendering results. It then starts a child process, which in turn, launches Panel. The fact that both Mosaic and Panel are running on the client machine, and thus sharing the same windowing system server, provides a convenient communication bridge between the two. The pixmap created by Mosaic will serve as a chunk of shared memory (for images) accessible both by Mosaic and Panel. In addition, Panel and Mosaic may send messages through the windowing server to coordinate the image displaying.

Beside coordinating the displaying of images, Panel functions as a load distributor. It delivers the rendering parameters to the selected VIS server(s), and fetches the resulting image portion(s). Thus the scenario following a user's action on the Panel will be (1) Panel requests VIS server(s) to render the image with the parameters from its GUI, (2) Panel fetches the returned image data, then writes it to the pixmap, (3) Panel sends a message to notify Mosaic upon completion, and (4) Mosaic bit-blots the image

in the pixmap into its DrawingArea widget. This will be addressed in more details under section 3.2. The configuration of this software system is depicted in figure 1 below.

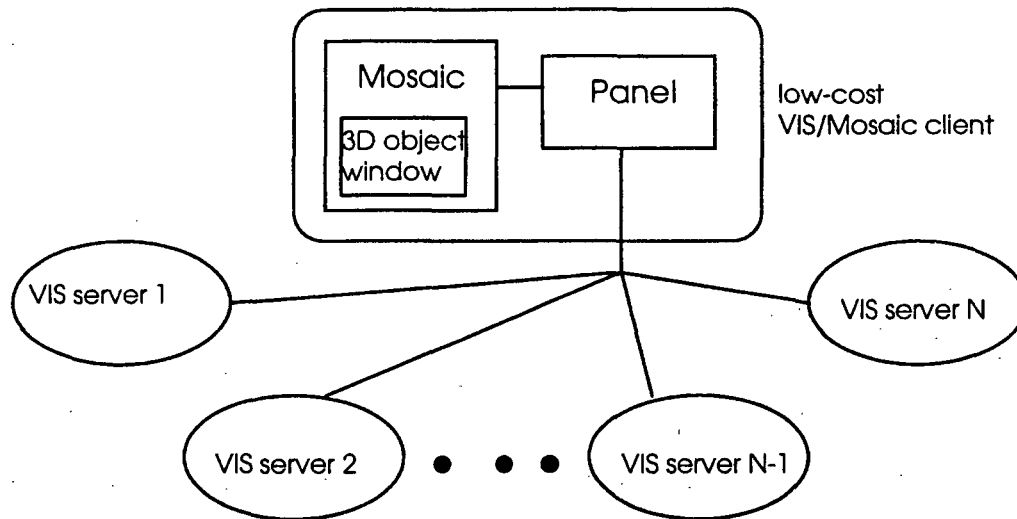


Figure 1: VIS/Mosaic software system.

3. Mosaic and the W³ server

Supporting of the 3D volume data object requires both Mosaic and the W³ server (PLEXUS, in our case) to recognize this data type. We also defined the roles of each of VIS, Mosaic, and PLEXUS.

3.1 Startup coordination

Since the administrator of a W³ site will declare to its W³ server the type of each W³ file, we added the 3D volume data object to the PLEXUS server, and declared the HDF volume datasets to be such type. The W³ client (Mosaic) is modified to handle this type of data when it sees the tag in the HTTP header. As mentioned in the last section, Mosaic creates a DrawingArea widget and a pixmap, and forks a process to start Panel, which in turn, starts the user specified VIS server(s). Once started, VIS server(s) download(s) the HDF data from the same PLEXUS server with the Universal Resource Locator (URL) from Mosaic. The modifications involve addition of a new tag into the PLEXUS code, enhancement of the HTML library on the Mosaic side, and implementation of URL loading function for VIS.

3.2 Interclient communication

We recognized the minimum set of communication protocols between Mosaic and Panel:

- (a) Mosaic notifies Panel in events when the window id is no longer valid. This happens when the user navigates from the current page to the previous (with BACK button), or to another document through a link.
- (b) Panel notifies Mosaic when the scene needs to be updated. This occurs when the user manipulates the data through the Panel's GUI, which may be any actions (rotations, scaling, changes in clipping planes, mode switching, for instance, from texture mapping to volume rendering) that result in rerendering of the picture.

The first case of (a) does not impose any problem since Mosaic can simply terminate VIS. The second case is harder to handle because Mosaic caches the page, and in principle, we expect VIS to remain running because it should not be loading the HDF data, which may be quite a few megabytes, again. In the previous case, we simply have Mosaic kill the child process that launched Panel, which may subsequently send a terminate message to the VIS servers. In the latter case, Mosaic will send a message to Panel requesting it to unrealize/unmap its window. Then when the user comes back to the 3D object page, Mosaic may send another message to Panel ordering it to realize the GUI window, and update the window id of the DrawingArea that Panel has been sending message to (ironically Mosaic does not cache the widgets/windows on the cached page).

In (b), Panel will send an update message directly to the DrawingArea window id provided by Mosaic. Upon receiving the redrawing message, the DrawingArea widget calls its registered refreshing function, which simply bit-blots the pixmap Panel updated into the window.

4. Conclusion and future work

Our accomplishment takes the technology of networked multimedia system (especially the W³) a step further by proving the possibility of adding new data types to the W³ servers and clients, and coordinating the execution of the applications that handle them with the W³ clients. The addition of the 3D volume data object in the form of HDF to the W³ is welcomed by many medical researchers, for it is now possible for them to view volume datasets without a high-cost workstation, and accessing datasets in the W³ fashion, through hypertext and hypergraphics links within an HTML page. As for the researchers who would like to share their findings with the world, they merely have to run a W³ server that serves the HDF files, which in turn, can be easily created with the various import facilities available freely from NCSA.

A very useful future development will be to define a clearer set of communication protocols, probably built on top of a popular network communication tool (ToolTalk) licensed by many major workstation/PC operating system vendors, that can be easily utilized with minimum modification in external application to communicate with Mosaic. Another possible solution is to use the Data Transfer Mechanism supported by Mosaic to handle the communication.

MetaMAP

Volumetric data--huge datasets--isn't practical to interact with entire dataset--multi-tiered system needed--Navigation is a problem--point-and-click selection of anatomical regions desirable

To effectively exploit the advantages of new mechanisms of information and retrieval, new approaches must be made towards incorporating existing programs as well as developing entirely new applications. There exists a need to integrate more sophisticated graphics into applications and to take a wider view of how that integration can take form. Nowhere is this more evident than in the design of the human-computer interface. According to James Foley in a 1987 Scientific American article entitled "Interfaces for Advanced Computing", "The interface between the user and the computer may be the last frontier in computer design (Foley, 1987).

Palette segmentation indexing (PSI) provides a new method for the creation of interactive computer applications which allows the user to interact directly with digital video images. The types of images that are most appropriate for this process are those which are made up of a number of discrete image elements, or objects, each of which has its own identity. An image of the anatomy of the arm would be a good example. The MetaMap process would allow the creation of a program where a user could interact with such an image merely by pointing to different areas of the arm picture with a mouse and clicking a button. The system could then identify the anatomical structure being pointed to, and automatically find and

display text or additional images which explain the important concepts concerning that anatomical structure.

MetaMapping deals with images as object-oriented entities. Single images are typically made up of multiple discrete elements, each of which has an identity in its own right. This is accomplished through a sophisticated technique, called palette segmentation indexing, for reprogramming of both the image and the color map so that the discrete image elements comprise unique sets of color indices. This results in a highly simplified mechanism for the identification of any particular image element or "object". It also produces a very efficient form of object-identity data storage in a spatial-resolution-independent system.

A MetaMap-based interface is most appropriate for hypermedia-style applications. "Hypermedia" describes a particular type of computer interface and database structure where any particular data item (or object) can be related, or linked, to any other item. This concept was first proposed by Vannevar Bush (Bush, 1945), and later popularized by Ted Nelson (Nelson, 1988). The object orientation of MetaMapped images, the space efficiency of the indexing data, and the simplicity of the identification process combine to extend the concept of hypertext applications to include hyperimages as well.

The MetaMap process is device independent. This means that programs which use it can be made to operate on many different kinds of computers. This process is protected by a United States patent (Doyle, #4,847,604), however a royalty free license will be granted to this project for implementation of the MetaMAP process in the browsing system.

A MetaMap-based interface falls within the category of "direct manipulation" interfaces. According to Ben Schneiderman (Schneiderman, 1982, 1983), direct manipulation interfaces can be identified by the three following "essential" characteristics: 1) Continuous representation of the objects of interest, 2) The interactive activity comprises physical actions instead of complex syntax (especially those actions which mimic the desired real world counterpart's actions or changes), and 3) Rapid incremental reversible operations whose impact on the object of interest is immediately visible, so that the objects upon which the actions are taken are exactly the same as those upon which the evaluation is made. Hutchins, et al. (1987), state that "a good Direct Manipulation interface is invisible - The user feels as if operations are actually done directly in the task domain. And if the computer novice is already knowledgeable in the task domain, then much of what is needed to use the interface is already known." Hutchins, et al., go on to state that a problem with direct manipulation interfaces is that they have difficulty distinguishing the depiction of an individual element from a representation of a set or class of elements. Such problems do not arise with palette segmentation indexing (MetaMapping). The improvements inherent in the MetaMap process enable it to fully exploit the potential of direct manipulation without encountering the problems of earlier systems.

The Principal Investigator (Doyle) originally developed this process with the aim of improving the interactivity of computer-aided medical instructional programs on very low-end microcomputer platforms. Previous such systems that used digital imagery dealt with the image as a single entity. These systems were predominantly text-based and did not allow the user direct interaction with the various elements which make up an individual image (Thursh, et al., 1985, 1986; Glickman, 1987). This was partly due to the fact that the computer hardware necessary to display high-quality color digital images was far too expensive to be considered for inclusion in an educational workstation. As a result, many existing computer-aided instructional systems have entirely text-based interfaces or depend upon the use of graphic symbols (icons) to represent real-world objects or concepts (Nelson, 1988). Using the MetaMap process, realistic images are employed to communicate instructions and concepts to the user. This is accomplished by exploiting several capabilities of today's high-quality graphic computer display hardware.

Human color perception outstrips the abilities of even the most powerful supercomputers. Due to this fact, developers of interactive graphic systems are faced with numerous compromises which must be made. If monochromatic (gray-scale) images are used, this problem can be significantly reduced. Whereas humans are capable of distinguishing approximately 7.5 million colors over a wide dynamic

range of intensities and saturations, variations in grayscale are only perceptible to a few hundred gradations (Murch, et al., 1988).

If an image is rendered in too few colors or grayscales, a phenomenon called contouring occurs. This involves the appearance of visible bands across areas that should be smoothly shaded, looking somewhat similar to elevation bands on a topographical map. Continuous tone images rendered in 2 to 8 grayscales show obvious contouring. If 16 to 32 gray levels are used, a drastic improvement is visible (figure 1). This usually produces only subtle contouring for most images. High frequency images, i.e., those that are highly textured or have many abrupt changes from dark to light, can typically be rendered well with fewer gray levels than softer, smooth shaded, low-frequency images.

This requirement can usually be reduced if an image is divided up into its constituent objects. Most objects have a characteristic tonal range which can often be represented with a limited number of gray values. As will be shown below, MetaMapping allows each class of image objects to be treated separately when determining the optimal number of grayscales or colors necessary to faithfully render the image.

Today's microcomputer systems are commonly capable of displaying 256 colors onscreen at a time. Each of these 256 colors can typically be any of 256 thousand or more possible combinations of hue, saturation and intensity. Many of the newer workstation-level computers can display between 4096 and 16 million colors onscreen simultaneously. The majority of images contain no more than 30 thousand distinguishable colors (Campbell, et al., 1986). The problem of contouring is more complicated with color imagery. Whereas with grayscale images, intensity changes alone contributed to perceptible banding, color rendition includes hue and saturation components to further complicate the issue.

One of the primary motivations in limiting the number of onscreen colors to as few as permissible, is that, with more colors displayed simultaneously, the computer must manage and store a significantly larger amount of data. One unit of computer memory (1 byte) can store the information for eight pixels in a 2 grayscale image, or one pixel in a 256 grayscale image. If a 16 million color image is to be represented, then it will require three units of memory (3 bytes) just to store one pixel's color information.

To combat this problem, computer hardware designers introduced the concept of a color look-up table (CLUT) to allow many different color combinations to be displayed while requiring usually only one byte per pixel to store the displayed image file.

The byte for each pixel location stores an index from 0-255 which is then linked to a table that contains 3 numbers, each of which describe the relative proportions of red, green and blue to be displayed onscreen at the pixel location. This table is usually referred to as the "color look-up table" (CLUT), "palette", or "color map." The collection of indices which together describe the entire image is called the "bit map." The palette values can be reset for each image to be displayed and a file containing these values is usually stored on disk together with the image file. When the computer reads in an image for display, it first reads the image file to find the index value for each pixel location on the screen, and then it reads the palette file to find the R, G, B numbers that dictate the color for that index. Some file formats, such as the popular "PCX" type, combine the image pixel data together with the palette data into a single composite file.

If the specific colors to be used to render an image are carefully chosen, this can result in the capability to display and store pleasing, high quality images while keeping the data file sizes manageable. Several methods have been published for the optimal selection and mapping of limited numbers of colors to render realistic looking images. One of the best, and the most commonly used today is the median cut algorithm (Heckbert, 1982; Campbell, 1986). This process uses a method called adaptive color quantization that uses color histogram analysis and specialized sorting techniques to convert an image from 16 million colors to a specified subset of values, i.e., 256 or 4096 colors.

Heckbert's method uses the standard R, G, B model (Murch, et al. 1988) for representing the original image space, however he states that better results could be obtained by using a color model that is more closely based on actual human color perception. Many companies, notably Kodak, are actively doing research to develop such models, but they usually keep most of their results secret so they can implement them in proprietary technology. One of the most promising perceptual-based models is the well-known Munsell color system. This system represents colors in an hierarchical "stem and leaf" organization where a range of similar colors appears on a single leaf. Ongoing research is underway to develop algorithms to allow computer systems to reference color values in such a way (Silverstein, 1990).

Developers of interactive computer educational training systems have just recently begun to exploit the ability to incorporate high quality graphics into their presentations. Often times it is desirable to allow the user to interact with the system by manual selection of a feature of an image on the video display. Developers using earlier computer systems with low resolution screens, and only a small number of color possibilities per pixel, often used the computer's bit map as a basis for referencing object identity on the computer display.

As was stated earlier, the computer represents each image as a set of indices corresponding to each pixel location. These pixels are arranged on a Cartesian coordinate system, called the bit map, where X and Y value relate to a specific location on the screen. To identify objects on the screen using bit-map methods, one must store a set of x,y coordinates in the computer's memory for each object to be identified. One bit-map-based method constructs a secondary bit map in memory that stores for each x, y screen location an index, or pointer, that links that screen location to additional data concerning the identity of the image object at that location. To illustrate this point, reconsider the earlier-described example of arm anatomy (figure 2). This user would move the mouse to point at a muscle that he or she couldn't identify. If the screen location where the mouse clicked was, for example, pixel location x,y = 118, 390, the computer system would then examine the secondary bit map at x, y = 118, 390 and find that the number at that location pointed to data indicating that the object in question was the brachialis muscle. The system would then respond with text, sound or other images to explain important facts about the brachialis muscle. As we will see below, there are a variety of problems associated with this approach.

An alternative approach would be to store just the x, y locations that outlined each image object. Then to identify the object, the system would have to search through all of the outlines relating to that image and determine for each outline if the selected screen coordinate lied inside, outside or on the outline. Once the relevant outline was located, the system would then look for a pointer from that outline data to other data identifying the anatomical structure in question.

...Mosaic's implementation of Interactive hotspots -- ISMAP...

The two methods of object identification described above have a variety of advantages and disadvantages associated with them. The first method, that of storing an object-identity pointer for each screen location, can be very fast if the resolution of the image is very low. It also provides good accuracy in that every pixel on the screen has a defined identity. As the resolution of the image increases, however, the speed of object identification decreases geometrically. Also the amount of extra data that the computer is required to manage for each image, the data overhead, is equivalent to the size of the original image file (100% overhead). To combat these problems, many developers instead use an invisible grid which is overlaid on the image. The system would then look for the closest grid intersection to the selected screen location. Each grid intersection would then contain a pointer to object identity information. This has resulted in drastic improvements in speed and reduced data overhead, but the accuracy of the earlier method was lost. Irregularly-shaped objects would often be erroneously associated by the identification algorithm with the wrong grid intersection and consequently with the wrong object identity information.

Because of this fact, the outline-search method is the most commonly found method today (Supercard, Opus, Toolbox). This technique had the advantage of good accuracy even for irregular objects, and both

fast identification times and low data overhead if the number of objects and complexity of object contours is low. If the number of objects is large, however, and/or the average complexity of object contours is high, then the advantages of this technique are lost. Also, oftentimes image objects can be grouped into classes on objects which share common identity information. A good example would be a microscopic image of a section through a lymph node. There may be several hundred cells visible on the screen, but they may be groupable into, for example, lymphocytes, macrophages, plasma cells and reticular cells. If one were to use outline methods and wanted to identify every cell in the image, then several hundred outlines would have to be stored, and both identification speed and data overhead would suffer considerably.

The current trend in display technology is towards megapixel (1000 x 1000) and higher display resolutions. Any interactive graphics system that is to have a long-term value must be easily converted to operate on new generations of hardware as it becomes available. Systems based on bit-map methods for object identification will have to be entirely rewritten to adapt to the required higher-density indexing schemes. Developers will also have to compensate for the geometric decreases in identification speed and increases in data overhead that will necessarily follow. This will, to some extent, be aided by normal increases in processor speed and memory capacity, but the performance of such systems will always suffer in comparison to color-map-based techniques.

The MetaMap process of palette segmentation indexing (PSI) was developed in response to the disadvantages of bit-map methods. Rather than establishing an object-identity pointer for specific screen locations based on the bit-map, MetaMapping involves using the value of the color index as the basis of object identification. In a 256 color system, for example, one class of objects could be rendered using color index values from 0-31, another class of objects using values 32-63 and so on. Suitable selection of R, G, B values for the CLUT would then render the objects within each class correctly. If one were again looking at an image of arm anatomy (figure 4), for example, and had clicked the mouse at pixel location $x, y = 118, 390$, then the system would look to see what color index was stored at that location. Let's say that the color index so selected was 55. The system would determine that this value falls between 32 and 63. The structure in question would therefore be classified as falling within the second range of index value. The system would then look to see that object #2 is the brachialis muscle and respond accordingly with text, sound or other images that explained the anatomical significance of the brachialis muscle.

MetaMapping images involves processing the image data so that each class of objects is rendered exclusively with index values that comprise a discrete subset of the available indices in the CLUT. This is the most labor intensive aspect of the process and is usually accomplished during the development of the interactive application. Although the object class subsets of indices are most easily grouped into contiguous ranges of entries in the CLUT (i.e., 0-31, 32-63, etc.), subsets may be defined by any consistent scheme, such as all even vs odd indices, common multiples, etc.

The advantages of such an object identification scheme are several:

- 1) Data overhead is minimized. The data storage load for a single image and object identification pointer table is typically less than 1% larger than the original image file.
- 2) Identification speed is maximized. Rather than searching through hundreds of thousands of pointers at each identification event, as some bit-map methods require, only 256 values must be searched for an image with 256 colors.
- 3) Applications are resolution independent. Performance will not degrade as display resolutions increase. MetaMap based applications can be easily converted to operate on higher-resolution systems.
- 4) Object oriented images are produced. The MetaMapping process encodes the object identification mechanism into the images themselves, so they can be easily incorporated into newly-emerging object oriented program development systems.

5) The number of objects within each class of objects is limited only by the spatial resolution of the display. The system response time and data overhead are not affected at all by large numbers of objects within object classes.

The one disadvantage of palette segmentation indexing is that, as the number of object classes within an image increases, the quality of the rendered image will decrease. This is compensated for partly by the fact that objects within a given class will tend to have characteristic colors and therefore can be effectively optimized using methods such as Heckbert's Median Cut algorithm (Heckbert, 1982). Color dithering can also be used to improve object rendering in images with many object classes. Although dithering produces somewhat grainy appearing images at today's average display resolution, this effect will be progressively less noticeable as display resolution increases in the future. Another factor that will minimize the problem of multiple object class rendering is the current trend towards 12-bit, color mapped display systems capable of displaying 4096 simultaneous colors from 16 million possibilities. These display capabilities can currently be found in graphics workstations such as the Personal Iris, from Silicon Graphics, Inc., and the RS/6000 from IBM Corp. Such display systems allow many more colors for each palette segment.

A greatly simplified protocol for implementing the MetaMap process with grayscale images would be as follows. First the number of object classes in the image is determined. This number is then divided into the number of colors in the CLUT to calculate the average size of a palette segment. The dynamic range of grayscales in the image is then compressed to the palette segment size using linear interpolation. Image objects within each class are then manually outlined and assigned to a particular palette segment. All of the indices of pixels within a given object class are then multiplied by the palette segment number to map them to that segment of the CLUT. The R, G, B values for all of the entries in each palette segment are then reassigned to correspond to an optimal approximation of the original image object tonal range.

This would complete the remapping of the images. Next, an image object pointer table would be created for each image to link the object identities to associated data. This would then be integrated into an interactive application for database access, training or education.

Much more sophisticated implementations of the process are possible as well. An example would be to use 24 bit color image digitizations as the original data in an automated application development system. The application developer would outline all of the objects within an object class. The system would prompt for an object class name. This name would be stored in a table in memory. The portions of the image corresponding to this object would then be copied to a buffer in memory and erased from the original display. This process would be repeated until the original image has been progressively erased to the point that the pixels representing each object class reside in discrete buffers, with the number of buffers equal to the number of object classes. The color or grayscale attributes of the pixels within each object class would then be analyzed and an adaptive color quantization algorithm (Heckbert, 1982) would be applied to optimally map each object class to a discrete segment of the CLUT and to set the R, G, B values for each segment accordingly. The object class names would then be correlated to the CLUT entries and used as a pointer table to link object identities to other related data. The image would then be rebuilt automatically onscreen and saved to disk.

Several types of links could be created between each object and other related data. One could easily design a system for a 3-button mouse so that the left mouse button caused the program to display related text, the middle button could invoke an audio description, and the right could bring up a related image. Pressing all 3 buttons could cause a related program, such as an animation or a test program, to be executed.

The MetaMap process of palette segmentation indexing has been implemented in the creation of a prototype interactive atlas of histology (figure 4). This system allows a medical student to browse through

a database of micrographic images in a manner similar to the way they look at slides in the lab. At any point, a mouse click anywhere on an image will elicit further information concerning that image. A particular mouse button combination will activate the zoom feature, which will cause the display of the next higher or lower magnification image within the database that is linked to the selected object type. This system allows the student to interact with the computer in an intuitive manner much the same as that a teaching assistant is used in the laboratory. No distracting labels are visible within the images until the student asks for further information by clicking the mouse. This system can also easily be converted into a testing mode where the computer presents images in either a predefined or random sequence and prompts the student to identify named structures by pointing to them and clicking.

Another important application of the MetaMap process involves the display of multiple color images onscreen in multimedia systems. In systems that don't use MetaMapping, only one color mapped image can be displayed at a time. When another image is loaded, this causes the palette from the new image to replace the palette from the previous image. In this case the second image looks correct, but the first image is usually distorted. If a MetaMap-based system is used, each image is treated as a separate object on the display. As new images are loaded, their palettes are dynamically loaded into discrete segments of the CLUT. This allows new images to be added without affecting ones already displayed. The presence of an object-identity pointer table would allow individual images to be modified without affecting the colors of the other images. If the CLUT has 4096 index entries, then 16 images could be simultaneously displayed with each image using 256 colors. The optimal palettes for each image could be computed in real time on a powerful workstation, or they could be computed in advance for display on less expensive machines.

MetaMapping would also be of considerable value for linking object identity information to relevant data in 3-dimensional systems for Medical imaging and scientific visualization. In such a case, bit-map methods suffer to a greater degree from the same problems encountered in 2-dimensions. The addition of a third dimension to the dataset exponentially compounds the difficulties of slow object identification speed and large data overhead. MetaMap-based systems would not suffer either of these disadvantages.

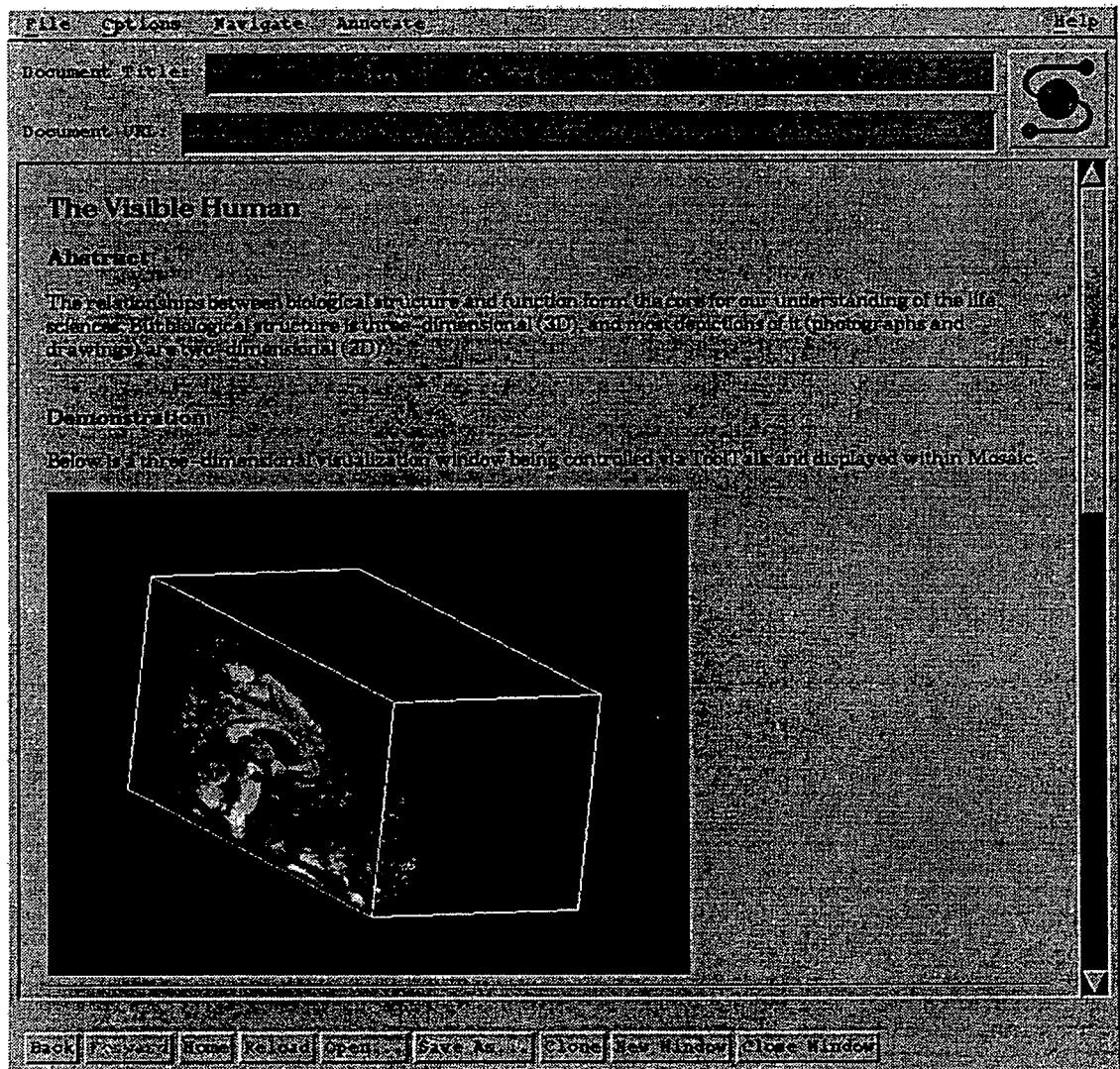
Palette segmentation indexing has additional areas of application as well. These include, but are not limited to the following: interactive catalogs; technical manuals; interactive educational systems for other image-oriented topics such as geography, the life sciences or art; intuitive interfaces for high stress military situations; archiving of aerial/satellite photographic data; diagnostic imaging annotation and archiving systems; cognitive rehabilitation for stroke and head-injury victims; foreign language training systems; and reading programs for preschoolers.

The MetaMap interactive digital video process is an attempt to redefine the role that computer graphic display images play in the function and purpose of application programs. It extends the concept of human-computer interaction to allow a user to interface directly with an image without the distractions of unwanted information or the mechanics of computer operation.

Metamapping allows useful interactive digital video applications to operate on inexpensive and commonly-available computer configurations and to utilize conventional economical mass-storage media. This process allows both direct-manipulation interfaces and hypermedia databases to begin to reach their full potentials. The tools necessary to produce truly innovative and effective interactive computer graphic presentations are now available. The most difficult aspects of generating such presentations, however, remain the initial conceptualization and design stages. Innovative tools such as the MetaMap process will not realize their fullest potentials until discipline-specific authors embrace these technologies.

The Visible Human Project

Importance of 3D data
 needed for clinical, educational and scientific purposes
 Loss of anatomists
 NLM's Visible Human project -- Standard Man/Woman
 Huge datasets -- problems in accessing/using such data
 education
 computer-aided diagnosis - deformable templates
 VR
 only a pilot -- the start of an ongoing process
 applicability of embedded IRV



The Visible Embryo Project

Much of what is known about early human morphological development was discovered through careful examination of the anatomy of embryos of human and other species at a variety of stages of growth. Early investigators prepared meticulous collections of serial microscopic sections through these embryonic

specimens. Through analysis of consecutive sections, and through handmade models built from these sections, pictures were generated of the development of the various shapes that would eventually transform into an adult organism. The accuracy and complexity of the models that could be generated, and, consequentially, the depth of knowledge about development that was learned, were constrained by the limitations of processing artifacts, human dexterity and a lack of adequate analytical tools. As a consequence, the study of development through the analysis of macro cellular structure seemed to hit a barrier that impeded significant further progress. Since then, advances in technologies for molecular biology have spurred many investigations into the molecular mechanisms of development. The important successes of these approaches have greatly shaped the study of developmental biology, and have shifted the primary focus of most investigations from macro cellular scales of interest to detailed studies of genetics, micro cellular environments and macromolecular functions. These many studies have, over the years, yielded massive amounts of information of interest to developmental investigators. Indeed the problem of accessing, synthesizing, and using this information has often been likened to trying to take a drink from a firehose.

A new branch of information science, or informatics, has sprung up to address the problems of analyzing and interpreting such massive volumes of data. Biomedical researchers are particularly needful of advances in this area. A good example of this situation concerns the Human Genome Project. Although there is no question that the creation of a complete map of human genetic sequences will be an immediate boon to many areas of research, the ultimate usefulness of such voluminous and highly detailed information resources will be severely constrained without an adequate macro cellular context within which to use the data.

Recent advances in visualization technology are beginning to allow investigators to break through these previous limitations and investigate macro cellular structure in search of universally-applicable rules for pattern formation and shape development in organisms. By applying these new technologies to the existing archives of cross-sectional image information that exist in the literature and in collections around the world, we can tap into an enormous amount of new information that can be extracted from these databases. Towards this aim, we have been working for several years on strategies to allow new information to be extracted from existing "image-centric" databases. This work initially began as an assortment of inquiries into somewhat disparate topics, but quickly congealed into a single project, "The Visible Embryo Project," that consolidates these previously scattered activities into one coherent endeavor.

Drawing from the world renowned Carnegie Collection of Human Embryology, the Visible Embryo Project begins with the creation of a large database of 3-dimensional image data of human embryos, covering all of the accepted developmental stages of human developmental stages of human development through the first trimester, as well as additional stages extending through the fetal period. Collections of light-microscopic serial cross-sections through preserved human embryos are used as the input data for the project. Since the specimens in these collections weren't originally prepared with the intention of 3-D computer reconstruction, fiducial markers were not introduced by the technicians when the tissue was processed decades ago. Among the earliest problems that we addressed in the Visible Embryo project, then, was the creation of methodologies for "reverse engineering" the morphology through analysis of information within the images of consecutive serial sections. These sets of consecutive image series were digitally processed and passed through a series of analytical procedures which created a set of images that were registered with each other, so that a voxel dataset of the 3-D structure of each embryo's anatomy was produced, similar to the voxel data produced by a conventional MRI scanner.

Various strategies need to be developed to enable these 3-D datasets to be quantitatively analyzed and compared in order to extract new knowledge concerning the mechanisms of human development. No single investigator has the skills, resources or time to address all of the research issues necessary to accomplish all of the goals of this work. We have therefore brought together a multidisciplinary research team of scientists in fields including anatomy, cell and developmental biology, surgery, image data acquisition, biomedical engineering, software development, medical imaging and visualization, educational technology, virtual reality, and many other fields in biology, medicine, engineering, and

computer science, to initiate the conception, design, and initial implementation of a national computational resource for developmental biologists. This team of collaborators across the country is working on the creation of tools for registration, segmentation, visualization, quantitative analysis & animation of embryonic anatomical developmental phenomena. One important consequence of this work will be the creation of a "morphological map" of gene expression. The three-dimensional models created by this project will provide workers in developmental genetics with a uniform "anatomical coordinate space" within which to report their findings and through which to compare findings across a large number of studies.

This project will result in the creation of what we call a national "metacenter" for developmental biology. The resulting national network of computational and data resources will serve the dual purpose of providing a testbed for new technology development in high performance computing and communications, as well as creating powerful new tools for the developmental biology research community.

This effort will establish an on-line computer-based multimedia information resource center which is physically distributed through high-speed networks among over 20 collaborating institutions across the nation, but in such a way that the distribution of resources is invisible to the user. Researchers, clinicians and students will be able to access the metacenter through any low-end Mac, PC or UNIX workstations which have Internet connections. Once on-line, the user will be able to access an enormous amount of information on developmental biology, including real time interactive control of the various 3D reconstructions of embryos from the Carnegie collection. The user doesn't even need to be aware of the fact that the images seen on the screen are being dynamically computed by a network of supercomputers distributed across the country. The power of this approach will allow the worldwide delivery of a complete medical school curriculum in human embryology through on-line access to the system.

The task of integrating access to such massive information and computational resources is nontrivial. Just one embryo from the 650 serially-sectioned specimens in the collection can yield as much as a terabyte of anatomical volume data (a 20mm specimen, sectioned at 5 microns and digitized at a resolution of 8000x8000 pixels/section at 36bits RGB produces 1.073 TB of voxel data). It is clear that no single workstation, or even supercomputer, can manipulate, process and analyze such a large quantity of data as a single unit, much less perform computational operations on a database of hundreds of such datasets. For this reason, the Visible Embryo team members at UCSF have developed tools to allow integrated Internet access (through NCSA Mosaic) to remote volume visualization engines which can distribute computation across a large number of graphics supercomputers connected by high-speed networking. This allows the integration, through NCSA Mosaic and the World Wide Web, of text-based, image, audio, and video data with real-time interactive control of high-performance visualizations embedded within Mosaic documents. We are also exploring the potential of using this technology for delivering interactive access to virtual reality applications through the Internet. The success of these efforts will allow widespread access to highly accurate and complex virtual reality simulations of human developmental anatomy, using inexpensive workstations and personal computers.

As biology moves into the information age, biologists can no longer sit idly aside and wait for the computer scientists and engineers to solve their problems with respect to dealing with the deluge of information entering the field at a constantly-increasing rate. It will be only through taking a proactive role in as part of a multidisciplinary team that biological researchers will be able to influence the directions of progress in the information sciences in ways that will support advances in the study of biological structure, function and development.

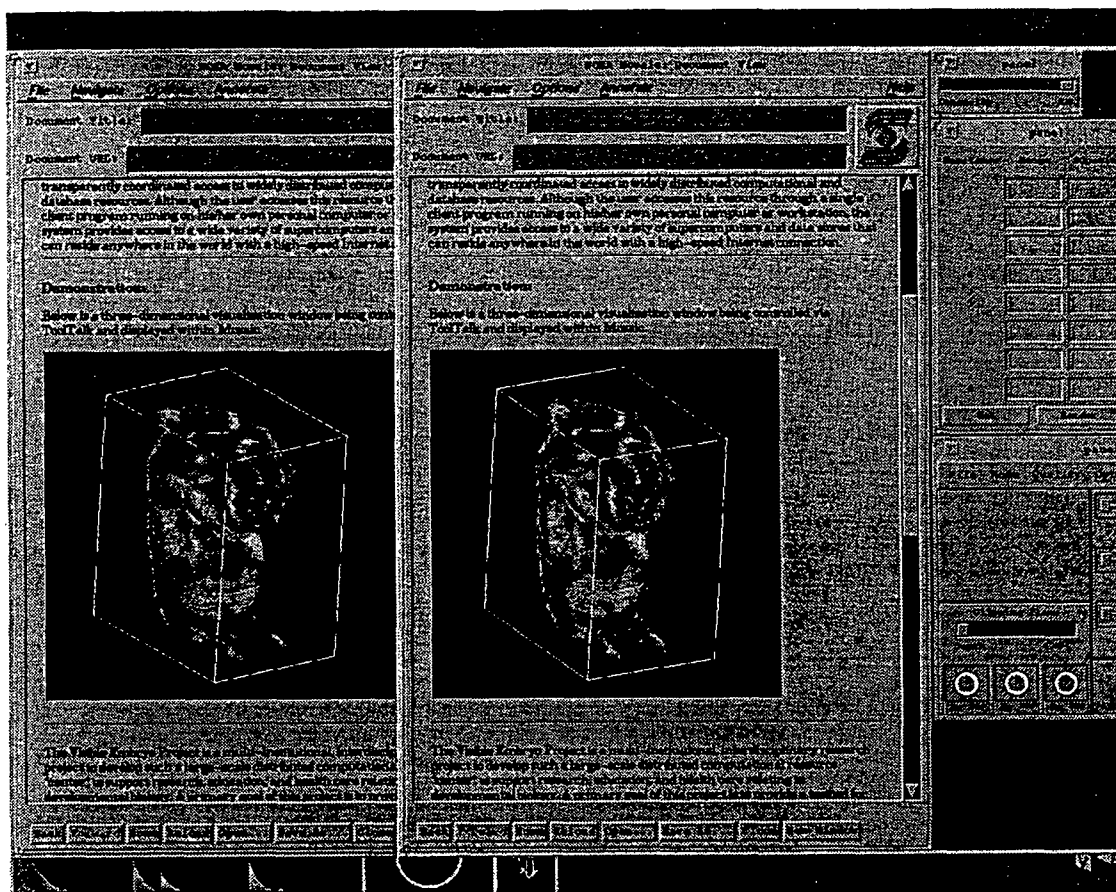


Figure 1: A stereo-pair illustration of interactive real-time control, embedded within an NCSA MOSAIC document, of a 3-dimensional volume reconstruction of human embryonic anatomy, showing a 7-week old human embryo found in the Carnegie Collection as 1997 serial cross sections. This technology was developed by the Center for Knowledge Management at the University of California, San Francisco, and was demonstrated there in November, 1993.

Molecular/genetic visualization

The recent revolution in genetic engineering is having an immense and growing impact on both science and industry and is likely, over the next several decades, to have consequences comparable to the electronic revolution of the present and previous decades. Two of the key unsolved problems in this burgeoning field are (1) the prediction of the three dimensional structures of proteins and other polymers from knowledge of their monomer sequences and (2) correlation of three-dimensional structure with catalytic activity. These closely related fields of research are both of fundamental scientific importance and considerable potential economic benefit.

DNA, the double helix, carries the genetic code and is the prime raw material of the genetic engineer. The coding sequences of DNA can now be determined so rapidly that the primary sequences of several tens of thousands of proteins are known. This number is increasing very rapidly. But the ultimate *function* of a protein depends on its three dimensional structure, and the three dimensional structures of only a few hundred proteins are available from long, laborious and expensive X-ray diffraction analyses. New structures appear at the rate of only a few dozen a year. This rate is hardly increasing, and certainly cannot match the rate of determination of genetic coding sequences. Although techniques such as

multidimensional NMR are now contributing valuable structural data to our knowledge base, no new methods are on the horizon which will provide the orders of magnitude increase in productivity required. Thus, it is critically important for scientists to utilize the existing database of protein structures with the most efficient, informative, and mentally stimulating means possible. While a couple of decades ago the case often had to be made that interactive computer graphics provided a real advantage to scientists and was not merely a generator of pretty pictures, it is now recognized by even the early critics that interactive computer graphics is a routine and useful tool in molecular modeling. Computer graphics requires numbers to generate its pictures, but the main value lies not in the numbers themselves, but in the insight gained when the graphics is used as a window on the process or system being studied. Computer-assisted insight is not always quantifiable, but is nonetheless real. High quality interactive computer graphics, with a well-designed graphical interface for the scientist user whose main interest is in chemistry or molecular biology, is an *idea generator*.

The UCSF Computer Graphics Laboratory has had a long history in designing, building and disseminating interactive molecular modeling tools. The laboratory receives the majority of its funding from the National Institutes of Health, National Center for Research Resources (grant RR-1081). The specific aims of the Laboratory are:

- To design, build, integrate and disseminate computation and visualization tools for structural biology and volumetric biomedical imaging
- To apply these software tools to problems in medicinal chemistry and molecular biology with particular emphasis on elucidation of molecular structure using nuclear magnetic resonance, studies of molecular interaction, drug design and protein engineering
- To apply these software tools to problems in medical radiology and human developmental anatomy....
- To assemble hardware with which the software can best be utilized
- To provide access to, training in, and assistance with this software and hardware for outside users
- To disseminate as documented source code the software developed at the resource to allow others to use the software both for research application and as a starting point and training tool for their own developments

File Options Navigate Annotate Help

Document Title: [REDACTED]


Document URL: [REDACTED]

Protein Data Bank - 2MBA: Myoglobin (Azide Complex)

Author(s): M.Bolognesi, S.Onesti, G.Gatti, A.Coda, P.Ascenzi, M.Brunori
 Revised: 15-Apr-92
 Revised: 15-Jan-90

M.Bolognesi, S.Onesti, G.Gatti, A.Coda, P.Ascenzi, M.Brunori: *Aplysia Limbina Myoglobin. Crystallographic Analysis At 1.6 Angstroms Resolution* Journal of Molecular Biology, v. 205 529-1989 ASTM |MOBAK UK|ISSN 0022-2836

Visualization:



References:

M.Bolognesi, G.Gatti, M.G. Guanzacodi, A. Mattevi, S. Onesti, A. Coda, P. Ascenzi, M. Brunori: *Aplysia Limbina Myoglobin. Molecular Rules For Ligand Binding And Reversible Unfolding* To Be Published

M.Bolognesi, A. Coda, G. Gatti, P. Ascenzi, M. Brunori: *Crystal Structure Of Ferric Aplysia Limbina Myoglobin At 2.0 Angstroms Resolution* J. Mol. Biol. V. 183 113-1985 ASTM |MOBAK UK|ISSN 0022-2836

M.Bolognesi, S. Onesti, P. Ascenzi, G. M. Giacometti, A. Merli, M. Brunori: *Reactivity Of Ferric Aplysia And Sperm Whale Myoglobins Towards Imidazole. X-Ray And Binding Study* J. Mol. Biol. V. 158 305-1982 ASTM |MOBAK UK|ISSN 0022-2836

Back Forward Home Reload Open... Save As... Clone New Window Close Window

One of the key software packages that has been developed at the UCSF Computer Graphics Lab is MidasPlus (Molecular Interactive Display and Simulation). This is an interactive graphics program designed for the efficient display and manipulation of proteins and nucleic acids, including very fast van der Waals surface calculations, space filling images with shadows cast from multiple light sources, and

real time "ribbon" diagrams for depicting protein secondary structure and nucleic acid helices and base pair planarity and directionality. MidasPlus emphasizes highly interactive display and manipulation, with a data structure designed for interactive selection, manipulation and docking of drugs and receptors. More than 500 papers have been published based on work done with MidasPlus and its predecessor, Midas. Through a licensing program developed by the Regents of the University of California, MidasPlus has been distributed to over 300 other laboratories. MidasPlus currently runs on the Silicon Graphics "IRIS" and "Indigo" family of workstations, Digital Equipment's "Alpha AXP" workstations equipped with an optional 3-D graphics card, and, with more limited capabilities, on the NeXT color workstation.

Linking information

Toni's stuff

Online Publishing

Despite the significant growth in digital networks and in the number of publications in digital form on those networks, there have been few attempts to create a comprehensive set of digital authoring tools which would allow authors, editors, and reviewers to collaborate in a networked environment to produce fully integrated documents. To be effective, these tools should be accessible and easy to use and be capable of performing all of the tasks--manuscript creation, editorial comments and reviewer tracking, copy editing and a final formatting step prior to publication--that are necessary for publishing, in print, on a network, or in a database of articles.

Authoring packages have been around for some time, but they vary by discipline. Mathematicians routinely use complex procedural coding such as eqn in Unix, and TeX or LaTeX on a variety of platforms. Generally, these author packages decline in sophistication as one moves away from computer science and mathematics to other fields such as biology and chemistry. The authors in these disciplines tend to use macro packages for specific publications, which are supplied by the publisher. Furthermore, these macro packages tend to be limited to the more popular word processing programs such as WordPerfect or MicroSoft Word. Disciplines such as medicine are even less sophisticated and very few authors use any tailored software. At the editorial level most journal publishers have customized software to aid editors in the management of the editorial process; that is to maintain reviewer lists and track the status of manuscripts. These programs typically feed this information into a database that can be used to generate statistical reports and project publication schedules. Very few tools exist, however, that assist in the editing and reviewing of electronic documents that contain complex mathematics, tabular material, graphic data, and video.

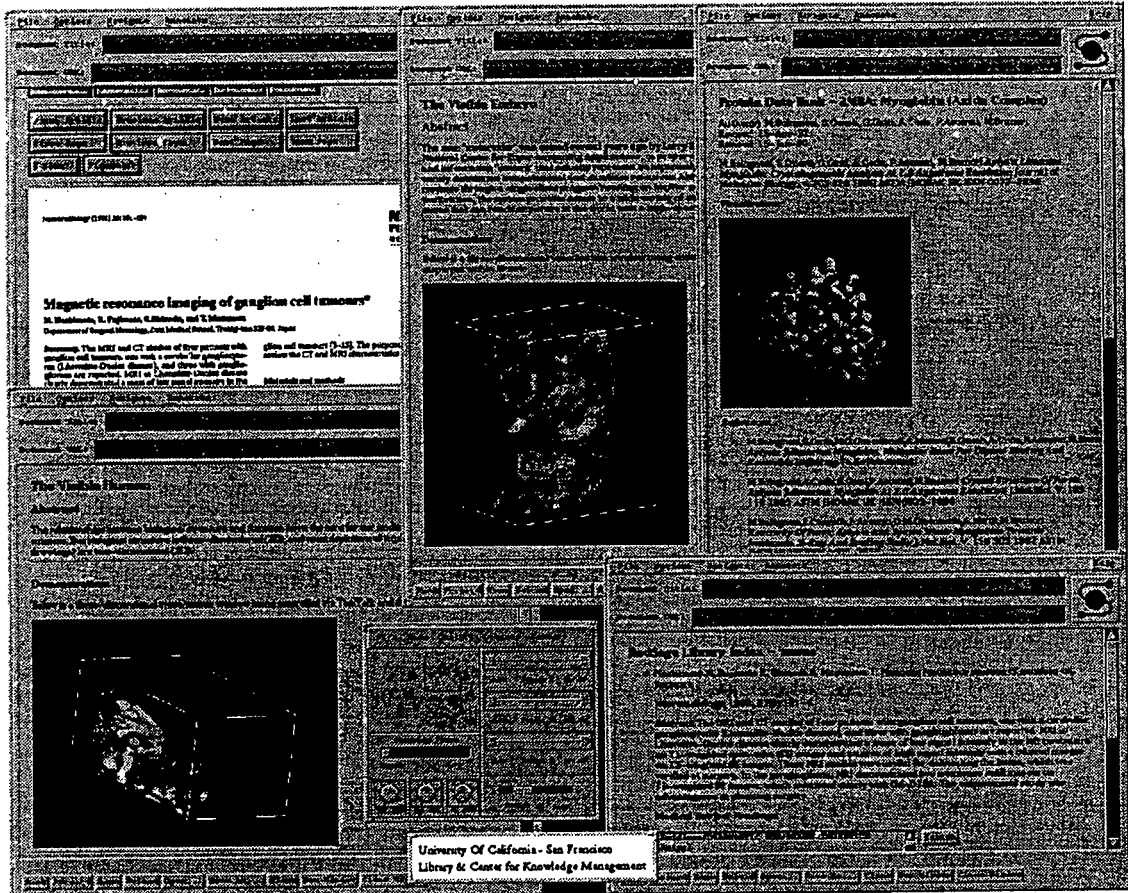
In the proposed system, we will develop a set of software tools for: (1) creating in a WYSIWYG environment a manuscript that includes features such as HTML+ coding with embedded links to visualizations, datasets or to the author's original dataset. (2) assisting the journal editor in the efficient and thorough execution of the peer review process. The goal here would be to preserve the anonymity of the process while enhancing the speed and depth of communication between authors, editors and reviewers. (3) formatting a reviewed and accepted manuscript in preparation for publication (be it in print or electronic form). This formatting program must support all different content (e.g. text, tabular, graphics) and technical (e.g. TeX, eqn, tbl) elements/requirements that may be present in the document.

An Integrated System

Integration is the key

MOSAIC allows that integration

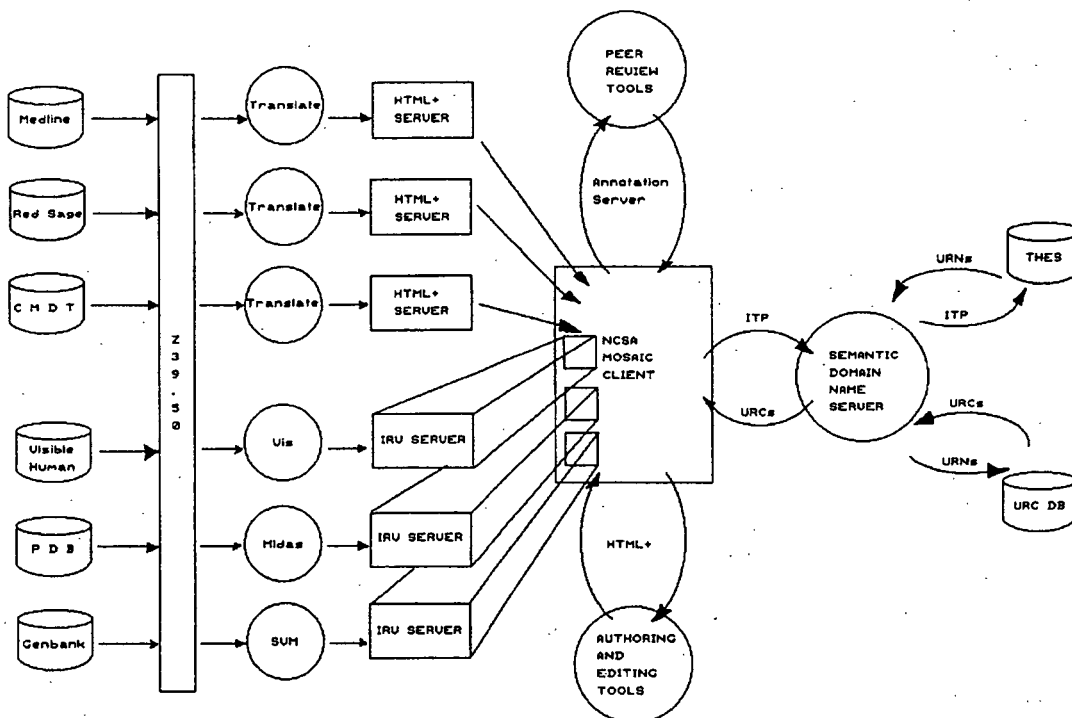
Common user interface, platform independent, distribution of computational load, free client, ubiquitous (de-facto standard)



Perhaps the greatest benefit of this research is that it will provide national leadership in a number of areas critical to the success of the coming National Information Infrastructure. The presently-proposed project will begin to produce a body of information that others can use in their efforts to plan the on ramps, off ramps, bridges, and toll booths for the next century's national "information superhighway."

Work Proposed:**General System Strategy:**

Below is a diagrammatic overview of the structure of the proposed system:



Definitions:	HTML+:	Hypertext Mark-up Language -- This is the language that World Wide Web databases are encoded in, and that MOSAIC interprets.
	IRV Server:	UCSF CKM's Interactive Remote Visualization Server -- This allows interactive real-time visualization tools to be embedded into MOSAIC documents.
	Vis:	UCSF CKM's distributed remote volume visualization tool
	Midas:	UCSF CGL's molecular visualization package
	SVM:	Sequence Visualization Module -- An as-yet unnamed tool for graphical display of genetic sequence data.
	ITP:	Informal Text Phrase -- A user-entered search term, or a word or phrase of text that the user highlights from within a document.
	URN:	Universal Resource Name -- A persistent, location-independent identifier for an object.
	URL:	Universal Resource Location -- The address of an object. It contains enough information to identify a communications protocol and retrieve the object.
	URC:	Universal Resource Characteristics -- Any combination of one or more URNs or URLs with meta information (e.g. author, format, compression method).

Explanation of system diagram:

The system will draw from a number of fundamental databases including bibliographic data (Medline) in the form of MARC records, journal publication data (Red Sage) in the form of SGML header and Postscript files, encyclopedic reference text data (CMDT) stored in an object-oriented SGML database, volumetric anatomical data (Visible Human Project) stored as NCSA HDF datasets, protein structure data (Protein Data Bank) stored as PDB files, and genetic sequence data (Genbank) stored as compressed ASCII strings (? , I'm guessing about Genbank).

These databases will reside behind a Z39.50 interface layer which yields, to the requesting client, the respective dataset in its native form. This data then goes through a translation layer where the data is either translated directly into HTML+ (Medline, Red Sage, CMDT) or loaded into a native-data visualization tool (Visible Human, PDB, Genbank). The HTML+ code is then passed to a set of HTML+ servers, which can be browsed by the MOSAIC client. The visualization data is handled differently. The graphical I/O of the relevant visualization tool is passed to an interactive remote visualization (IRV) server, which handles both mapping of the display output from the visualization tool onto embedded live-visualization windows within the MOSAIC-browsable HTML+ documents, as well as capture of user-entered mouse and keyboard events within the visualization windows and transmission of those mouse and keyboard events back to the relevant visualization tools. The user, browsing the system with the project's enhanced version of the MOSAIC client, is presented with data and visualizations derived from these various databases, yet embedded into coherent, multimedia MOSAIC documents.

For multimedia documents that have been explicitly pre-composed, the linking of these various data resources can take the form of universal resource names (URNs) that are encoded as tags into the HTML+ documents. This is passed to the system's semantic domain name server, for resolution of the information object's location and retrieval means. The URNs are used as indices in order to look up the relevant universal resource characteristics (URCs) in a URC database, which yields the universal resource location (URL), or physical address, of the information object in question.

Semi-automatic means will be provided for a user to search for arbitrary information objects on the system by either keying in a search word or phrase, or by highlighting a not-already-hyperlinked section of text that (s)he happens to be viewing within the MOSAIC client at the time. This informal text phrase (ITP) is then passed to the semantic domain name server, which passes it on to a universal resource thesaurus (which will incorporate elements of the NLM's UMLS system). The thesaurus compares the ITP to its database of terms and phrases and returns a rank-ordered list of URNs that are likely to match the object in question. These URNs are then passed to the URC database for resolution of URLs that point to information objects on the Internet that are most likely to match the ITP that the user employed to initiate the search. The user is presented with a rank ordered set of textual descriptions of likely matches which are hyperlinked, via their URLs, to the data in question. Clicking upon a selection from this list loads the related data into the relevant visualization server (IRV) or HTML+ server, and a second MOSAIC window pops up to allow viewing or interaction with that dataset.

A set of authoring and editing tools will be designed to allow the interactive WYSIWYG creation of HTML+ documents, as well as allowing the embedding of visualizations, etc., which can be created using the interactive remote visualization tools, and which can use data from the various scientific databases mentioned above. Alternatively, the author can use his/her own datasets, which would be uploaded to an Internet-accessible World Wide Web server. The journal editor can use the same set of tools to edit submitted articles and to communicate changes to the text with the author. This, of course, would occur in a private, access-controlled, area of the system, so that confidentiality of the material to be published can be controlled.

It is important to understand that there are two distinctly different types of links involved in this system...

Other private, access-controlled HTML+ servers will be used to administer the peer review process. A modification of NCSA's MOSAIC-based group annotation server will be developed to allow the journal editor to exercise precise control and documentation of each reviewer's comments and suggestions.

Detailed Work Plan

Foundation Databases

Following is a sketch of the workplan for the Z39.50 interface to RightPages in the first year of the grant. I have not included development of the basic SGML technology that AT&T will be funding independently:

Z39.50 interface to RightPages (year 1)

#	Months	Task
	2	Design (protocol translation, client and server components)
	5	Implementation of all components
	2	Component testing
	1	Integration into RedSage system and testing
	2	Packaging and documentation (both system and RuserS)

Our plan is to have our contractor contribute to the implementation of the various SGML databases: CMDT, URC, and the semantic domain server and thesaurus components. Are you specifying workplan details for those efforts? I guess I am not sure at the moment how long they will take without learning

more about the contributions of the other collaborators, plus being further along in the development of the underlying OO SGML stuff itself. Generic database development, depending on complexity, might look something like this: # Months Task

2-4	DTD definition, plus any additional class definitions
0-4	implementation of any additional class methods etc.
0-2	implementation of any additional content prep or import tools and components
1-2	initial document data load and testing

Translation Modules

MOSAIC-based Browser-Editor

Interactive Remote Visualization Servers

MOSAIC Extensions

Browsing vs detailed views

Vis

Midas

Primary goals of the molecular visualization component of the presently-proposed project include:

- To develop an "Interactive Molecular Visualizer" for display and manipulation of molecular structure data, based on UCSF CGL's MidasPlus package, which will interface with the enhanced MOSAIC client developed at UCSF's CKM. To use a client/server architecture in order to maximize the number of hardware platforms capable of interactively viewing three dimensional molecular structure;
- To convert the exiting MidasPlus User's Manual (130 pages, including an extensive index) into Hypertext Markup Language (HTML), so that an on-line hypertext-based user's guide is available;
- To collaborate on the design and implementation of a Semantic Domain Name Server with the specific goal of integrating protein structure data sources such as the Protein Databank [5] into the MOSAIC HTML framework. We intend not only to provide direct access to the PDB database (at least for experimental purposes), but also to provide a mechanism for a server to locate and access on-line protein structure data remotely as a part of "following" an HTML link.
- As resources permit, to integrate additional structure and sequence visualization tools being developed at the Computer Graphics Laboratory into the CKM enhanced MOSAIC client.

To interface MidasPlus and the enhanced MOSAIC client, CGL will collaborate with the CKM on two key project components: the requirements analysis and formulation of design specifications for data processing to be performed locally on a MidasPlus/MOSAIC client versus that processing to be performed on a more powerful server (i.e. division of functionality between client and server); and the design of the communications protocol which will be used between the client and server. The client-server division is an important software architectural decision because it greatly influences the performance optimization possibilities. By properly dividing the work load between client and server, it is possible to reduce simultaneously the data communications between the applications as well as the computational load on each end. A careful design should permit the use of "low-end" hardware platforms such as Intel 486-based PC's for clients, and thus make interactive three dimensional molecular visualization available to a significantly larger fraction of the scientific user community. The actual communications protocol used between client and server is also an important aspect of the software architecture. A good protocol will facilitate the software development, while a poor protocol will create more work during implementation and reduce efficiency.

Molecular structure data generally consist of a set of atomic coordinates rather than the regularly spaced sample points that an MRI scanner produces; hence the images generated from molecular structure data are more suitably described using an algorithmic format such as OpenGL [6,7]. Once a suitable data exchange format between the molecular visualizer and MOSAIC is chosen, we plan to implement the visualizer as a remote service available to the enhanced MOSAIC client. The implementation will also include performance analysis to determine whether interactive response is sufficient and to locate bottlenecks where improvements are needed

Of critical importance to the successful deployment of our new molecular visualizer for MOSAIC is the availability of high quality documentation for the scientist user. Since it is desirable that even casual users be able to efficiently manipulate and control the molecular structures that they see, we need a user "manual" that may be browsed as a standard MOSAIC HTML+ document. As stated

above, we plan to convert our existing MidasPlus User's Manual (see appendix) written using UNIX "troff" formatting commands into HTML. Although this task does not present any particular challenges intellectually, it will nevertheless be demanding since the schedule calls for us to begin this project before the new authoring and editing tools are completed. We must still be able to generate a printed version of the manual (using PostScript) in addition to the MOSAIC-viewable document, so there are several issues that must be dealt with if we are to be successful.

RE: Semantic Domain Name Server...

Semantic Domain Name Server

**Kazic's stuff
URN Thesarus
URC Database**

Authoring and Editorial Tools

**Work Plan for Authoring/Editing Tools
Phase I (6 months)**

Interview and hire scientist for technical work Under the direction of Bob Badger and Craig Van Dyck, Springer-Verlag New York will interview and hire a computer scientist who has sufficient background and knowledge to develop authoring/editing tools that are suitable and consistent with the aims of the project.

Installation of hardware and software

Under the direction of Craig Van Dyck, the scientist will install and configure hardware and install Galaxy, Avalanche and SoftQuad software systems. Thorough testing of these systems connecting with the University of California, San Francisco (UCSF) and AT&T Bell Labs through Internet will be performed at Springer-Verlag New York's office.

Phase II (12 months)

I Develop author package and sample content

--Needs analysis: Determine the specific needs of the authoring package by becoming familiar with the publishing process; and by understanding the requirements of UCSF's Center for Knowledge Management and the National Center for Supercomputer Applications' HTML+ coding.

--Develop and review the specifications of the authoring program with all of the principal project investigators.

--Evaluate existing programs, tbl and other editors, to determine which could be adapted or modified. It is very probable that existing editors in Unix or others could be modified to fulfill most but probably not all of the program specifications.

--Develop final program.

--Work with Washington University, UCSF's CKM, and UCSF's CGL to develop sample content.

--Work with UCSF and Washington University to test the program with sample content.

Phase III (12 months)

I Develop journal editor and peer review program.

--Needs analysis: assess requirements of journal editors' work flow; UCSF's hardware requirements; and information about journal editing procedures from the staff of Springer-Verlag New York.

--Develop specifications for program. Revise specifications with Springer New York staff and all key project participants.

--Write a test program with author package and content.

Phase IV (18 months)

I Adapt an existing commercial program to format content from SGML-based URC database.

--Access in cooperation with Bell Labs and UCSF's CKM the requirement for a program to store documents in the database.

--Develop and review specifications of the formatting program.

--Assess commercial programs that meet specifications.

--Assess storage and retrieval with sample documents.

**HTML+ Authoring, with Embedded Objects
Annotation Server
Peer Review
Editorial Process**

Testbed Facility

WWW server

Beta Testing

Evaluation

Expected Accomplishments

Timetable

Year 1

Year 2

Year 3

Year 4

Plans for Preservation, documentation and dissemination of results

Set up a WWW server

Documented source code

Papers presented at meetings

Results from Prior NSF Support

Siggraph '92 -- Showcase demonstration funded by a combination of NSF and NIH money:

3-D Visualization Tools

Software tools were developed to allow the interactive three-dimensional visualization of the embryo reconstruction in real time. Figure 3 shows the display of the application as it appeared at the

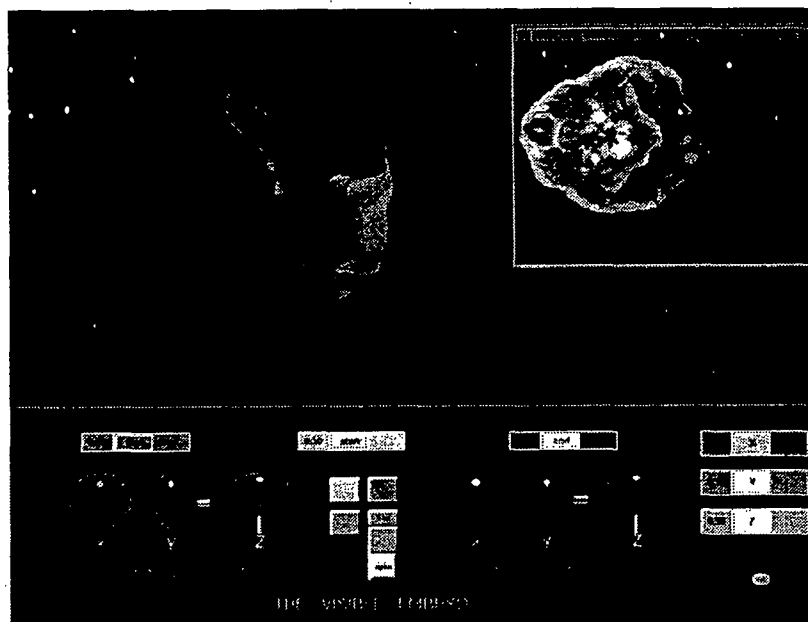


Figure 3: Screen image of the 3-D visualization program demonstrated at SIGGRAPH '92, in Chicago. This application was distributed between a Silicon Graphics Crimson VGXT workstation and a Convex supercomputer.

SIGGRAPH '92 conference in Chicago (Doyle, et al., 1992). The left of the screen shows a surface-based model of the embryo's exterior. This model was built from data which was derived, through three-dimensional interpolation, from the original embryo dataset. Two-hundred volume slices of the embryo (stored as texture maps) can be interactively displayed at this lower resolution while the model is rotated freely in three dimensions. A cutting-plane can be seen to intersect the surface-based model. This cutting plane can be interactively controlled to intersect with the embryo model at any arbitrary angle and position. To the right of the screen, one can see a window that displays a high-resolution image of the oblique section through the embryo as indicated by the interactive cutting plane. In order to maintain the quick response needed for effective real-time interaction, the computational load of this application was distributed so that the interface panel, seen at the bottom of the screen, and the 3-D surface model were running on the CPU of the Silicon Graphics workstation. Computation of the high-resolution oblique section image displayed in the right window took place on the Convex supercomputer. Both of these operations occurred simultaneously, communicating through a high-speed fiber optic network.

Virtual Reality

The surface-based embryo-model described above was also implemented within a virtual reality environment, called "The Cave," at the 1992 SIGGRAPH conference. The Cave was a 10' x 10' room made up of back-projection screens upon which were projected stereo views of three dimensional data

that a viewer, wearing LCD stereo glasses and a 3-D tracking device, could move around in as the system tracked his or her motion. One could walk around the data and receive the sensation that one was actually "in" the computer graphic environment. This system was also shown at the Supercomputing '92 conference and the 1992 meeting of the Radiological Society of North America.

Bibliography

1. Bush, V.: As We May Think, in *The Atlantic Monthly*, July 1945: 101-108.
2. Campbell, G., T. DeFanti, J. Fredericksen, S.A. Joyce, L.A. Leske, J.A. Lindberg and D.J. Sandin: Two Bit/Pixel Full Color Encoding, *Proc. ACM SIGGRAPH*, August, 1986, 20/4:215- 223.
3. Doyle, M.D.: Method and Apparatus for Identifying Features of an Image on a Video Display, United States Patent #4,847,604, July 11, 1989.
4. Doyle, M.D.: The Interactive Digital Video Interface, in *Spatial Display and Spatial Instruments: Proceedings of a conference sponsored by NASA Ames Research Center*, NASA Conference Publication 10032, 1989, pp 24-1 to 24-4.
5. Foley, J.D.: Interfaces for Advanced Computing, *Sci. Am.*, October, 1987, pp 127-135.
6. Heckbert, P.: Color Image Quantization for Frame Buffer Display, *Proc. ACM SIGGRAPH*, July, 1982: 297-307.
7. Hutchins, E.L., J.D. Hollan and D.A. Norman: Problems with Direct Manipulations, in *Readings in Human-Computer Interaction*, Ed. R.M. Baecker and W.A.S. Buxton, 1987, pp 468-471.
8. Murch, G.M. and J.M. Taylor: The Effective Use of Color in Graphics, in *Color in Computer Graphics: ACM SIGGRAPH Tutorial Notes*, August, 1988: 1-33.
9. Nelson, T.: Managing Immense Storage, *Byte*, Jan. 1988: 225- 238.
10. Schneiderman, B.: The Future of Interactive Systems and the Emergence of Direct Manipulation, *Behavior and Information Tech.*, 1982, 1:237-256.
11. Schneiderman, B.: *Direct Manipulation: A Step Beyond Programming Languages*, *IEEE Computer*, August, 1983, New York: IEEE Computing Society.
12. Silverstein, L.D.: Human Factors for Color Display Systems: Concepts, Methods, and Research, in *Color and the Computer*, H.J. Durrett (Ed), Academic Press, NY, 1990
13. Thursh, D.R., F. Mabry and A. Levy: The Knowledge Access, Management and Extension System in Pathology, in Levy, A.H. and B.T. Williams (eds) *Proceedings AAMSI Congress*, 1985, pp 365-406.

14. Thursh, D.R., F. Mabry and A. Levy: Computers and Videodisks in Pathology Education: ECLIPS as an Example of One Approach, *Human Pathology*, 1986, 17/3:216-218.
15. Glickman, J., W. Eicholzar, D. Screnci and L. Glass: PC- Based Imaging, *Biomed Imaging & Comm.*, November, 1987, pp B12-B16.
16. Osgood, D: The Difference in Higher Education, *Byte*, 1987, 12/2:165-183.

Biographical Sketches

Doyle
Ferrin
Kazic
Badger
Story
Hardin

Letters of commitment

Organizational Roles:

UCSF CKM:

- Development of Z39.50-compliant experimental (subset) databases for storage of Visible Human data, PDB data, and Genbank sequence data.
- Cooperation with AT&T in the development of an object-oriented SGML-based database for the Handbook of Current Medical Diagnosis and Treatment (CMDT)
- Development of an experimental Z39.50 interface to Medline data (will be unnecessary if DLA can provide such an interface to Melvyl Medline early enough into the project

UC's
timeline)

- Development of translator servers to translate Medline MARC records, CMDT SGML data and Red Sage SGML/Postscript data into HTML+
- Development of a set of HTML+ documents that act as browsers to Medline, CMDT, and Red Sage

- Refinement and further development of Vis to allow better distribution of computation and better integration with MOSAIC.

- Cooperation with CGL to adapt Midas for integration within MOSAIC, and to identify and adapt a suitable program for graphical display of genetic sequence data.

- Refinement and further development of the interactive remote visualization server, and its incorporation (with NCSA's help) within the MOSAIC environment.

- Development, in cooperation with NCSA, of an enhanced version of the MOSAIC client to allow easier integration of external programs within MOSAIC-readable documents.

- Development, in cooperation with Springer-Verlag and NCSA, of an interactive WYSIWYG editor for creation of HTML+ documents, and for embedding visualizations created using CKM's IRV tools, as well as development of a modified version of NCSA's group annotation server to support the peer review process.

- Development, in cooperation with AT&T, of an object-oriented SGML-based URC database

- Development, in cooperation with UCSF's CGL, UCSF's Radiology Dept., Washington Univ., and AT&T, of a Semantic domain name server and a URN Thesaurus, based upon AT&T's object-oriented SGML database technology.

- Development, in cooperation with UCSF's CGL, UCSF's Radiology Dept., Washington Univ., and Springer-Verlag of a set of sample content for use in evaluating the effectiveness of the system, as well as for demonstration of the results of the project.

UCSF CGL:

- Cooperation with CKM to adapt Midas for integration within MOSAIC, and to identify and adapt a suitable program for graphical display of genetic sequence data.

- Contributing to the refinement and further development of the interactive remote visualization server, and its incorporation (with NCSA's help) within the MOSAIC environment.
- Development, in cooperation with UCSF's CKM, UCSF's Radiology Dept., Washington Univ., and AT&T, of a Semantic domain name server and a URN Thesaurus, based upon AT&T's object-oriented SGML database technology.
- Development, in cooperation with UCSF's CKM, UCSF's Radiology Dept., Washington Univ., and Springer-Verlag of a set of sample content for use in evaluating the effectiveness of the system, as well as for demonstration of the results of the project.

Washington University:

- Development, in cooperation with UCSF's CKM, UCSF's CGL, and AT&T, of a Semantic domain name server and a URN Thesaurus, based upon AT&T's object-oriented SGML database technology.
- Development, in cooperation with UCSF's CKM, and UCSF's CGL., and Springer Verlag of a set of sample content for use in evaluating the effectiveness of the system, as well as demonstration of the results of the project.

AT&T Bell Laboratories:

- Development of Z39.50 interface to the RightPages server..
- Cooperation with CKM in the development of an object-oriented SGML-based database for the Handbook of Current Medical Diagnosis and Treatment (CMDT)
- Development, in cooperation with CKM, of an object-oriented SGML-based URC database
- Development, in cooperation with UCSF's CGL, UCSF's Radiology Dept., Washington Univ., and CKM, of a Semantic domain name server and a URN Thesaurus, based upon AT&T's object-oriented SGML database technology.

Springer-Verlag:

- Development, in cooperation with UCSF's CKM and NCSA, of an interactive WYSIWYG editor for creation of HTML+ documents, and for embedding visualizations created using CKM's IRV tools, as well as development of a modified version of NCSA's group annotation server to support the peer review process.
- Development, in cooperation with UCSF's CKM, and UCSF's CGL., and Washington Univ. of a set of sample content for use in evaluating the effectiveness of the system, as well as for demonstration of the results of the project.

NCSA:

- Cooperation with CKM in developing an enhanced version of MOSAIC to allow easier integration of a client module for CKM's interactive remote visualization server.

- Cooperation with CKM and Springer-Verlag in the modification of NCSA's group annotation server to facilitate the peer-review process.

Budget

Resource requirements for AT&T are easy; we will supply all equipment required for our side and only pay for an external contractor with grant money:

line item	% effort	yr 1	yr 2	yr 3	yr 4
B2: 1 programmer		100 sal	123,750.00	129,937.50	136,434.37 143,256.09

Unless I misunderstand, we will pay for travel, space, etc for this programmer out of the 25% of our total programmer-related costs.

Sorry to be so late with this. A bunch of post-holidays crises all converged recently, but they are by and large laid by now. I will improve my performance in the future.

Cost Sharing**Springer-Verlag facilities in New York**

Springer-Verlag New York occupies six floors in the historic FlatIron building at 175 Fifth Avenue in the heart of New York City with approximately two hundred employees at the location. Most of the employees in New York are engaged in various activities related to professional and reference publishing. The authoring and editing tools development will be carried out within our production department which has approximately forty employees. These employees are engaged in all aspects of book/journal publishing. After accepting manuscripts from authors, they oversee copyediting, composition, final page layout and the production of files for final printing. The production facilities in New York have access to Internet through a newly installed node and it is planned that the authoring/editing project will make use of the Internet node facility.

In-kind Contributions

As expressed in the budget of the editing/authoring portion of the project, Springer will contribute at least 25% of the project's cost. Springer's contribution includes an office for the project scientist at Springer's US headquarters in New York City. In addition, Springer expects to provide the scientist on the project with secretarial help when needed, and use of office equipment such as a PC, laser printer, fax, and photocopy machine as well as other office necessities.

More important, though, Springer will contribute the management of the project as well as its considerable knowledge of the authoring, editing, and publishing process during the development of the network tools. Springer estimates Dr. Badger will contribute at least 30% of his time to the project and, additionally, two production managers at Springer will also contribute at least 20% of their time each.

Current and Pending Support

Appendices

Midas Manual

Red Sage docs

Visible Embryo Project description

Stereo viewer & explanation

Digital Libraries Grant Budget										
Year 1										
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total Directs	Indirects	Contract DOI	Grant Requested	
UCSF CKM	43%									
Doyle	1	30%	99,750	29,925	28%	38,304	16,471	0	54,775	
Marlin	1	40%	67,000	26,800	28%	34,304	14,751	0	49,055	
Ang	1	50%	51,000	25,500	28%	32,640	14,035	0	46,675	
Stavri	1	35%	53,000	18,550	28%	23,744	10,210	0	33,954	
Salomon	1	40%	50,000	20,000	28%	25,600	11,000	0	36,600	
RA 1	0.5	100%	15,000	15,000	28%	19,200	8,256	0	27,456	
RA 2	0.5	100%	15,000	15,000	28%	19,200	8,256	0	27,456	
Subtotal				150,775		192,992			275,979	
UCSF CGL										
Ferrin	1	10%	103,126	10,313	22%	12,530	5,388	0	17,918	
Petersen	1	20%	45,914	9,183	28%	11,708	5,034	0	16,743	
Huang	1	50%	64,216	32,108	28%	40,938	17,603	0	58,541	
Subtotal						60,000	25,800		85,800	
Wash U	57%									
Subtotal									175,705	
NCSA										
Subtotal									268,175	
AT&T	0%									
Subtotal									123,750	
Springer	35%									
Subtotal	1								155,769	
UCSF Personnel Subtotal									361,779	
Subcontractor personnel subtotal									542,549	
Equipment				Item cost	Qty	Cost	Tax			
Collaborator workstations				3,087	15	46,305	3,936		50,241	
SGI Onyx w/ RE2				74,792	1	74,792	6,357		81,149	
Alpha (cluster)				4,630	8	37,040	3,148		40,188	
Maintenance									48,042	
Supplies										
Galaxy				4,000	4	16,000	1,200		17,200	
Avalanche				9,000	2	18,000	1,350		19,350	
SoftQuad A/E				900	3	2,700	203		2,903	
App Builder				2,500	4	10,000	750		10,750	
MO disks				115	10	1,150	86		1,236	
Travel										
UCSF/NY,NJ round trip				850	15	12,750	956		13,706	

UCSF/St Louis round trip				650	9	5,850	439		6,289
UCSF/Urbana round trip				650	6	3,900	293		4,193
Other									
Subtotal						228,487	18,718		295,247
Year 1 TOTALS									1,199,575
=====									
Year 2									
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total Directs	Indirects	Contract DOI	Grant Requested
UCSF CKM	43%								
Doyle	1	30%	104,738	31,421	28%	40,219	17,294	0	57,514
Martin	1	40%	70,350	28,140	28%	36,019	15,488	0	51,507
Ang	1	50%	53,550	26,775	28%	34,272	14,737	0	49,009
Stavri	1	35%	55,650	19,478	28%	24,931	10,720	0	35,652
Salomon	1	40%	52,500	21,000	28%	26,880	11,558	0	38,438
RA 1	0.5	100%	17,000	17,000	28%	21,760	9,357	0	31,117
RA 2	0.5	100%	17,000	17,000	28%	21,760	9,357	0	31,117
Subtotal				160,814		205,842			294,354
UCSF CGL									
Ferrin	1	10%	103,126	10,313	22%	12,530	5,388	0	17,918
Pettersen	1	20%	49,128	9,826	28%	12,528	5,387	0	17,915
Huang	1	50%	68,711	34,356	28%	43,803	18,835	0	62,639
Subtotal						68,861	29,610		98,471
Wash U	57%								
Subtotal									183,768
NCSA									
Subtotal									250,087
AT&T	0%								
Subtotal									129,937
Springer	35%								
Subtotal	1								156,462
UCSF Personnel Subtotal									392,825
Subcontractor personnel subtotal									540,191
Equipment				Item cost	Qty	Cost	Tax		
Collaborator workstations				3,087	15	46,305	3,936		50,241
SGI Onyx w/ RE2				74,792	1	74,792	6,357		81,149
Alpha (cluster)				4,630	8	37,040	3,148		40,188
Maintenance									48,042
Supplies									

	MO disks				115	10	1,150	86		1,236
Travel										
	UCSF/NY, NJ round trip				850	15	12,750	956		13,706
	UCSF/St Louis round trip				650	9	5,850	439		6,289
	UCSF/Urbana round trip				650	6	3,900	293		4,193
Other										
	Subtotal						181,787	15,215		245,044
Year 2 TOTALS										1,178,059
=====										
Year 3										
					Salary		Total		Contract	Grant
Personnel	Appt	% Effort	Base	Requested	Fringe	Directs	Indirects	DOI	Requested	
UCSF CKM	43%									
Doyle	1	30%	109,975	32,993	28%	42,230	18,159	0	60,389	
Martin	1	40%	73,867	29,547	28%	37,820	16,263	0	54,082	
Ang	1	50%	56,228	28,114	28%	35,986	15,474	0	51,460	
Stavri	1	35%	58,432	20,451	28%	26,178	11,256	0	37,434	
Salomon	1	40%	55,125	22,050	28%	28,224	12,136	0	40,360	
RA 1	0.5	100%	19,000	19,000	28%	24,320	10,458	0	34,778	
RA 2	0.5	100%	19,000	19,000	28%	24,320	10,458	0	34,778	
Subtotal				171,155		219,078				313,281
UCSF CGL										
Ferrin	1	10%	112,902	11,290	22%	13,718	5,899	0	19,616	
Petersen	1	20%	52,567	10,513	28%	13,405	5,764	0	19,169	
Huang	1	50%	73,521	36,761	28%	46,870	20,154	0	67,024	
Subtotal						73,992	31,816			105,808
Wash U										
Subtotal										192,214
NCSA										
Subtotal										89,788
AT&T										
Subtotal										136,434
Springer										
Subtotal		1								158,432
UCSF Personnel Subtotal										419,089
Subcontractor personnel subtotal										432,651
Equipment										
				Item cost	Qty	Cost	Tax			
Collaborator workstations				3,087	15	46,305	3,936			50,241

SGI Onyx w/ RE2				74,792	1	74,792	6,357		81,149
MPEG terminals				4,000	15	60,000	5,100		65,100
Alpha (cluster)				4,630	8	37,040	3,148		40,188
Maintenance									66,270
Supplies									
MO disks				115	10	1,150	86		1,236
Travel									
UCSF/NY,NJ round trip				850	15	12,750	956		13,706
UCSF/St Louis round trip				650	9	5,850	439		6,289
UCSF/Urbana round trip				650	6	3,900	293		4,193
Other									
Subtotal						241,787	20,315		328,372
Year 3 TOTALS									1,180,113
=====									
Year 4									
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total Directs	Indirects	Contract DOI	Grant Requested
UCSF CKM	43%								
Doyle	1	50%	115,474	57,737	28%	73,903	31,778	0	105,682
Martin	1	40%	77,560	31,024	28%	39,711	17,076	0	56,786
Ang	1	50%	59,039	29,520	28%	37,785	16,248	0	54,032
Stavri	1	50%	61,354	30,677	28%	39,267	16,885	0	56,151
Salomon	1	40%	57,881	23,152	28%	29,635	12,743	0	42,378
RA 1	0.5	100%	21,000	21,000	28%	26,880	11,558	0	38,438
RA 2	0.5	100%	21,000	21,000	28%	26,880	11,558	0	38,438
Subtotal				214,110		274,061			391,907
UCSF CGL									
Ferrin	1	10%	112,902	11,290	22%	13,718	5,899	0	19,616
Petersen	1	20%	56,247	11,249	28%	14,343	6,167	0	20,510
Huang	1	50%	78,667	39,334	28%	50,150	21,565	0	71,715
Subtotal						78,211	33,631		111,841
Wash U	57%								
Subtotal									201,047
NCSA									
Subtotal									94,277
AT&T	0%								
Subtotal									143,256
Springer	35%								
Subtotal	1								160,531

UCSF Personnel Subtotal							503,748
Subcontractor personnel subtotal							449,333
Equipment				Item cost	Qty	Cost	Tax
Collaborator workstations				3,087	15	46,305	3,936
SGI Onyx w/ RE2				74,792	1	74,792	6,357
Alpha (cluster)				4,630	8	37,040	3,148
Maintenance							
							48,042
Supplies							
MO disks				115	10	1,150	86
							1,236
Travel							
UCSF/NY,NJ round trip				850	15	12,750	956
UCSF/St Louis round trip				650	9	5,850	439
UCSF/Urbana round trip				650	6	3,900	293
							4,193
Other							
Subtotal						181,787	15,215
							245,044
Year 4 TOTALS							1,198,126

Summary										
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total		Contract	Grant	
						Directs	Indirects	DOI	Requested	
UCSF CKM	43%									
Subtotal										
UCSF CGL										
Subtotal										
Wash U	57%									
Subtotal										
NCSA										
Subtotal										
AT&T	0%									
Subtotal										
Springer	35%									
Subtotal		1								
UCSF Personnel Subtotal										
Subcontractor personnel subtotal										
Equipment										
Supplies										
Travel										
Other										
Subtotal										

4-Year TOTALS										4,755,873
<hr/> <hr/>										

Digital Libraries Grant Budget										
Year 1										
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total		Contract	Grant	
						Directs	Indirects	DOI	Requested	
UCSF CKM	43%									
Doyle	1	30%	99,750	29,925	28%	38,304	18,471	0	54,775	
Martin	1	40%	67,000	28,800	28%	34,304	14,751	0	49,055	
Ang	1	50%	51,000	25,500	28%	32,640	14,035	0	46,675	
Stavri	1	35%	53,000	18,550	28%	23,744	10,210	0	33,954	
Salomon	1	40%	50,000	20,000	28%	25,600	11,008	0	36,608	
RA 1	0.5	100%	15,000	15,000	28%	19,200	8,256	0	27,456	
RA 2	0.5	100%	15,000	15,000	28%	19,200	8,256	0	27,456	
Subtotal				166,775		192,992			276,978	
UCSF CGL										
Ferrin	1	10%	103,126	10,313	22%	12,530	5,388	0	17,918	
Petersen	1	20%	45,914	9,183	28%	11,708	5,034	0	16,743	
Huang	1	50%	64,216	32,108	28%	40,938	17,603	0	58,541	
Subtotal						60,900	28,000		88,900	
Wash U	57%									
Subtotal									176,796	
NCSA										
Subtotal									268,176	
AT&T	0%									
Subtotal									123,760	
Springer	35%									
Subtotal		1							166,769	
UCSF Personnel Subtotal									361,779	
Subcontractor personnel subtotal									842,649	
Equipment				Item cost	Qty	Cost	Tax			
Collaborator workstations				3,087	15	46,305	3,936		50,241	
SGI Onyx w/ RE2				74,792	1	74,792	6,357		81,149	
Alpha (cluster)				4,630	8	37,040	3,148		40,188	
Maintenance									48,042	
Supplies										
Galaxy				4,000	4	16,000	1,200		17,200	
Avalanche				9,000	2	18,000	1,350		19,350	
SoftQuad A/E				900	3	2,700	203		2,903	
App Builder				2,500	4	10,000	750		10,750	
MO disks				115	10	1,150	88		1,238	
Travel										
UCSF/NY,NJ round trip				850	15	12,750	956		13,706	

UCSF/St Louis round trip				650	9	5,850	439		6,289
UCSF/Urbana round trip				650	6	3,900	293		4,193
Other									
Subtotal						228,487	18,718		295,247
Year 1 TOTALS									1,199,575
=====									
Year 2									
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total Directs	Indirects	Contract DOI	Grant Requested
UCSF CKM	43%								
Doyle	1	30%	104,738	31,421	28%	40,219	17,294	0	57,514
Martin	1	40%	70,350	28,140	28%	36,019	15,488	0	51,507
Ang	1	50%	53,550	28,775	28%	34,272	14,737	0	49,009
Stavri	1	35%	55,650	19,478	28%	24,931	10,720	0	35,652
Salomon	1	40%	52,500	21,000	28%	28,880	11,558	0	38,438
RA 1	0.5	100%	17,000	17,000	28%	21,760	9,357	0	31,117
RA 2	0.5	100%	17,000	17,000	28%	21,760	9,357	0	31,117
Subtotal				180,814		295,842			284,354
UCSF CGL									
Ferrin	1	10%	103,128	10,313	22%	12,530	5,388	0	17,918
Petersen	1	20%	49,128	9,826	28%	12,528	5,387	0	17,915
Huang	1	50%	68,711	34,358	28%	43,803	18,835	0	62,639
Subtotal						68,861	29,610		98,471
Wash U	57%								
Subtotal									183,768
NCSA									
Subtotal									250,087
AT&T	0%								
Subtotal									128,937
Springer	35%								
Subtotal									168,462
UCSF Personnel Subtotal									392,826
Subcontractor personnel subtotal									540,181
Equipment				Item cost	Qty	Cost	Tax		
Collaborator workstations				3,087	15	46,305	3,936		50,241
SGI Onyx w/ RE2				74,792	1	74,792	6,357		81,149
Alpha (cluster)				4,630	8	37,040	3,148		40,188
Maintenance									48,042
Supplies									

	MO disks				115	10	1,150	86		1,236
Travel										
	UCSF/NY,NJ round trip				850	15	12,750	956		13,706
	UCSF/St Louis round trip				650	9	5,850	439		6,289
	UCSF/Urbana round trip				650	6	3,900	293		4,193
Other										
	Subtotal						181,787	15,216		246,044
Year 2 TOTALS										1,178,059
=====										
Year 3										
Personnel		Appt	% Effort	Base	Salary Requested	Fringe	Total Directs	Indirects	Contract DOI	Grant Requested
UCSF CKM		43%								
	Doyle	1	30%	109,975	32,993	28%	42,230	18,159	0	60,389
	Martin	1	40%	73,867	29,547	28%	37,820	16,263	0	54,082
	Ang	1	50%	56,228	28,114	28%	35,968	15,474	0	51,460
	Stavri	1	35%	58,432	20,451	28%	26,178	11,256	0	37,434
	Salomon	1	40%	55,125	22,050	28%	28,224	12,136	0	40,360
	RA 1	0.5	100%	19,000	19,000	28%	24,320	10,458	0	34,778
	RA 2	0.5	100%	19,000	19,000	28%	24,320	10,458	0	34,778
	Subtotal				171,165		219,078			313,281
UCSF CGL										
	Ferrin	1	10%	112,902	11,290	22%	13,718	5,899	0	19,616
	Pettersen	1	20%	52,567	10,513	28%	13,405	5,784	0	19,189
	Huang	1	50%	73,521	36,761	28%	46,870	20,154	0	67,024
	Subtotal						73,992	31,818		106,808
Wash U		57%								
	Subtotal									192,214
NCSA										
	Subtotal									89,788
AT&T		0%								
	Subtotal									136,434
Springer		35%								
	Subtotal		1							168,432
UCSF Personnel Subtotal										419,089
Subcontractor personnel subtotal										432,651
Equipment					Item cost	Qty	Cost	Tax		
	Collaborator workstations				3,087	15	46,305	3,936		50,241

SGI Ornyx w/ RE2				74,792	1	74,792	6,357			81,149
MPEG terminals				4,000	15	60,000	5,100			65,100
Alpha (cluster)				4,630	8	37,040	3,148			40,188
Maintenance										66,270
Supplies										
MO disks				115	10	1,150	86			1,236
Travel										
UCSF/NY,NJ round trip				850	15	12,750	956			13,706
UCSF/St Louis round trip				650	9	5,850	439			6,289
UCSF/Urbana round trip				650	6	3,900	293			4,193
Other										
Subtotal						241,787	20,316			328,372
Year 3 TOTALS										1,180,113

Year 4										
Personnel						Salary	Total	Contract	Grant	
		Appt	% Effort	Base	Requested	Fringe	Directs	Indirects	DOI	Requested
UCSF CKM		43%								
Doyle	1	50%	115,474	57,737	28%	73,903	31,778	0		105,682
Martin	1	40%	77,560	31,024	28%	39,711	17,076	0		56,786
Ang	1	50%	69,039	29,520	28%	37,785	16,248	0		54,032
Stavri	1	50%	61,354	30,677	28%	39,267	16,885	0		56,151
Salomon	1	40%	57,881	23,152	28%	29,635	12,743	0		42,378
RA 1	0.5	100%	21,000	21,000	28%	26,880	11,558	0		38,438
RA 2	0.5	100%	21,000	21,000	28%	26,880	11,558	0		38,438
Subtotal				214,110		274,061				391,907
UCSF CGL										
Ferrin	1	10%	112,902	11,290	22%	13,718	5,899	0		19,818
Petersen	1	20%	58,247	11,249	28%	14,343	6,167	0		20,510
Huang	1	50%	78,667	39,334	28%	50,150	21,565	0		71,715
Subtotal						78,211	33,631			111,841
Wash U		57%								
Subtotal										201,047
NCSA										
Subtotal										94,277
AT&T		0%								
Subtotal										143,266
Springer		35%								
Subtotal		1								160,631

UCSF Personnel Subtotal							503,748
Subcontractor personnel subtotal							448,333
Equipment			Item cost	Qty	Cost	Tax	
Collaborator workstations			3,087	15	48,305	3,936	50,241
SGI Onyx w/ RE2			74,792	1	74,792	6,357	81,149
Alpha (cluster)			4,630	8	37,040	3,148	40,188
Maintenance							48,042
Supplies							
MO disks			115	10	1,150	86	1,236
Travel							
UCSF/NY,NJ round trip			850	15	12,750	956	13,706
UCSF/St Louis round trip			650	9	5,850	439	6,289
UCSF/Urbana round trip			650	6	3,900	293	4,193
Other							
Subtotal					181,787	15,216	246,044
Year 4 TOTALS							1,198,126

=====

Summary									
Personnel	Appt	% Effort	Base	Salary Requested	Fringe	Total		Contract	Grant
						Directs	Indirects	DOI	Requested
UCSF CKM	43%								
Subtotal									
UCSF CGL									
Subtotal									
Wash U	57%								
Subtotal									
NCSA									
Subtotal									
AT&T	0%								
Subtotal									
Springer	35%								
Subtotal		1							
UCSF Personnel Subtotal									
Subcontractor personnel subtotal									
Equipment									
Supplies									
Travel									
Other									
Subtotal									

4-Year TOTALS										4,755,873
=====										