Early Experience with a Repository for Patterned Injury Data

David Stotts, John Smith, Kevin Jeffay,
Prasun Dewan, Dana Smith
Computer Science Department
University of North Carolina
Chapel Hill, NC 27599-3175
`colab-fac@cs.unc.edu`

William Oliver
Armed Forces Institute of Pathology
Walter Reed Army Hospital
Washington DC
`oliver@cs.unc.edu`

ABSTRACT

We have constructed a proof-of-principle system called the Repository for Patterned Injury Data (RPID) for supporting collaborative forensic medicine. The early RPID prototype is built on ABC/DGS, a graph-server and collaborative hypermedia system built in the UNC Collaboratory. ABC provides collaboration services for work groups via shared artifacts, giving common views of the information and allowing conferencing over the data. A second prototype is underway that has more flexible control of multi-person creation of, and access to, the shared patient data and pathology artifacts. We conclude by describing a planned third prototype, to be built not on ABC, but on a modification of the WWW *httpd* distributed data server.

KEYWORDS: forensic pathology, ABC, DGS, common view, shared artifact, access control, group access, World Wide Web, httpd, distributed data server

1 Overview

Forensic specialists have long understood the importance of toolmarks and trace evidence in the investigation of violent crimes and in successfully prosecuting those who commit these acts. However, except in specific areas such as forensic odontology and tire and shoeprint impression analysis, there is little formal work in the area of patterned injury analysis as a problem in forensic pathology. A better ability to use patterned injuries to determine what object or objects were used to commit violence, matching a possible weapon or object to a wound with some degree of certainty, or of matching a mark on a body with an object at a scene would be a tremendous boon.

We believe there are three primary reasons why large-scale formal work has not been done. The first is that developing expertise in the area is profoundly experiential. The universe of possible objects and the variety of wounds that any particular object can cause is large, and experience is gained slowly. Second, there is no central collection of patterned injuries. A centralized collection of solved cases provides a uniform teaching base, a standard by which other cases and other approaches to cases can be evaluated, and a resource for approaching unsolved cases. Finally, it is difficult for pathologists to consult on a large-scale basis about patterned injuries.

The usefulness of multi-person interactive consultation about images has been demonstrated in radiology and other medical disciplines, and is a driving force behind the imaging workstation systems being implemented by the FBI.

The Repository for Patterned Injury Data (RPID) project is addressing each of these problems. First, we are building a digital library for forensic medicine, an electronic information repository to support collaboration on medical cases involving patterned injuries. The library is to contain traditional data forms (text, images) as well as newer multi-media data forms (audio, video, handwriting). Second, we are assembling a computer and communications infrastructure to provide access to the library for real-time interactive consultation and database exploration. Third, we will carry out a continuous series of evaluative and experimental studies to validate the designs and guide further system development.

The design of the RPID was reported in the first Digital Libraries conference.[SSDJ94] For clarity here, we briefly summarize the structure and goals of the project. Following this recap, we illustrate the current prototypes and discuss the interactions we have established for consulting pathologists. We conclude with our plans for subsequent versions of the RPID.

2 Recap of Project Goals and Design

Prevention of violent crime continues to be an important national priority, and it is increasingly important in the daily lives of our citizenry. Those actively involved in investigating these crimes and in apprehending the people who commit them must be able to pool their knowledge and expertise to provide maximum effectiveness. The Repository for Patterned Injury Data (RPID) we are creating, and our research with it, will enable more effective interaction of forensic pathologists. Specifically, it will enable forensic pathologists to consult a large testbed of patterned injury data, share new case data, apply enhancement and analysis algorithms to images, and consult remotely with one another.

The analysis of patterned injury in forensic pathology presents a challenge that draws uniquely upon both the medical and forensic expertise of the investigator. As a problem in wounding, it is a medical challenge. As a problem in image enhancement and pattern analysis, it is a forensic challenge. This skill benefits heavily from experience. The investigator must have both experience in wound analysis and a knowledge of the universe of discourse, wherein lies the object which caused the injury.

For instance, discerning the general properties of an object (that it is rounded, sharp- edged, etc.) is well within the general knowledge base of most forensic pathologists. Recognizing that a specific injury is likely to have been caused by, say, a socket wrench requires the examiner not only to know about pathology but also to have some knowledge of automotive tools. This need for an encyclopedic recall of objects and object properties makes much patterned injury analysis difficult, especially when injuries occur in areas which abound with specialized tools, implements, and objects (e.g., construction sites, factories, hair-dressing salons, etc.). Moreover, recognizing the types of marks made by partial or oblique blows is a further challenge in analyzing the geometry of impressions.

No forensic pathologist or odontologist can be an expert in all areas of hardware manufacture and utilization. Instead, we must rely on the experience of our colleagues. Unfortunately, when an expert in the field retires, a wealth of specialized experience is lost, usually along with a career's worth of valuable patterned injury data and analysis results. The RPID project is an attempt to save such knowledge and to make such experience permanently available to pathologists and investigators across the country. We are doing this by establishing an electronic registry of solved patterned injuries, and by developing a computing and communications infrastructure wherein pathologists and investigators can electronically access the data, search for cases germane to their current problems, and consult with one another.

Key capabilities of the RPID include:

- multimedia database of valuable forensic medicine collections
- users can generate, add new data
- image display, analysis, and modification
- hypermedia interlinking of multiple images and text
- video teleconferencing, integrated with collaborative data viewing
- automatic data collection for system evaluation and user studies

Key benefits of RPID to law enforcement and forensic personnel include:

- a distributed, but integrated repository of pattern injury data that can be used by forensic pathologists country-wide, even world-wide
- collaborative consultation of data, bringing broad experience to bear in solving cases
- ability to construct arguments based on case data for presentations in court and other appropriate settings
- capturing expert knowledge and experience for future use, helping stem the loss of expertise when pathologists retire

3 Overview of Collaboration Systems and ABC

Our first RPID prototype has been built using a software system developed by the UNC Collaboratory Project over the past few years. Called the Artifact-Based Collaboration (ABC) system, it supports development of work artifacts by groups engaged in both synchronous (working at the same time) and asynchronous (non-real-time, like email) work.[SS91] It also includes tools for creating and using ephemeral products and for developing shared intangible knowledge.

Figure 1 gives a schematic overview of the architecture of ABC. The system is designed to be used by distributed groups whose members may be located a considerable distances from one another, clustered at a single site, or combinations of clusters and scattered individuals. Thus, a key assumption is that members do much of their work through workstations connected to a high-speed network. This network connects each workstation with a hypermedia data storage facility as well as with other workstations. Users work with browsers that enable them to see and manipulate structural data and with applications that enable them to work with traditional content data, such as text and diagrams. A conferencing component allows several members to share any browser or application so that all can see the same information and each, in turn, can edit or manipulate the data. The design also includes audio and video communication so that members can talk to and see one another, both to supplement computer conferences and for conventional conversations. Thus, ABC integrates support for three major forms of collaborative work — individual work on the artifact, collective work on the artifact, and conversations/discussion, — thereby supporting both synchronous and asynchronous activities. To understand ABC more clearly, one must understand five key concepts or system features: virtual screens, hypermedia data store, browsers/applications, conferencing, and audio/video communication. We discuss each of these in turn.

Figure 1: Artifact-Based Collaboration (ABC) System. Three workstations are connected to a hypermedia storage system and to one another by a high-speed network. The same conferenced browser can be seen on each workstation, with supporting video windows. Other nonconferenced browsers and applications can also be seen on each workstation.

Virtual screen

ABC runs within the X Window System under the UNIX operating system. However, the data users work with is stored in the hypermedia data storage system, described later, rather than the UNIX file system. This requires users of ABC to have a different mental model of their data and to use different tools to work with it. In addition, the system provides several kinds of generic function, such as conferencing and hyperlinking, that apply to any program or application running within it. Consequently, it is important for users to be aware of whether they are working within UNIX or within ABC at any given moment.

To do this, ABC provides a facility called a virtual screen.[JLMS92] It is a window with an identifying label. However, it can include other windows, and it can be expanded to cover the entire screen. All ABC programs run within this virtual screen, and any program that runs within it references the hypermedia data store, rather than the UNIX file system. Access to ABC generic functions is provided by an additional menu bar attached to the top of each program window within the ABC virtual screen. Thus, ABC can be viewed as an environment within the larger UNIX context, or it can be expanded to appear to be the entire system.

Figure 2: Artifact for a collaborative project. Higher level nodes contain graphs, whereas nodes in lower level graphs contain file data, (in this case, text). Hyperlinks connect a node in one graph with nodes in another, one of which is anchored within the node's content.

Hypermedia data store

ABC encourages a group to think of the entire collection of information it builds and works with as a single, integrated structure, rather than as a set of separate and independent files.[SSS93] This structure constitutes the group's artifact. For a software development project, the artifact might include early concept papers, requirements, specifications, the architecture, source code, maintenance manuals, user documentation, and perhaps conference papers and journal articles that describe the system. It could also include the private or personal data of the individual members of the team.

In terms of the ABC data model, the artifact consists of a collection of separate graphs that are composed to form a single large, interrelated structure. Currently, ABC supports three types of graphs: trees, networks, and lists. Each separate graph corresponds to some presumably logical entity, such as a short document or a section or chapter of a larger document. Graphs, in turn, consist of sets of nodes and links. Nodes normally represent concepts, whereas links represent relationships between concepts.

Some graph types are better suited for particular tasks than others. For example, one could use a tree to represent the hierarchical structure of a document, similar to an outline. The title would appear as the label of the node at the top, nodes for major sections under that, subsection nodes below each section, and so forth. However, because tree graphs permit only links from a parent node to its child nodes, they do not allow links between sibling nodes or links that would otherwise cross the hierarchy. If more flexibility is needed, for example, to record ideas and relationships generated during a brainstorming session, one can use a network graph, which permits links between any pair of nodes.

An important concept in ABC is *node content*. Although nodes can be used to represent concepts directly, they can also contain two types of information. First, nodes can contain a block of data similar to a conventional file. For a short document represented as a tree, the actual text, diagrams, or other forms of information that constitute its substantive content can be stored as the contents of the RleafS nodes at the bottom of a tree. A printed version of the document can, then, be derived by having the system go around the tree and gather up all of the pieces stored as node contents and send them to a printer. Second, nodes can contain graphs. That is, separate, disjoint graphs can be stored as the contents of nodes, just like the blocks of data described previously. For example, a group could collectively develop the overall structure of a document as a two-level overview tree that identifies the document as a whole and the chapters to be included in it. They could then work individually or in teams on the detailed plans for each chapter as separate graphs that are contained within the chapter nodes. By successively deeper nesting of graphs, groups can build structures that are very large, but ones that can be viewed at different levels of detail, making them easier to understand and easier to work with. Figure 2 illustrates how a group might organize all of its information as a single artifact.

Finally, the ABC data model includes a second type of link, called a *hyperlink*. The links described to this point are structural links. They are constrained by the rules that apply to a given graph type. For example, a node in a tree graph can have (at most) one incoming link, but any number of outgoing links. This constraint, although helpful in some contexts, precludes relationships that are also desirable, such as cross references, that would violate the basic tree structure. Hyperlinks provide this flexibility without violating the integrity of the graph type.

Hyperlinks can define two types of relationships that cannot be defined using structural links. First, they can join nodes (or points within the contents of nodes) to one another within a graph that would violate the type of the graph if the relationship were defined using a structural link. Second, hyperlinks can join nodes in one graph to nodes in another graph. Because a structural link must exist within some particular graph, they cannot be used for this purpose. This second use of hyperlinks is especially important for collaborative work. Maintaining the internal consistency of the artifact becomes a major problem for large projects. Hyperlinks provide a mechanism through which groups can identify dependencies within the artifact so that when a change is made in one place — for example, a specification document — one can follow hyperlinks to other places within the artifact — for example, the source code and user documentation — that may be affected by the change and, thus, require update or verification. Figure 2 shows both types of hyperlinks.

Browsers and applications

Members of a collaborative group work with ABC through two types of tools: browsers and applications. Browsers enable users to view and work with the structure of the artifact. Through them, users create new graphs as the contents of nodes; add, delete, and move the nodes within a graph; create and edit links between nodes; add or change identifying labels on nodes; or simply view an existing graph. Currently, ABC includes browsers for trees, networks, and lists, corresponding to the three types of graphs supported by the data model. Other browsers could be developed, such as a decomposition diagram editor, whose underlying data model is a graph but whose appearance differs from conventional graph representations.

Applications are used to work with blocks of nongraph data. Examples include text editors, drawing programs, spreadsheets, CAD/CAM tools, and so forth. ABC provides an open architecture with respect to applications, so long as the application operates within the X/UNIX environment. Thus, a group can use ABC browsers for working with the structure of the artifact, but the tools it is accustomed to using for working with the individual files/blocks of data contained within the graph structure.

Conferencing

ABC supports shared X conferencing as one of the generic functions it adds to any browser or application running within it. Conferencing is accessed through the second title bar ABC adds to a virtual screen and to all browsers and applications that appear within it .[JLMS92] This allows members of a distributed group to work together at the same time on a document, drawing, or other form of information by sharing any browser and application running in the ABC environment. They all see the same display, and they can take turns providing input to the conferenced program.

One unusual feature of ABC conferencing is that it permits multiple browsers/applications. to be shared within a single conference. This can be done by conferencing a virtual screen and all browsers and applications running within it, as opposed to a single program. Thus, for example, if a question comes up during a conference that cannot be answered by reference to currently shared data, a member can start a browser or application on other data and then bring the second program into the conference.

By integrating conferencing into the collaborative environment, ABC enables members of a group to shift easily and smoothly between individual, asynchronous work to synchronous, collective work. And, of course, there is no issue of moving data from one context to the other, because the group works with portions of the artifact in both situations.

Audio and video

Although computer conferencing can help distributed groups work together on the artifact in ways that would not be possible otherwise, members also need to talk with one another, hold meetings, and carry on both formal and informal discussions. No technology can replace face-to-face encounters. But it may be able to supplement those encounters, particularly for groups that are widely distributed and for whom frequent face-to-face interaction is not possible.

ABC includes audio and video communication within its basic design. Currently, this function is implemented outside the ABC environment proper, using the telephone system for audio and cable television technology for video. In the next stage of our work, we plan to fully integrate audio and video communication into ABC. To do this, we will build on work done at UNC by Jeffay.[JSS92] Using specialized scheduling algorithms developed by his multimedia research group and conventional data compression hardware, Jeffay's system can deliver full-motion video and high-quality audio over conventional packet-switched data networks. We plan to incorporate this type of digital audio and video as a generic function, analogous to conferencing, that can be accessed from within ABC and controlled by ABC applications. Thus, users will be able to schedule and setup audio/video conversations analogous to starting a computer conference, enabling them to make smooth transitions from individual work, to conversations with colleagues, to video supplemented computer conferences.

Figure 3: Screen from RPID prototype; shown are image list for a case, autopsy report with links to images, and links within an image to other information.

4 First RPID Prototype

We have assembled an initial collection of forensic medicine cases for early experiments with RPID. Data for each case includes photos of the scene surrounding the events, photos of the person involved, text reports from the attending physicians, text reports from law enforcement personnel, traditional database information entered by the pathologists involved (victim age, sex, name, event classification, etc.), and computer-generated graphics resulting from analysis of case data.

The data for the repository is coming from the extensive archives of the Armed Forces Institute of Pathology (AFIP); from the Office of the Armed Forces Medical Examiner's (OAFME) Lindenberg collection of microscope slides and pictorial data; and from the Milton Helpern Forensic Pathology Museum. Other specialized collections, such as images of tire tread marks, muzzle imprints, and others, are maintained by individual pathologists and researchers; we intend to bring these data into the repository, once it has been established. Investigators who have expressed interest in submitting their data into the collection include Dr. Marcella Fierro, who has an extensive collection of muzzle imprints and soot deposition patterns, Dr. Richard Froede, former Armed Forces Medical Examiner, and Dr. Homer Campbell of the New Mexico Office of the Medical Investigator. The North Carolina

Chief Medical Examiner's Office is also contributing 15 years of state autopsy records.

We have encoded this information in the ABC collaboration system[JLMS92] and have adapted several of the ABC browsers to allow rendering and annotation of the images. Figure 3 shows an ABC screen of one case from RPID. In this view, autopsy photos are collected and links among them have been created to compose related views. We have also linked in the text report from the autopsy, with the link anchor connecting specific regions in the photos with specific paragraphs in the report. In the photo, link anchors show up as rectangles in contrasting colors.

In this prototype, we depend on non-integrated collaboration support systems for information sharing and group interactions. In the Colab we support XTV shared X window system.[AWGN88,Abe90] With XTV, the collaborators can view and analyze common images, with all collaborators seeing the same output from a single tool process. When reports are viewed, all see the same text. When a link in an image is selected, the destination image (or report) is displayed on all workstations. The common viewspace is achieved by multiplexing the X server protocol streams from the individual workstations participating in the shared X sessions.

5 Plans for Subsequent Prototypes

Our initial prototype has been built on the Distributed Graph Server (DGS) in UNC's Colab, using the ABC collaborative hypermedia system for organizing the data and accessing the DGS. For subsequent prototypes we plan to work with a scaled-down system designed around an augmentation of the WWW distributed data server *httpd*. We expect that this design approach will give great leverage, since the *httpd* is well-tested, in wide-spread use, and gives off-the-shelf solutions to several problems such as user authentication, data encryption, and compatibility with other systems.

5.1 Increased flexibility in sharing data spaces

An important abstraction in a digital library is the notion of a shared workspace. A shared workspace is a window on the screen that gets updated in response to actions of other users. Several projects, both within UNC and elsewhere, are actively researching issues in the design and implementation of shared workspaces.

In the initial RPID prototype, sharing is done by multiplexing X protocol streams, giving multiple users identical screen views. We are developing a second prototype that encodes more specific, and more flexible, sharing semantics based on abstract data objects defined in the implementation code. The basic image database remains the same; for interaction control we are using the Suite multi-user interface building system.[DC92]

In general, the RPID library helps pathologists resolve cases by producing autopsy reports on them. These reports are represented internally as linked data, amenable to printing or to browsing as hypermedia documents. Using Suite, we create roles for the people involved in creating and using data in each case, and we assign access privileges based on roles. Suite manages these administration of these privileges as well as basic concurrency control on the database.

5.2 Use of the WWW server for distributed data exchange

The ABC system supporting the RPID is built on a distributed data server called DGS,[SSS93] a research project in the UNC Colab. While a full implementation of the database within the ABC/DGS structure has many advantages, the unstructured flat graph paradigm of the World Wide Web has grown rapidly in the past two years to become the *de facto* standard of browsable and sharable information systems. The implementation of our system within the WWW would allow greater use of existing implementations and the concomitant savings of time and resources. The wide availability of WWW implementations across systems would allow dissemination of the product to users with a wider range of infrastructures. There are several major server modifications to make, as well as several new applications required to interact with the server to provide the collaboration functions characteristic of the RPID.

Using *httpd* we can get a full-scale RPID operating sooner, with more reliability and less expense. However, we will have some changes to make in the server, resulting in a specialized version. For example, we will need to add the ability for the server to manage hyperlinks that have multiple sources and multiple destinations; this structure has proven useful in other projects[FS94] for representing collaboration protocols in hypermedia. The current *httpd* server does not support such links. Our modification will be upwardly compatible with existing *httpd* versions and with existing WWW data.

Another alteration we must design and implement is a pseudoserver to sit between the interface clients and the server. The pseudoserver will capture traffic between clients and the server to broadcast actions to groups of collaborating users. When one user is browsing data, the collaborators will see the same image. The pseudoserver will also be a common collection point for capture of user/system interactions (called protocols in the evaluation section below). We must collect these activity traces to perform valid assessments of system performance and utility. The pseudoserver technique has been successfully used in other collaboration systems architectures.[SSS93,Men93,AWF91]

5.3 Custom-built application software

Though we plan to use a modified *httpd* distributed data server, current WWW interface clients do not provide all of the functionality we need for the data sharing and collaborative manipulation aspects of the RPID. Thus, we have several custom-designed applications systems to construct to interact with the data server.

To support browsing, objects will be linked with one another along a variety of dimensions. For example, all of the data associated with a given case could be linked into a tree that included as one branch the images, as another branch the various reports associated with the case, and, as a third branch, video clips of the crime scene. However, images in one case could also be linked to similar or related images in other cases according to specific features, such as length of wound, depth, or shape. Thus, users will be able to browse within the material from a given case but also across the primary structure of the library to data associated with other cases. The system will record a trace of users' browsing paths to facilitate subsequent analysis, as a learning aid, and as a means of capturing one form of expert knowledge. We will have to construct Web clients that give better support for hierarchy and link typing than the current flat HTML standard allows.

To support search, data will be characterized in terms of specific features and parameters on those features when they are entered in the repository. Features will be used to generate links in the hypermedia graph structure. They will also be stored in a conventional information retrieval system. Consequently, users will be able to submit queries to the retrieval system, obtain a set of objects, which they may then view or otherwise access through the hypermedia system. Once a user has arrived at a given object, he or she will be able to branch out using hypermedia browsing facilities to other objects linked to it. Thus, the system will combine capabilities of both hypermedia browsing and information retrieval.

The system will be extensible so that additional applications can be included in the environment. This will enable users to invoke specialized tools – such as image enhancement programs. Once such applications have been run, their output can also be stored in the repository for future use. Thus, the system will provide a flexible, easy to use environment for exploratory data analysis. The system must also include provision for audio and video conversations and discussions. Initially, teleconferencing will be supplied by enhanced telephone lines and specialized hardware/software configurations incorporated into users' workstations. As digital technologies mature, we hope to include teleconferencing through the computer network.

Since consultation is so important and since it is so hard to get people together at the same time, particularly if their skills are in high demand, the system will also include facilities to enable a user to organize bodies of material, including his or her recorded statements, that can later be viewed and responded to by another, consulting user. Thus, collaborators will be able to carry on extended, asynchronous "conversations" with regard to specific data without both having to be available at the same time. A similar facility for collecting, recording, and playback will also be useful to forensic pathologists when presenting evidence in the courtroom.

5.4 User Authentication and Data Encryption

Security is a concern for a distributed system containing sensitive and confidential patient medical information. Several companies are producing secure versions of the *httpd* server, and we anticipate basing the RPID on such a secure server. In a secure server, transactions are encrypted in both directions (into and out of the server), and will allow safe transmission of data; if intercepted, the encrypted data will be unintelligible and so useless to any illicit recipient. We will have to acquire the source for a secure server so we can make our collaboration-supporting modifications and user evaluation modifications.

6 Related Work

While there are several projects reported in the literature that provide hypermedia data in a medical context[Fri88,BMJL91,FBDK94] the RPID project has some unique aspects.

The previous efforts have not been in the context of large or widely distributed data sets. They have allowed physicians to simulate in hypermedia their research notebooks, for example; or they have provided centralized facilities for small library subsets.

A unique emphasis of the RPID is enabling of collaboration among users of the library. Another unique aspect of RPID is the research focus on building an infrastructure that will allow the easy integration of new user tools and source data as the library grows. Thus, we are designing for, and investigating the practicality of, continual expansion of the data and increasing distribution among the library sites.

There are also numerous projects that provide image storage and manipulation facilities to medical personnel using images like radiographs and micrographs. These systems tend to focus heavily on the graphics capabilities, and are little more than traditional databases otherwise. A stunning, though currently non-digital, database is the "Visual Diagnosis of Child Abuse" slide set being collected by the National Resource Center on Child Abuse and Neglect. Marcella Fierro has collected an excellent database of muzzle imprints and related patterned injuries. Dr. Fierro's database is unfortunately not digitized; we hope to add it to our database system once a useful system of exploration and consultation is developed. Many departments of Pathology have established World Wide Web (WWW) sites, such as the University of Texas at Houston, University of Washington, University of Alberta, UVA, University of Utah, Fujita Health University, and the Victorian Institute of Forensic Pathology in Australia. These sites have (non-forensic) image galleries available or under construction, but do not represent interactive database efforts. A good listing of currently available links can be found at the Univ. of Alberta site

`http://synapse.uah.ualberta.ca/synapse/000p0025.htm`

The NIJ is also funding a number of literature databases, but these do not involve image analysis directly.

A number of new and important databases and manipulations systems have been developed in non-medical forensics, including the AFIS fingerprint identification system, Drugfire shell casing analysis database, the FBI shoe impression collection, and others. Again, however, there are important differences between these systems and the RPID with respect to interactivity, consultation, and image search methods.

7 REFERENCES

[Abe90] M. Abel. Experiences in exploratory distributed organization. In *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, pages 489–510. Lawrence Erlbaum, 1990.

[AWF91] H. Abdel-Wahab and M. A. Feit. XTV: A framework for sharing X Window clients in remote synchronous collaboration. In *Proc. IEEE Conf. on Communications Software: Communications for Distributed Applications and Systems*, pages 159–167, April 1991.

[AWGN88] H. Abdel-Wahab, S. U. Guan, and J. Nievergelt. Shared workspaces for group collaboration: An experiment using internet and unix interprocess communications. *IEEE Communications Magazine*, 26(11):10–16, November 1988.

[BMJL91] A. M. Burger, B. D. Meyer, C. P. Jung, and K. B. Long. The virtual notebook system. In *Proceedings of ACM Hypertext '91*, pages 395–401. ACM, December 1991.

[DC92] Prasun Dewan and Rajiv Choudhary. A high-level and flexible framework for implementing multi-user user interfaces. *ACM Transactions on Information Systems*, 10(4):345–380, October 1992.

[FBDK94] J. Fowler, D. G. Baker, R. Dargahi, V. Kouramajian, H. Gilson, K. B. Long, C. Petermann, and G. A. Gorry. Experience with the virtual notebook system: Abstraction in hypertext. In *Proc. of the 1994 ACM Conference on Computer Supported Cooperative Work (CSCW '94)*, pages 133–143, October 1994.

[Fri88] Mark E. Frisse. Searching for information in a hypertext medical handbook. *Communications of the ACM*, 31(7):880–886, July 1988.

[FS94] R. Furuta and P. D. Stotts. Interpreted collaboration protocols and their use in groupware prototyping. In *Proc. of the 1994 ACM Conference on Computer Supported Cooperative Work (CSCW '94)*, pages 121–131. ACM, New York, October 1994.

[JLMS92] K. Jeffay, J. K. Lin, J. Menges, F. D. Smith, and J. B. Smith. Architecture of the artifact-based collaboration system matrix. In *Proceedings of CSCW '92 (Toronto)*, pages 195–202. ACM Press, 1992.

[JSS92] K. Jeffay, D. Stone, and F. D. Smith. Kernel support for live digital audio and video. *Computer Communications*, 15(6):388–395, 1992.

[Men93] J. Menges. The X engine library: A C++ library for constructing X pseudoservers. In *Proc. of the 7th Annual X Technical Conference*, pages 129–141, 1993.

[SS91] J. B. Smith and F. D. Smith. Abc: A hypermedia system for artifact-based collaboration. In *Proc. of Hypertext '91*, pages 179–192, December 1991.

[SSDJ94] D. Stotts, J. Smith, P. Dewan, K. Jeffay, F. D. Smith, D. K. Smith, S. Weiss, J. Coggins, and Wm. Oliver. A patterened injury digital library for collaborative forensic medicine. In *Proc. of Digital Libraries '94*, pages 25–33, June 1994.

[SSS93] D. E. Shackelford, J. B. Smith, and F. D. Smith. The architecutre and implementation of a distributed hypermedia storage system. In *Proceedings of ACM Hypertext '93*, pages 1–13. ACM, November 1993.