N94-35044

Interactive Archives of Scientific Data

Lloyd A. Treinish
IBM Thomas J. Watson Research Center
Yorktown Heights, NY
lloydt@watson.ibm.com

Abstract

Data generators in many disciplines are rapidly improving, typically much faster than the techniques available to manage and use the data they produce. Appropriate data management techniques coupled with the process or methods of scientific visualization, or at least the technologies that support them show promise in helping to address some of these problems. Consider browsing, for example, in a role for feature identification by the scientist/user to serve as guide in the data selection process. To date, most efforts associated with data browsing have focused on simple images with image data. Unfortunately, these techniques are not applicable to many classes of data or when more than one data set is to be considered. By recalling that browsing is more of a subjective process involving the human visual system and that this is one of the origins of the notion of scientific visualization as a method of computing, then the utilization of visualization strategies for qualitative presentation of data becomes a viable approach. For browsing to be effective it must be interactive with near-real-time system response. With data sets of interesting size, e.g., ≥ O(1 GB), immediate interaction cannot take place on current conventional systems (i.e., high-end graphics workstations). Even though a 1 GB data set is admittedly modest by today's standards, the access and visualization of the entire data set or even a large fraction of it place significant burdens on the floating point and bandwidth capacities of the computer system being employed. This idea can also be extended to environments without high-bandwidth access to an interactive system by considering the distribution of compressed visualizations instead of data for predefined access and browsing scenarios.

Introduction

The instrumentation technology behind data generators in a myriad of disciplines, whether observational or computational, is rapidly improving, especially in the earth and space sciences. The capability to produce data is typically growing much faster than the techniques available to manage and use them. Traditional bulk access to data archives do not scale well to the volume and complexity of current or planned data streams nor the demanding application of such data, such as analysis and modelling. A fundamental change in the access of archives of these data from static, batch systems to dynamic, interactive systems is required. As a first step in enabling this paradigm shift, consider appropriate data management techniques coupled with the process or methods of scientific visualization. The technologies that support visualization show promise in helping to address some of these problems.

Data management

Data management relevant to the access and utilization of large scientific data sets (e.g., locating, retrieving and using data of interest) may be classified into four levels:

I. At the back-end are warehousing issues related to problems of media, bandwidth, protocols, formats.

II. Above that are data models, access techniques, data base systems, query languages and programming interfaces.

III. As a medium between the access of data and the user, consider browsing as an aid for feature identification, which serves as a guide in the data selection process.

IV. At the front-end are human factors issues (e.g., system ergonomics, human perception) and the incorporation of domain-specific knowledge to permit a system to be usable and useful (e.g., domain-driven task-based interfaces and tools).

At level I, high-speed and high-capacity storage and networking are either available now or will be common in the near future. A chief concern is not the volume of data nor the signaling speeds of these devices, but the effective capacity and bandwidth that can be utilized by applications that require warehousing (i.e., above level I). Typical storage and communications protocols are not a match for GB archives, yet current archives are in the TB range,

while those being planned are measured in PB. In addition, accepted data distribution mechanisms such as CD-ROMs hardly scale to data rates planned for projects like NASA's Earth Observing System, which are measured in TB/day (i.e., 1 TB/day is about 2000 CD-ROMs/day).

At level II and to a limited extent level I, consider what is stored and how it is stored. Self-describing physical storage formats and structures should be used. There are many interfaces and structures developed by or used in the scientific community [Treinish, 1992b; Brown et al, 1993], which are driven by application, politics, tradition and requirements. For the kind of data rates that need to be supported for specific computational codes, applications, etc., it is NOT sufficient to provide only bulk access. The accessing software must be able to do so in a meaningful way. This means that the self-describing formats must have associated disk-based structures and software interfaces (e.g., at least abstract data types for programmers and applications, simple utilities for those do not wish to program). They must be transparently accessible at the desk of an end-user. These structures and interfaces must be consistent with the high-speed access to be provided. This must also include task-to-task communication (e.g., transparent workstation access to high-speed storage).

Such requirements demand the utilization of a generalized data model as a mechanism to classify and access data as well as map data to operations. Further, it defines and organizes a set of data *objects* [Treinish, 1992b]. The model (e.g., Haber et al, [1991]) must provide a self-describing

- representation of the physical storage structure (e.g., format)
- structural representation of the data (e.g., data base schema)
- higher-level logical structure

Since the behavior of access to data organized via such a model is in terms of a logical structure, a generalized data model provides a foundation of a data access server as shown in **figure 1**. Otherwise, performance of high-speed devices will not be realized. High-data-rate computations like signal processing, visualization and some classes of modelling also require such support.

Level II also requires enabling tools for data/information systems. Having efficient storage and access are critical for driving applications, but will not be directly useful for helping find data of interest. At the very least, there is a need for low-level metadata management that enables both content and context for the warehousing information. Traditionally, a RDBMS could be used for its implementation. Although, this concept was first prototyped over 10 years ago, a RDBMS cannot handle semantics, spatial information or the bulk [Treinish and Ray, 1985]. The RDBMS would not have any data per se, but would have pointers to the bulk storage enabling simple querying of what is there and where it is stored by characteristics such as spacecraft, instrument, mission, code, date/time, investigator, owner, etc. It would have to supplemented by a non-relational system to adequately support spatial metadata (e.g., Fekete [1990]), as shown in **figure 1.**

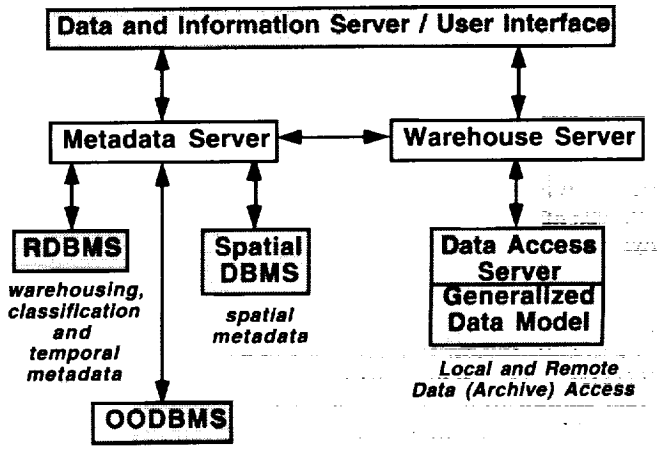

Figure 1. Data management for interactive archives.

There are many challenges in the implementation of an effective data system at level II -- to provide complete access to data of interest simply and easily. A key role for data systems is a means of efficient and intelligent searching and querying for relevant data based upon an assumption that the data volume and complexity is sufficiently large that practical examination of more than a tiny fraction of archive contents is prohibitively expensive. Therefore, the implementation of searches as *meta*-operations on abstractions of the archive (e.g., contextual, domain-driven, spatial, user-driven and visual), though difficult, are highly beneficial. Visual searches or browsing imply the perusal of pictorial representations of data or their abstractions with sufficient content for a user to determine the utility for further, more detailed examination. The advent of practical

methods of scientific visualization show promise in the implementation of visual searches. Hence, consider a very simple notion: *looking* at data of interest in an "appropriate" fashion to determine merit of access for further study. Thus, visualization can be an adjunct to archive management.

History

The idea of visually browsing data is hardly new. Scientists have always used visual abstractions (on paper) of their experimental results for communication. Often these same visualizations en masse were filed and later perused to help recall findings from earlier work. The mechanism for storing, scanning and distribution of such visualizations was the same as for text -- initially shelves and files of paper followed by boxes of microfiche. With the advent of digital data collection instrumentation (e.g., seismic soundings for oil prospecting, study of the earth's atmosphere from spacecraft), this same paradigm was adopted with computer-generated paper- and later microfiche-based visualizations. These visualizations were based upon the graphics technology of the era and the traditions in these fields (e.g., line drawings of iso-contours). Monochromatic microfiche became an effective means of storing visual information compactly that were relatively inexpensive to generate after the cost of the production equipment was amoritized. Further, they were easily distributed, cheap to examine remotely and archiveable. These are important virtues to consider in a modern visualization approach to browsing that goes beyond the relatively limited medium of microfiche.

For example, **figure 2** is a photographic reproduction of a microfiche (approximately three inches x five inches in size) showing contour maps of daily total column ozone observed by a NASA spacecraft over Antarctica during two months of 1986. Although the system that generated the microfiche is primarily intended for interactive use in the analysis of many kinds of earth and space science data [Treinish, 1989], the needs of a low-cost browsing medium could only be met by employing its tools in the batch production of microfiche until fairly recently.

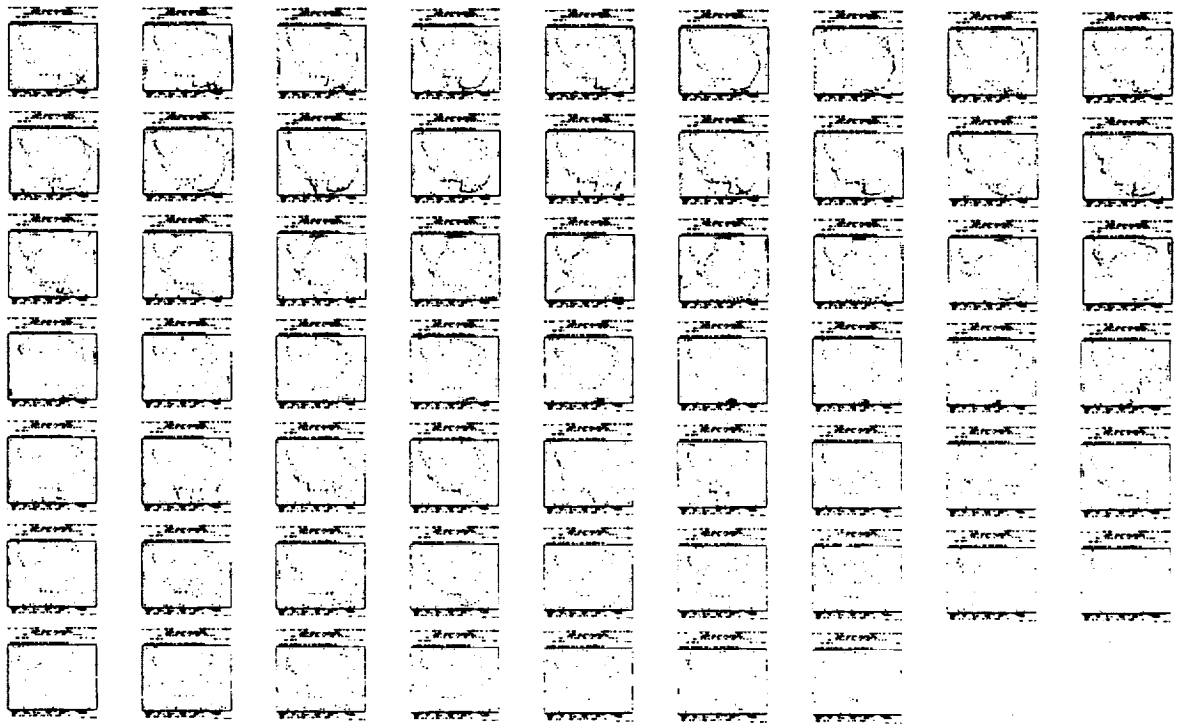

Figure 2. Azimuthal equidistant contours maps of Antarctic daily column ozone for September 24, 1986 through November 23, 1986 (from microfiche).

The advent of digital instrumentation, especially in remote sensing, created another role for browsing that is based upon the assumption that there is insufficient computing resources to support the analysis of all acquired data from all experiments of a specific project or mission. Hence, visual abstractions were created to help identify what subset of data should actually be fully processed to support scientific investigations. These graphical representations, known as *summary plots*, were generated on microfiche and distributed to all participants in a project. They usually contained a time sequence of simple visualizations of instrument outputs in specific modes at sufficient resolution to suitably identify a set of "events" as interesting, and thus warrant further processing. The particular presentations were chosen to highlight the signatures of such events within the limited graphical techniques that could be supported on monochromatic microfiche (line drawing, simple gray-scale polygon fill). Users of such a mechanism would receive a stack of microfiche each week, for example, and visually browse them, looking for patterns in the plots that would be indicative of something interesting (e.g., [Treinish, 1982]).

With improvements in technology to support interactive computing, most efforts associated with data browsing as level III in the aforementioned hierarchy of archive data management and access, have focused on simple images with image data. Furthermore, implementations are generally confined to one data set or a small number of similar data sets [cf, Simpson and Harkins, 1993; Oleson, 1992]. This has been a conscious choice based upon the need for supporting only a limited domain and/or driving interactivity. Unfortunately, these techniques are not applicable to many classes of data or when multiple disparate data sets are to be considered simultaneously, which are the characteristics of most current or planned archives.

Approach

There are four issues associated with interactive data browsing using visualization from archives of scientific data:

A. What are the visual abstractions for data presentation and interaction?

B. How are the browsing products distributed?

C. How is interactivity achieved?

D. What is the mechanism for integrating browsing into a data system?

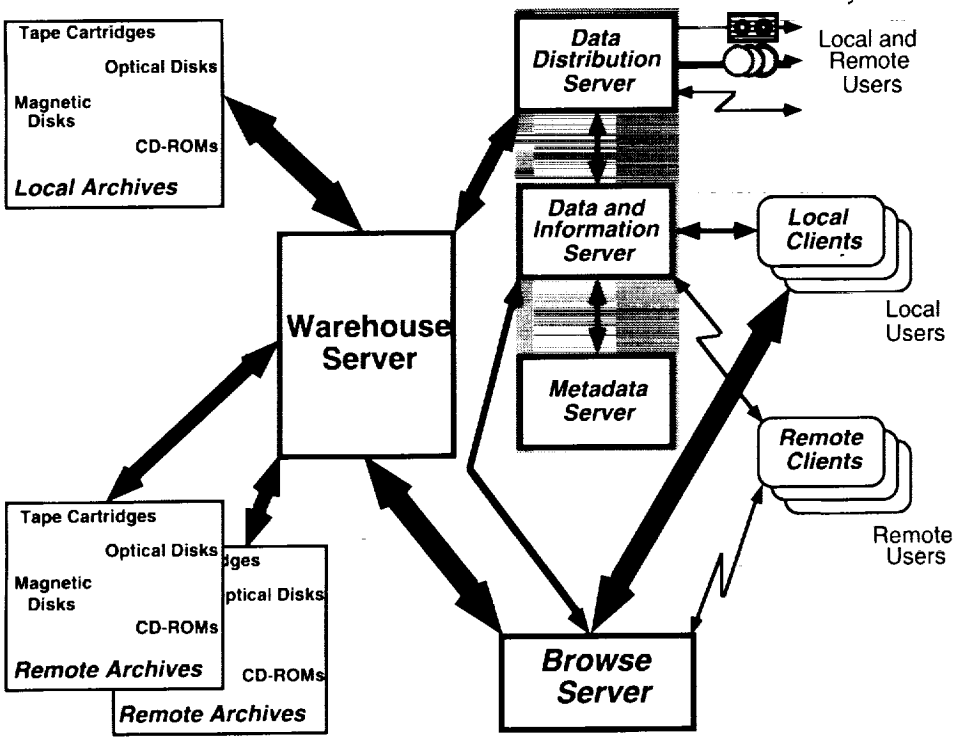

Figure 3 is a schematic of a simple interactive archive system. Each of the aforementioned levels of data management are shown, such that level I services are provided via archive, warehouse and data distribution servers. Level II services are within the gray box: data and information, and metadata servers. **Figure 1** illustrates the relationship between these components and specific data management technology.

Figure 3. Architecture for an interactive archive system utilizing visualization browsing.

Browsing (level III) is provided via the browse server, and local and remote clients. At this level of detail, any level IV services would be embedded within the local and remote clients. The schematic only indicates the major interaction paths between each component as arrows. The arrow thickness corresponds to relative bandwidth of communications, where thickest arrows would imply the greatest bandwidth (e.g., ANSI HiPPI), and the thinnest arrow would imply the least (e.g., Ethernet). A jagged arrow refers to traditional remote network communications. In addition, remote communications could be disconnected by distribution of products to a remote site via an alternate medium (e.g., CD-ROM) to be utilized with a local client at the remote site.

Abstraction

Efforts in scientific visualization typically focus on the analysis of data in a post-processing mode. This is still the emphasis even with the potential advent of computational steering or telescience, the remote sensing equivalent. Browsing is a subjective process involving the human visual system, which is consistent with one of the origins of the notion of scientific visualization as a method of computing. Consider the process by which an individual as an observer walks into a room. That person scans the room, and identifies and categorizes the contents (objects) in the room. From the classification or browsing process, further interaction with those contents may take place. Hence, the requirements for qualitative browsing presentation are not the same as for analysis.

As with primitive summary plots, browsing visualization methods should intentionally transform the structure of the data to highlight features to attract attention, scale dimensions to exaggerate details, or segment into regions [Rogowitz and Treinish, 1993a]. Traditional image browsing is very limited in this regard. Highlighting can only be achieved by modifying a pseudo-color or false-color presentation, which may not be sufficient even for image data or data than can be represented by an image. Of course, this method cannot even be considered for data that cannot be visualized as a simple image (e.g., any non-scalar data, any data of more than two dimensions).

Visual highlighting is often ill-advised for analysis because of the intentional distortion. In general, the browsing visualization should show data in the proper context (e.g., spatial or temporal). This is especially true in the earth and space sciences, where acquired data need to be registered in some (geographic) coordinate system for viewing. If the browsing visualization is to show multiple parameters, then the highlighting techniques must preserve the fidelity of the data (e.g., regions of missing data, original mesh structure) well enough, so that the artifacts in the presentation are not erroneously interpreted as potential (correlative) features in the data. Although the focus of such techniques herein is not for quantitative study, they may have a role in such tasks [Treinish, 1993a].

Distribution

One potential problem with the visual browsing of large data archives is the need to be in relatively close proximity to either the data archives or a facility that can generate the browse products (i.e., high-bandwidth access to an interactive system). In general, high-bandwidth communications between an archive and its users is not always practical, given typical geographic dispersal of scientists and their need for utilizing more than one archive.

Since interactive browsing has usually considered only image data, the distribution of browse products has focused on the distribution of these same images in some form, usually lossy compressed because of the typical size of image archives [e.g., Simpson and Harkins, 1993]. In this sense, the browsing and visualization media and the data are all the same. Hence, the compression is of the original data, where the compressed data will be easier to distribute compared to the original data, but they only apply to a limited class of data (i.e., images). Furthermore, the resultant compressed form may not be suitable for browsing and the quality may be poor.

Alternatively, consider the distribution of compressed visualizations instead of compressed data. These visualizations would be represented as images, so that available image compression technology could be easily utilized. As with image data, uncompressed visualizations would be of high quality but of sufficient volume to be impractical to distribute to remote sites. The compressed visualization images, would be easy to distribute as with compressed image data. The key difference is that the compressed visualization could apply to any collection of data and would be of high quality. Hence, one could consider the distribution of compressed visualizations as the *summary plots* of the 1990s because the lack of resources to access or distribute all archived data or their abstractions is similar to the lack of resources to

process or analyze all acquired bits, which motivated the generation of summary plots in the past.

Such an approach could be extended for predefined access and browsing scenarios, where the compressed visualizations are available for on-line remote access or via CD-ROM. In this case, the viewing of such compressed visualizations would require relatively simple, low-cost display software and hardware, which is becoming more readily available in desktop environments. In general, the costs associated with the access and distributed of compressed visualizations will be similar to those of image data, but of course, be significantly less than those of data. There will be an additional cost of generating the visual browse products, which is justified given the added value and obviating the need for distributing the data themselves.

Interactivity

For visual browsing to be effective it must be interactive. Otherwise, it is little different than watching television or traditional image browsing. One aspect of the interaction is in terms of data management: selection of and access to data of potential interest and metadata to help in guiding that process. In terms of visualization, the ability to interact with the browse "objects" (e.g., spatially, temporally) in near-real-time is critical. This requires rapid access to the data and generation of the browse products.

Implementation

Abstraction

The requirements to create qualitative visualizations that are effective as browse products do have implications on the software used to create them. Registration of data into an appropriate coordinate system for viewing requires the support of user-defined coordinate systems. To be able to properly show more than one data set simultaneously requires the ability to operate on different grid structures simultaneously and transformation of grid geometry independent of data. Depending on the visualization strategies being used, rendered images may need to contain different geometries (e.g., points, lines, surfaces and volumes independent of color or opacity or of original grid structure). (See Treinish [1993a] for a discussion of these ideas with respect to data analysis.)

A commercial scientific visualization environment (IBM Visualization Data Explorer - DX) has been used to experimentally implement the aforementioned browsing techniques. DX is a general-purpose software package for scientific data visualization. It employs a data-flow-driven client-server execution model and is currently available on Unix workstation platforms (e.g., manufactured by Sun, SGI, IBM, HP and DG) as well as a parallel supercomputer, IBM POWER Visualization System [Lucas et al, 1992].

Distribution

Near-real-time browsing of visualizations at sufficient resolution to see relevant contents requires the distribution of a large number of images. Clearly, lossy compression is necessary to drive viewing with update rates near the refresh rate of display controllers. For utilization remote from the archive or browse server, low-cost, simple display software and hardware is needed as well. There is much literature on data compression strategies and algorithms (e.g., a few hundred citations reported over the last decade by the National Technical Information Service alone, [NTIS, 1992]), which will not be discussed herein. This notion of visualization distribution is built upon the extant and growing body of implementations of compression algorithms, which are being utilized in scientific, multimedia and entertainment applications.

The idea of distributing imagery for visualization is not new. For example, Johnston et al [1989] experimented with both block-truncation and Lempel-Ziv compression for the distribution of visualization animation. Rombach et al [1991] discussed the Joint Photographic Experts Group (JPEG) compression scheme for the distribution of cardiographic imagery from different sources like ultrasound, magnetic resonance imagery and angiography. In these and other cases, the authors considered a low-cost viewing environment on the desktop as being critical, especially if the expense of generating the images to be distributed is high. Therefore, in this initial implementation, block-truncation (lossy, i.e., reducing the number of colors to represent a full-color pixel), modified Lempel-Ziv (lossless, e.g., like Unix compress) and temporal coherence between animation frames (lossy, e.g., the Moving Picture Experts Group, MPEG [LeGall, 1991]) will be considered.

To illustrate the viability of this approach, consider a modest data set, composed of a rectilinear scalar field of 32-bit floating-point numbers (e.g., atmo-

spheric temperature) at one-degree of geographic resolution at seven levels in the earth's atmosphere. Therefore, each time step would require about 1.7 MB. If these are daily data then less than a year would fit on a single CD-ROM, uncompressed. This does not include ancillary data required for annotation such as coastline maps, topography or other reference material. Lossy compression would not be relevant since the data are not imagery. Lossy compression could be applied to each layer of the atmosphere individually. However, the results would be rather poor (i.e., the two-dimensional spatial resolution is already low, 180 x 360), and spatial coherence for the entire volume could not be maintained. If lossless compressed, decompression could be expensive (e.g., Lempel-Ziv) or inconvenient (e.g., scaled/encoded 12- or 16-bit integers). Either compression approach is highly sensitive to the contents of the data set.

Alternatively, visualization compression is independent of data characteristics and only the resolution of the visualization image(s) drive the compression/distribution/decompression cost. Of course, the distribution of uncompressed browse visualizations is expensive, potentially more than that of the uncompressed data. Lossless compression although cheaper to distribute would still require the decompression process, and could also be more expensive than that for the data themselves. Hence, the lossy compression of the visualization imagery is the best approach from both a cost perspective as well as from that of image quality. Hence, for sequence of 640x480 24-bit image representations of the simple volumetric data set, over 14 years worth of frames for such daily data could be stored using a simple 8:1 block truncation compression (i.e., each 24-bit pixel is represented by three bits) on a single CD-ROM. Using 32:1 JPEG compression, a sequence of over two years of these images at hourly resolution could be stored on a single CD-ROM.

Interactivity

For browsing to be effective it must be interactive with near-real-time system response. With data sets of interesting size, e.g., ≥ O(1 GB), immediate interaction cannot take place on current conventional systems (i.e., high-end graphics workstations). Even though a 1 GB data set is admittedly modest by today's standards for data generation, the access and visualization of an entire data set for browsing or even a large fraction of it place significant burdens on the floating point and bandwidth capacities of the computer system being employed. The bandwidth requirements are derived from the bulk access speeds of large data sets and the transmission of images sufficiently fast to be interactive. The floating point requirements stem from three classes of computation for visualization:

1. Transformation (e.g., warping, registration)

2. Realization (e.g., contouring, color mapping, surface deformation)

3. Rendering (i.e., creating images)

Although the visualization requirements are different, the computational needs of interactive browsing are very similar to those of visualization in a virtual world environment (e.g., [Bryson and Levit, 1992]).

Experimentation with a commercial parallel supercomputer (IBM POWER Visualization System) and the aforementioned DX environment has shown the viability of such interactive visual browsing, even with multiple data sets. In this effort, the PVS and DX combination has been used as the browse server with local and remote clients on workstations as indicated in **figure 3**. The PVS functions as an archive server in this context. **Figure 4** shows the relationship between the browse server, and the data and information server and its components in the interactive archive system shown in **figure 3**.

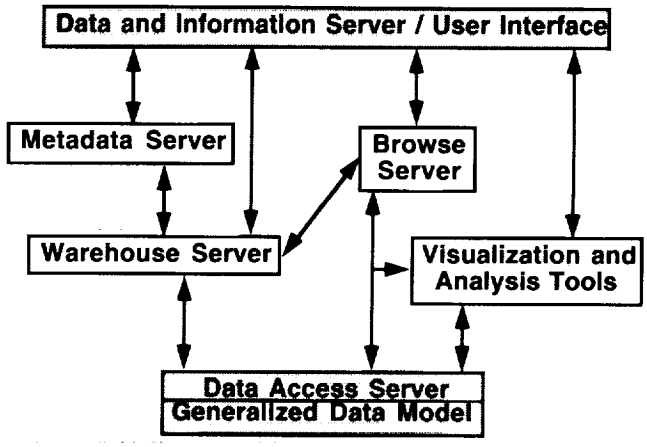

Figure 4. Architecture for a data and information system incorporating visualization browsing.

The IBM POWER Visualization System (PVS) is a medium-grain, coherent shared-memory parallel supercomputer with the interactivity of a workstation. This has been achieved via a programmable (general-purpose) approach instead of special-purpose hardware with balance among floating point

performance via moderate parallelism, large physical memory and high-speed external and internal bandwidth. The PVS consists of three major hardware components: server, disk array and video controller. The server is a symmetric multi-processor with up to 32 processors (40 MHz Intel i860XR or 44 MHz Intel i860XP), a 1.28 GB/sec (at 40 MHz) internal backplane supporting a hierarchical bus structure, hierarchical memory (16 MB local memory per processor and up to 2 GB global/shared), ANSI HiPPI communications, fast and wide SCSI-2, and an IBM RISC System/6000 support processor (for local area network and storage access). The server supports parallelized computations for visualization via DX. The disk array is a HiPPI-attached RAID-3 device with either 50 MB/sec or 95 MB/sec sustained access speeds, or a fast and wide SCSI-2 four-bank RAID-3 device with 76 MB/sec sustained access speeds. It provides access to archived data to be browsed. The video controller is a programmable 24-bit double-buffered with 8-bit alpha overlay (for custom XWindow server) frame buffer attached to an IBM RISC System/6000 workstation. It receives images from the PVS server via HiPPI (either compressed or uncompressed) at resolutions up to 1920x1536, including HDTV, for real-time image updates. The video controller provides an interface for interaction with and viewing of the browsing visualization at speeds up to 95 MB/sec.

Results

Abstraction

Figure 5 shows a traditional two-dimensional visualization of ozone data similar to those shown in **figure 2**. The data are realized with a pseudo-color map and iso-contour lines for September 30, 1992. The rectangular presentation of the data is consistent with the provided mesh in that it is torn at the poles and at a nominal International Date Line. The ozone data are overlaid with a map of world coastlines and national boundaries in magenta as well as fiducial lines (of latitude and longitude) in white.

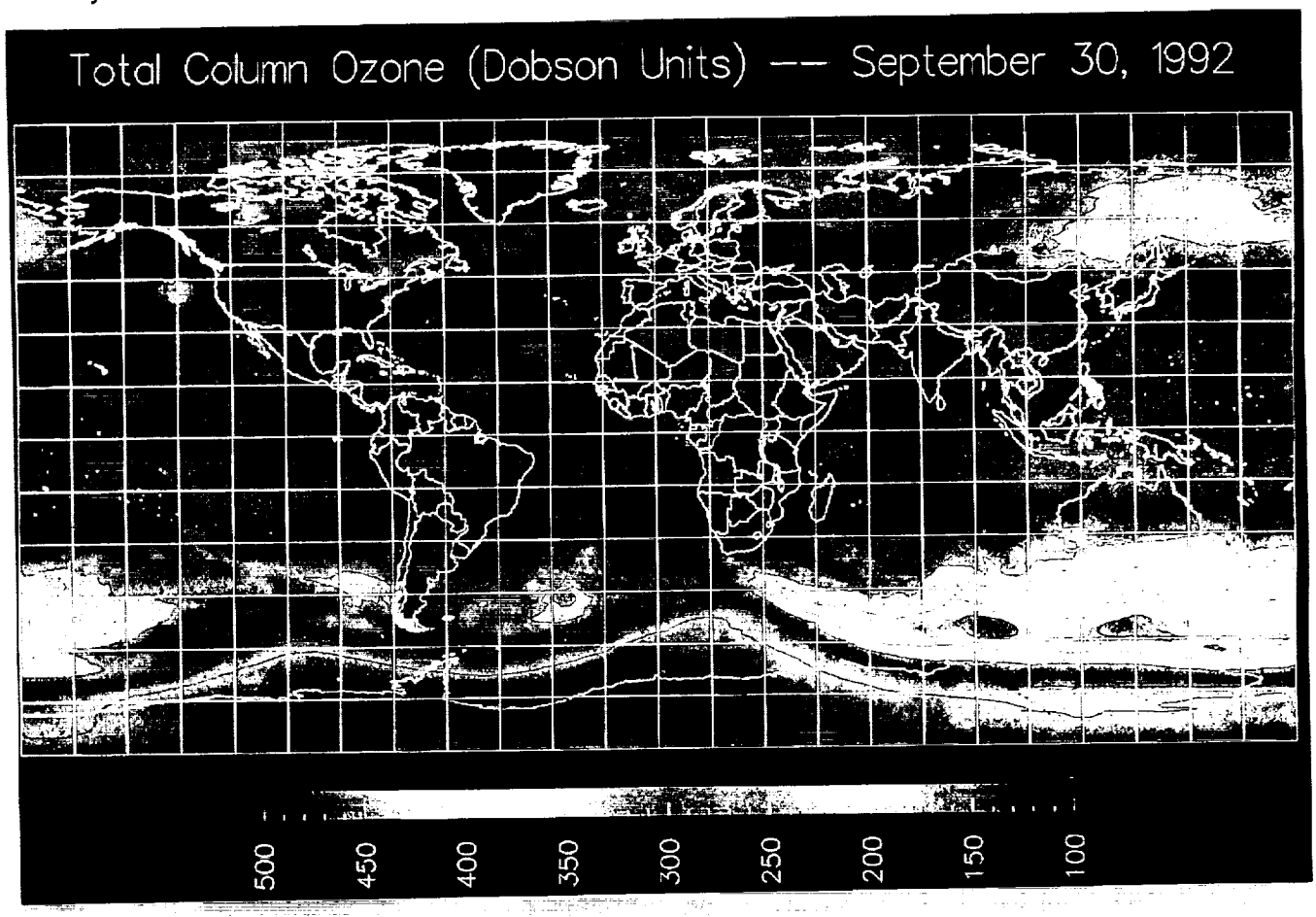

Figure 5. Pseudo-color-mapped global column ozone on September 30, 1992.

To provide a qualitative impression for browsing, the data are transformed to a three-dimensional continuous spherical surface in **figure 6**. The ozone is triply redundantly mapped to radial deformation, color and opacity so that high ozone values are thick, far from the earth and reddish while low ozone values are thin, close to the earth and bluish. Replacing the map for annotation is a globe in the center of this ozone surface. The use of three redundant realization techniques results in textures for qualitatively identifying regions of spatial or temporal interest. The gauges on the left illustrate the daily **total ozone** statistics. The pseudo-hour hand position ranges from 100 to 650 Dobson Units, while the color corresponds to that of the ozone surface. From the top they show the mean, minimum and maximum for each day. The value corresponding to geographic view for each frame is shown next. At the bottom is a bar chart indicating the standard deviation of the daily measurements. This approach to qualitative visualization is potentially applicable to a large variety of simulated or observed earth and planetary data on a large spatial scale, especially for two and three-dimensional scalar fields.

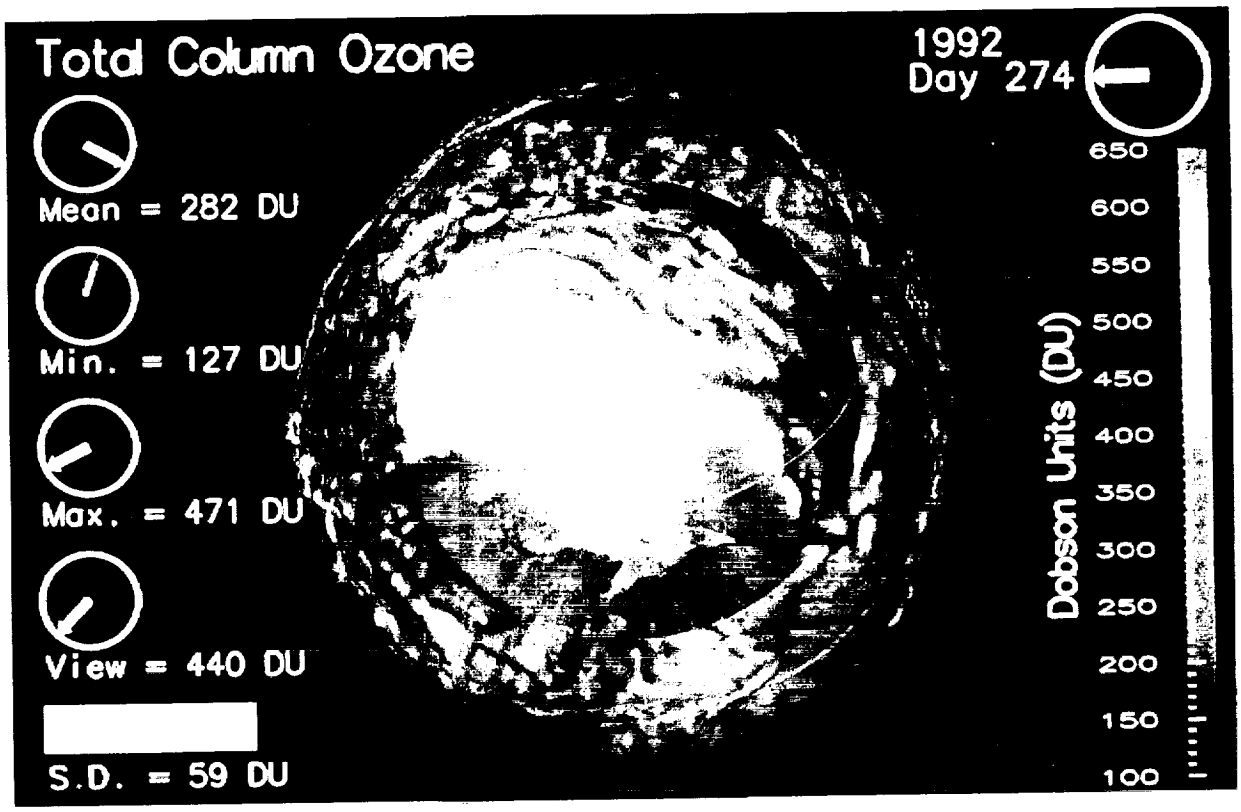

Figure 6. Radially deformed pseudo-color and opacity-mapped spherically warped surface of global column ozone on September 30, 1992 with annotation.

A browsing animation of the ozone would be one that illustrates the data on a daily basis as in **figure 6** for the entire archive of available data (i.e., late 1978 through early 1993). The geographic view of these data would change with each day to provide reasonable coverage of the entire globe over a complete year. The view would be chosen to concentrate on interesting regions such as the poles during appropriate seasons like spring. Treinish [1992a] and Treinish [1993a] are examples of such a browsing animation, which are useful a posteriori of its generation for identifying periods of time or geographic regions that warrant further study.

These browsing animation sequences are derived from about 1 GB of data, a two-dimensional scalar field

over a torn geographic mesh. For each of the over 4700 frames of the sequence, several calculations are required to create a visualization image. Each image is actually composed of two images, which have been blended. There is a background image, which is composed of frame-variant contents under a constant view: primarily opaque polygonal text, dials and bars. This annotation changes to summarize daily statistics. There is a foreground image, which is also composed of frame-variant contents, but with a frame-variant view. Each foreground image contains a static globe with the surrounding translucent ozone surface. For each day, the ozone data are transformed (irregularized to remove regions of missing data and warped onto a sphere), realized (color and opacity mapped and surface deformed) and full-color rendered (about 45,000 to 50,000 translucent quads for the ozone; 259,200 full-color-mapped opaque quads on a sphere with normals for the globe). The foreground and background images are brightened and blended to compose each final frame. Each frame at workstation resolution (about 1.2 million pixels) using DX on a 32-way (40 MHz) PVS required about 12 seconds of computing time. Hence, the entire animation took about 15 hours at that resolution.

Distribution

Current efforts on data distribution have focused on the application of compression techniques to three sample animation sequences on a 32-way (40 MHz) PVS equipped with a RAID-3 HiPPI disk array capable of 50 MB/sec sustained access speeds. The first example is a high-resolution sequence of 2040x1536 32-bit (8-bits of red, green, blue, alpha) images, 88 frames in length totalling about 1052 MB. An 8:1 block-truncation lossy compression (i.e., each 32-bit pixel is represented by 4 bits) required about 41 seconds, resulting in a rate of approximately 26 MB/second or 2.15 Hz disk to disk. Lempel-Ziv lossless compression was applied to the entire sequence as a whole, not on a frame-by-frame basis. As expected the results were considerably slower, requiring about 2 minutes, 2 seconds, yielding a rate of approximately 8.7 MB/second disk to disk to achieve 45.4% compression. In both cases, the compression algorithms were parallelized on the PVS.

The second example is with a 151-frame sequence of 640x480 32-bit images of about six months worth of animation similar to that illustrated in **figure 6**. Results with this considerably smaller collection (about 177 MB) are quite similar, pointing to the potential scalability of the shared-memory, symmetric multiprocessor systems like a PVS to this problem. For the 8:1 lossy compression, about eight seconds were required yielding a rate of approximately 23 MB/second or 18.9 Hz disk to disk. The lossless compression of the entire sequence required about 22 seconds to achieve 62.6% compression at 8.0 MB/second disk to disk.

The third example is with the 5853 frames of a digital video (D1) sequence [Treinish, 1993b]. Most of the sequence is composed of frames similar to **figure 6** -- one for each day from January 1, 1979 through December 31, 1991. Each D1 frame is composed of 10-bits each of YUV (a chrominance and intensity-based specification of color) at 720 x 486 for playback at 30 Hz. Hence, this 3 minute, 15 second sequence is 10.6 GB in size, which is maintained as a single file on a PVS. A PVS-based MPEG compression facility was used to create a 250:1, lossy-compressed MPEG-1 sequence. Approximately, 12 minutes, 5 seconds were required for this operation, yielding a rate of 14.7 MB/second or 8 Hz, disk to disk.

Interactivity

Figure 7 is a snapshot of a DX Motif-based interface for a prototype interactive browsing system. It provides very simple modes of interaction: selection of space and time (i.e., geographic regions or seasons of interest) for browsing via the spherically warped presentation shown in **figure 6**. There are dial widgets for the specification of geographic viewing centroid of the global "object", and slider widgets for selection of the year to examine and the viewing width. A VCR-like widget provides control over the choice of the portion of the year to browse by Julian day. Optionally, a zonal slice of the data being browsed at the specified longitude may be shown as a pseudo-colored line plot of the latitudinal distribution. The data illustrated in **figure 7** are the aforementioned global column ozone data derived from the same 14-year archive as shown in **figures 2, 5** and **6**.

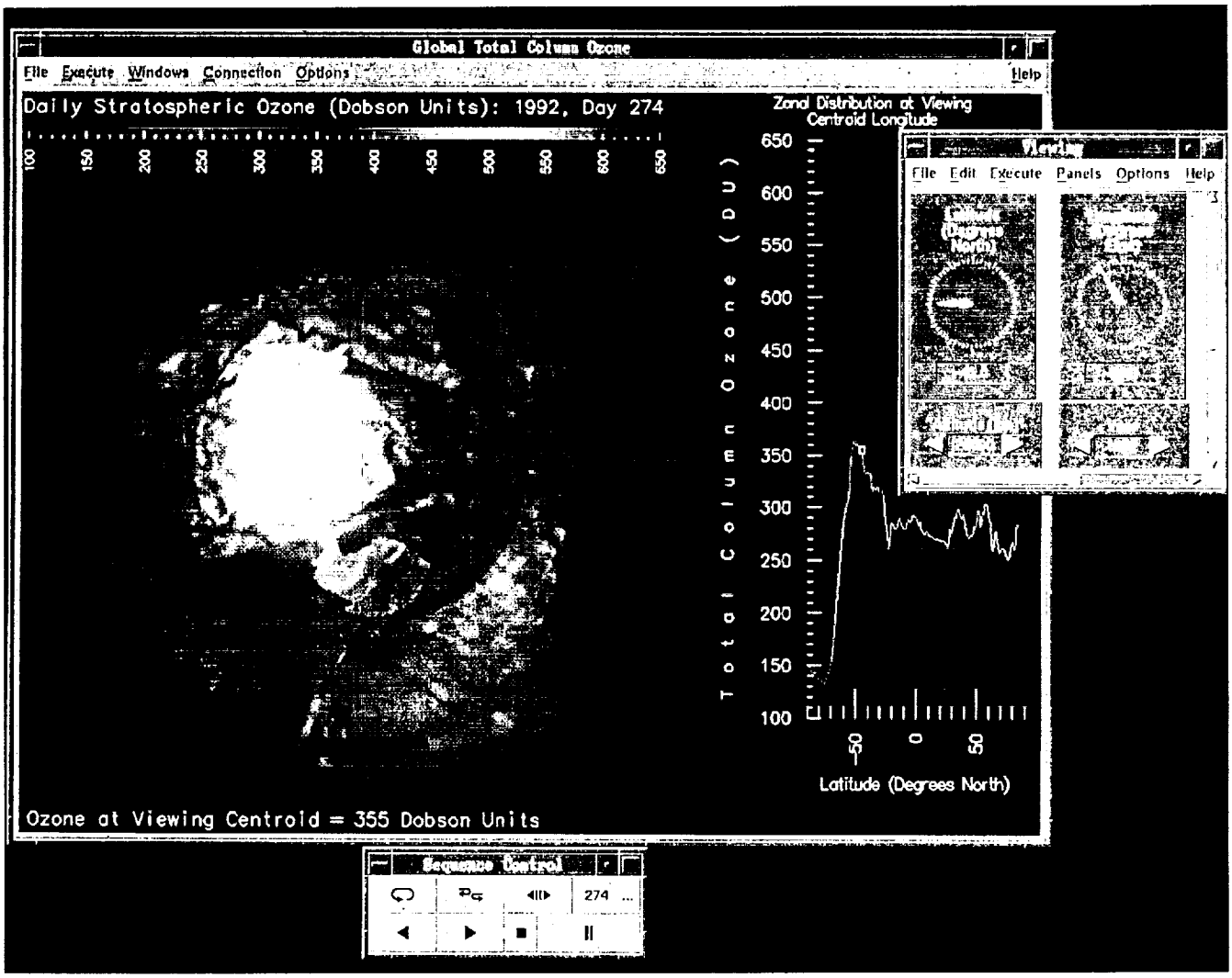

Figure 7. Example user interface for browsing showing global column ozone on October 1, 1987.

The prototype browsing system is built as a client-server, consistent with the architecture of DX, as shown in **figure 8**. A PVS functions as the browse server in this implementation. High-speed display of browsing visualizations is local to the PVS. Remote display is via standard XWindow services with update rates limited to what the network infrastructure can provide.

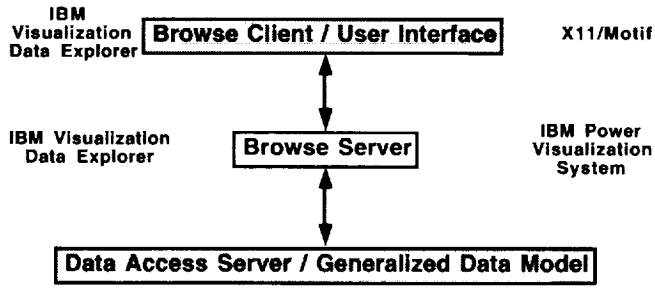

Figure 8. Architecture for a prototype interactive browsing system.

Conclusions

A focus on qualitative methods of presenting data shows that visualization provides a mechanism for browsing independent of the source of data and is an effective alternative to traditional image-based browsing of image data. To be generally applicable, such visualization methods, however, must be based upon an underlying data model with support for a broad class of data types and structures.

Interactive, near-real-time browsing for data sets of interesting size today requires a browse server of considerable power. A symmetric multi-processor with very high internal and external bandwidth demonstrates the feasibility of this concept. Although this technology is likely to be available on the desktop within a few years, the increase in the size and complexity of archived data will continue to exceed the capacity of "workstation" systems. Hence, a higher class of performance, especially in bandwidth, will generally be required for on-demand browsing.

A few experiments with differing digital compression techniques indicates that a MPEG-1 implementation within the context of a high-performance browse server (i.e., parallelized) is a practical method of converting a browse product to a form suitable for network or CD-ROM distribution.

Future Work

From this initial prototype implementation of an interactive data browser, there are several areas for future work. Since practical low-cost decompression of JPEG-compressed images is becoming available on the desktop, experimentation with JPEG is warranted [Pennebaker and Mitchell, 1993]. As with JPEG, the MPEG-1 motion video compression technique is becoming available for multimedia applications of video sequences on the desktop, whether the animation is distributed via network or CD-ROM. Additional testing with MPEG-1 and the higher quality, MPEG-2 as it becomes available is required within the browse server as well as on various desktop playback systems.

The second area of research would focus on fleshing out more of the interactive archive architecture schematically illustrated in **figures 1, 3, 4** and **8**. Specifically, the prototype interface and visualization could migrate to a data-driven one conceptually similar to the primitive implementation discussed by Treinish [1989], which could be integrated with metadata and data servers to achieve a browsing archive system (i.e., data management services at the aforementioned levels I, II and III). This would also imply the availability of integrated data and information services similar to those in the rudimentary system described by Treinish and Ray [1985]. Such an approach would be further enhanced by the integration of the prototype browsing system with tools for data analysis, which are already available.

The strategies for qualitative visualization have focused on only a few methods for spherically-oriented data with large spatial extent. Clearly, investigation of alternative approaches of highlighting features in such data are required, for which there are a number of potential issues [Rogowitz and Treinish, 1993b]. In addition, the extension of this browsing architecture to other classes of data is also warranted.

Acknowledgements

All of the data sets discussed above were provided courtesy of the National Space Science Data Center, NASA/Goddard Space Flight Center, Greenbelt, MD.

References

1. Brown, S. A., M. Folk, G. Goucher and R. Rew. *Software for Portable Scientific Data Management.* **Computers in Physics**, 7, n. 3, pp. 304-308, May/June 1993.

2. Bryson, S. and C. Levit. *The Virtual Wind Tunnel: An Environment for the Exploration of Three-Dimensional Unsteady Flows.* **Proceedings IEEE Visualization '91**, pp. 17-24, October 1991.

3. Fekete, G. *Rendering and Managing Spherical Data with Sphere Quadtrees.* **Proceedings IEEE Visualization '90**, pp. 176-186, October 1990.

4. Haber, R., B. Lucas and N. Collins. *A Data Model for Scientific Visualization with Provisions for Regular and Irregular Grids.* **Proceedings IEEE Visualization '91 Conference**, pp. 298-305, October 1991.

5. Johnston, W. E., D. W. Robertson, D. E. Hall, J. Huang, F. Renema, M. Rible, J. A. Sethian. *Video-based scientific visualization.* **Proceedings of a Workshop on Geometric Analysis and Computer**

Graphics, University of California at Berkeley, Berkeley, CA, pp. 89-102, May 1989.

6. LeGall, D. J. *The MPEG Video Compression Standard.* **Proceedings IEEE COMPCON Spring '91**, pp. 334-335, 1991.

7. Lucas, B., G. D. Abram, N. S. Collins, D. A. Epstein, D. L. Gresh and K. P. McAuliffe. *An Architecture for a Scientific Visualization System.* **Proceedings IEEE Visualization '92**, pp. 107-113, October 1992.

8. National Technical Information Service. **Data Compression (Citations from the NTIS Database).** NTIS Document PB92-802750, U. S. Department of Commerce, Springfield, VA, 1992.

9. Oleson, L. The Global Land Information System. Eros Data Center, U. S. Geological Survey, Sioux Falls, SD, 1992.

10. Pennebaker, W. B. and J. L. Mitchell. **JPEG: Still Image Data Compression Standard**. To be published by Van Nostrand Reinhold, 1993.

11. Rogowitz, B. E. and L. A. Treinish. *Data Structures and Perceptual Structures.* **Proceedings of the SPIE/SPSE Symposium on Electronic Imaging, 1913**, pp. 600-612, February 1993.

12. Rogowitz, B. E. and L. A. Treinish. *An Architecture for Perceptual Rule-Based Visualization.* **Proceedings IEEE Visualization '93**, pp. 236-243, October 1993.

13. Rombach, M. R., U. Solzbach, U. Tites, A. M. Zeiher, H. Wollschlager, H. Just. *PACS for cardiology - Perspective or Fiction?* **Proceedings IEEE Computers in Cardiology**, Venice, Italy, pp. 77-79, September 1991.

14. Simpson, J. J. and D. N. Harkins. *The SSable System: Automated Archive, Catalog, Browse and Distribution of Satellite Data in Near-Real Time.* **IEEE Transactions on Geoscience and Remote Sensing, 31**, n.2, pp. 515-525, March 1993.

15. Treinish, L. A. *The Dynamics Explorer Summary Plot Software System.* NASA/Goddard Space Flight Center, Internal Report, March 1982.

16. Treinish, L. A. *An Interactive, Discipline-Independent Data Visualization System.* **Computers in Physics, 3**, n. 4, July 1989.

17. Treinish, L. A. *Climatology of Global Stratospheric Ozone (1979 through 1991).* **Proceedings IEEE Visualization '92**, video supplement, October 1992.

18. Treinish, L. A. *Unifying Principles of Data Management for Scientific Visualization.* **Proceedings of the British Computer Society Conference on Animation and Scientific Visualization**, Winchester, UK, December 1992 and **Animation and Scientific Visualization Tools and Applications** (R. Earnshaw and D. Watson, editors), Academic Press, pp. 141-169, 1993.

19. Treinish, L. A. *Visualization of Stratospheric Ozone Depletion and the Polar Vortex.* **Proceedings IEEE Visualization '93**, pp. 391-396, October 1993.

20. Treinish, L. A. *Climatology of Global Stratospheric Ozone (1979 through 1991).* **ACM SIGGRAPH Video Review, 93**, August 1993.

21. Treinish, L. A. and S. N. Ray. *An Interactive Information System to Support Climate Research.* **Proceedings of the First International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology**, American Meteorology Society, pp 72-79, January 1985.