

ornl

RECEIVED

ORNL/TM-13193

OCT 16 1996

OSTI

**OAK RIDGE
NATIONAL
LABORATORY**

LOCKHEED MARTIN



Monitoring and Visualizing Information Resources

D. Scott McCrickard
Thomas H. Rowan

MANAGED AND OPERATED BY
LOCKHEED MARTIN ENERGY RESEARCH CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL-27 (3-96)


DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P.O.Box 62, Oak Ridge, TN 37831; prices available from (423) 576-8401, FTS 626-8401.

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

**Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.**

Computer Science and Mathematics Division
Mathematical Sciences Section

MONITORING AND VISUALIZING INFORMATION RESOURCES

D. Scott McCrickard †
Thomas H. Rowan ‡

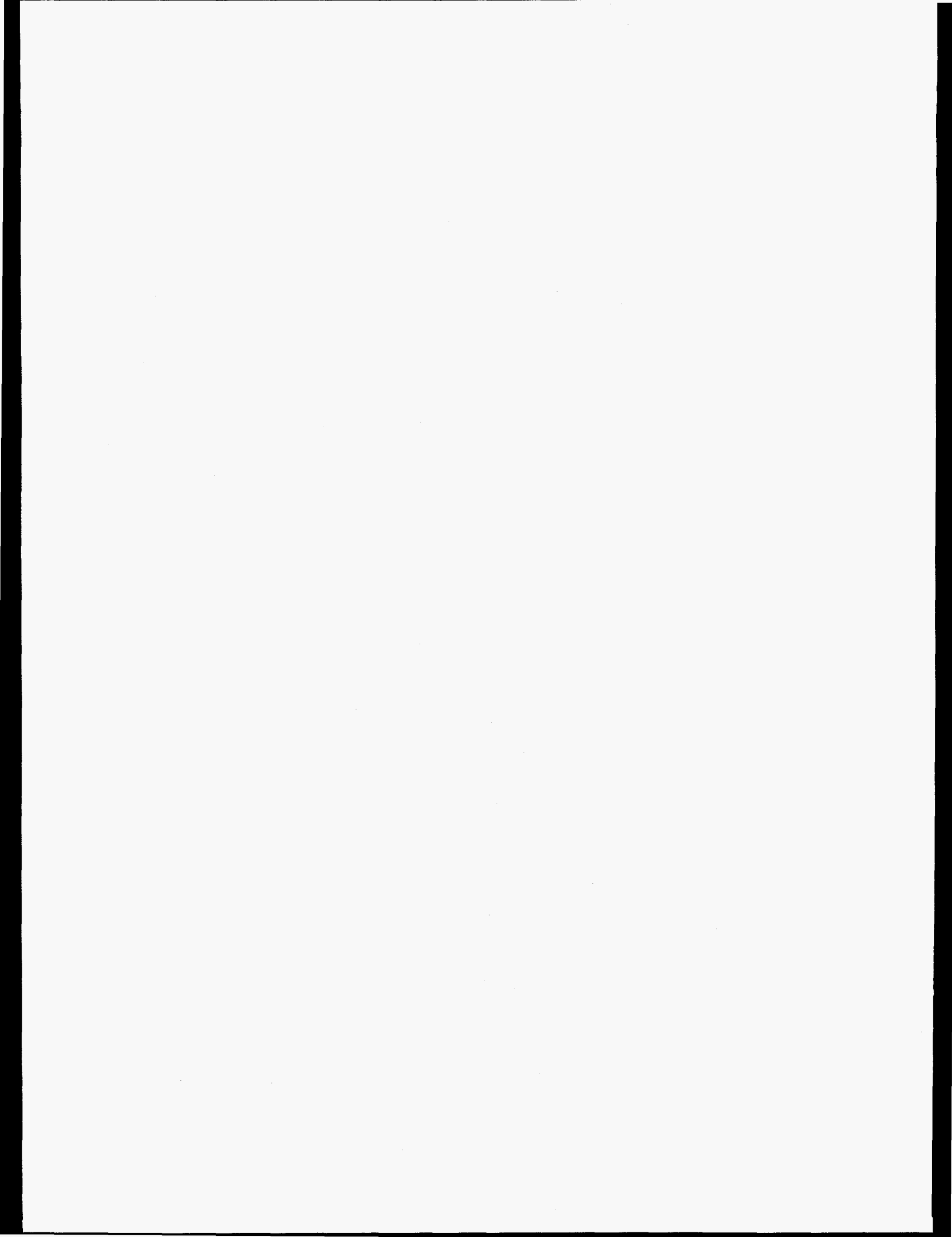
† Graphics, Visualization, and Usability Center
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
Email: mccricks@cc.gatech.edu
(Author to whom correspondence should be addressed)

‡ Mathematical Sciences Section
Oak Ridge National Laboratory
P.O. Box 2008, Bldg. 6012
Oak Ridge, TN 37831-6367
Email: rowan@msr.epm.ornl.gov

Date Published: July 1996

Research was supported in part by the Applied Mathematical Sciences Research Program of the Office of Energy Research, U.S. Department of Energy and by the U. S. Department of Energy, Defense Programs Office of Economic Competitiveness National Information Infrastructure Major Partnership.

Prepared by the
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
managed by
Lockheed Martin Energy Research Corp.
for the
U.S. DEPARTMENT OF ENERGY
under Contract No. DE-AC05-96OR22464



Contents

1 Introduction 1

2 Information monitoring and visualization issues 1

 2.1 Unifying resource monitoring and display 1

 2.2 Information display techniques 2

 2.3 Graphical data display 2

 2.4 Summary 3

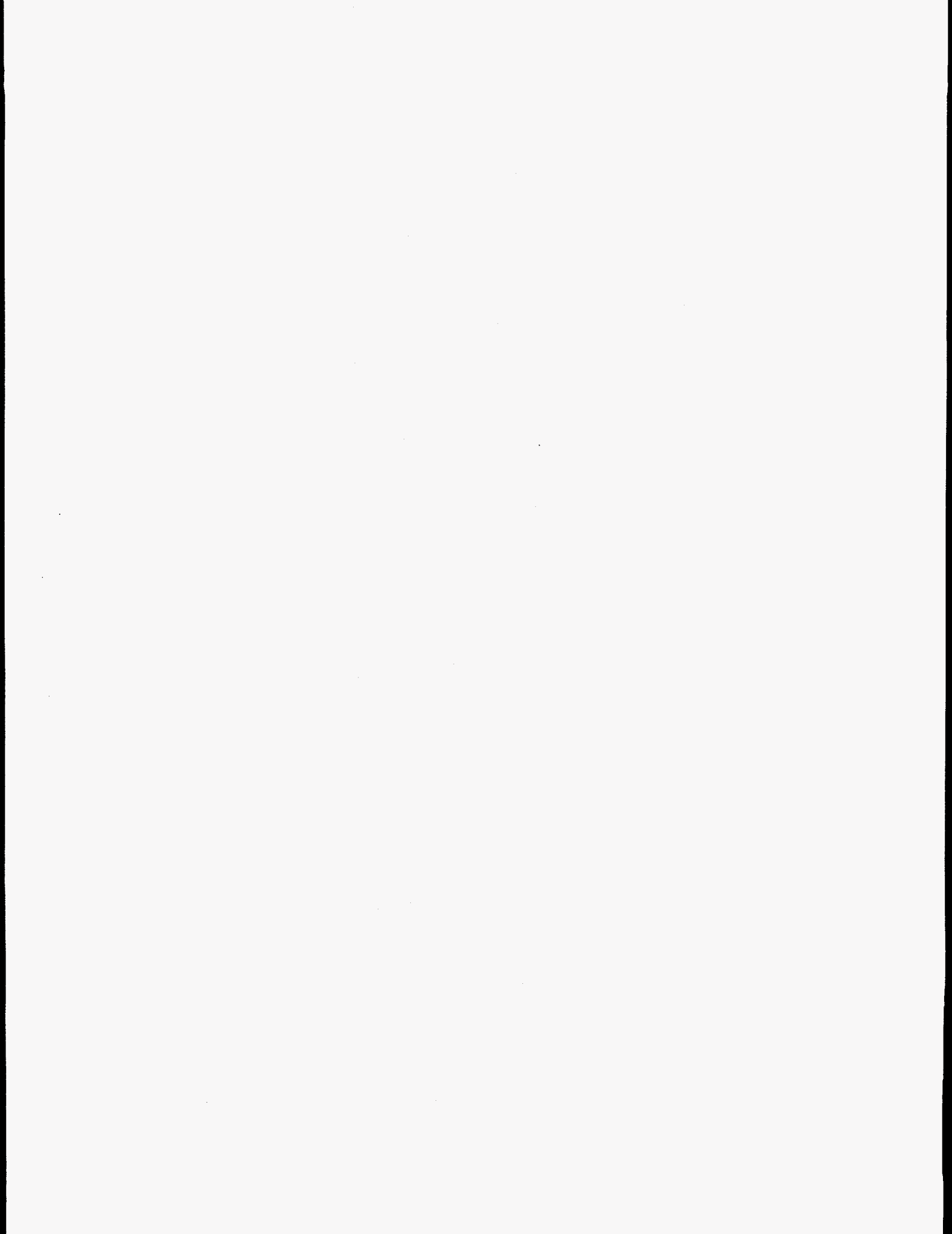
3 Irwin 3

4 Irwin's confluent zoom 4

5 Future work 5

6 Conclusions 6

7 References 6



MONITORING AND VISUALIZING INFORMATION RESOURCES

D. Scott McCrickard
Thomas H. Rowan

Abstract

The continuous increase in information necessitates monitoring and display techniques that maximize comprehension yet minimize effort. In this paper, we discuss the use of hypertools, confluent zoom, and graphical encoding of text as solutions to this problem, and we introduce Irwin, an information resource monitoring and display tool.

Keyword: information visualization, confluent zoom, email, World-Wide Web

1. Introduction

The exponential growth of the Internet combined with improving technology in telephone pagers, fax machines and modems results in a constant flood of information to users. A variety of tools can organize, filter, and display this information, but each of these tools requires frequent accesses to stay updated on important happenings. If users know when and in what ways the information changes, they can make informed decisions on when to access a resource, thus better utilizing their time. For example, since some WWW sites change infrequently, a user should be alerted of the change when it takes place. Conversely, some news groups and mailing lists have lots of activity, and a user may want to monitor their activity but access them only when something of interest occurs. These kinds of changes need to be summarized into a display that will maximize comprehension while minimizing effort.

This paper explores techniques for monitoring and visualizing information resources. We discuss the advantages of hypertools in the monitoring and display of information. We introduce the concept of a confluent zoom as a way to improve understanding and navigation. Several techniques are described for graphically encoding information in a small space. Finally, we introduce Irwin, an information resource monitor that exemplifies many of the discussed techniques.

2. Information monitoring and visualization issues

In this section we explore techniques for unifying resource display and monitoring into a single interface, we discuss methods for navigating through large amounts of data in a limited space, and we outline ways to convert the textual information into graphical form.

2.1. Unifying resource monitoring and display

The large number of browsing, searching, and visualization tools necessitates a method for unifying these tools in a single location. Companies such as America On-Line (AOL) and CompuServe achieve this goal by integrating their own tools into a single interface. Unfortunately, as technology advances, these interfaces quickly become obsolete and must be modified and redistributed to all users. For example, the mail tool on AOL was designed for small, low traffic, in-house email messages. More advanced mailers designed to handle large amounts of email are unavailable to AOL users.

Alternatively, a collection of tools could be bundled as hypertools, where separate programs run simultaneously and communicate with each other. This hypertool concept is in keeping with the philosophy of allowing independent tools cooperate as seen in program environments like Field [9] and Yeast [5]. Field integrated a wide variety of Unix tools using a message-based integration mechanism. Yeast widened the domain to include all event-action applications. Since these techniques make use of existing tools and mechanisms, an updated tool would require little or no change to the environment. Even a new type of tool could be integrated into the environment with minimal effort.

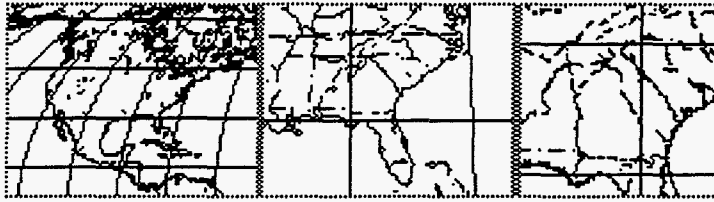


Figure 1: Confluent zoom from North America to Georgia.

2.2. Information display techniques

While it is important for a user to obtain information about interesting resources at any time with minimal effort, the information should be provided in a way that does not interrupt the user's work. With information taken from any number of tools at any time, displaying a comprehensive summary in a limited space becomes a formidable task.

Zooming provides a popular method for interacting with a large information space. The Pad++ interface [1] uses zooming as its primary navigation method. The user starts with an overview of the entire information space and selects a portion of interest. The interface zooms in on that section, replacing the existing view with an enlargement of the selected portion. Unfortunately, enlarging a section eliminates the view of the remaining information, potentially resulting in loss of context. In addition, since zooming can require a significant amount of user interaction, it should be avoided.

To resolve this problem, we introduce the concept of a confluent zoom. In a confluent zoom display, more detailed information is shown as the user reads from left to right. The leftmost views show an overview of the information resources but little detail. The middle views provide increasingly more detail on selected portions of the view to the left, and the rightmost view displays the object or text of the most interest. Since intermediate views have proven useful in graphical tools such as map browsers and medical imagery [7, 8], we expect that the intermediate views will smooth the transition from the high-level overview to the detailed zoom view.

Figure 1 provides an example of confluent zoom. The leftmost view is easily recognized as North America. The middle view focuses on the southeastern United States, and the rightmost view is centered on the state of Georgia. The context provided by the leftmost view makes it easier to recognize the middle and right views.

2.3. Graphical data display

The large amount of resource information requires methods for encoding this information into graphical form. In some manner, attributes of the text in the document must correspond to attributes of graphics. For instance, author, date, and size could be encoded with the color, size, and shape of graphical objects.

The pixel-to-character mapping employed in the history-enriched digital objects project [4] exemplifies one possible encoding. Each line of text is mapped to a line of pixels in the display. Thus, by looking at the graphical representation, the user can see the layout of the text. The SeeSoft program visualization tool [2] augmented this

display by mapping lines of code to lines of pixels, then coloring the lines of interest according to parameters such as author or date. If a user wants to identify the author of a portion of the code, then SeeSoft will assign a color to each programmer and color the lines appropriately. A similar method could be employed in a resource information visualization display. User-selected keywords would be assigned colors, then the display would color the word representations according to this assignment.

Assigning colors to words would only cover a small set of interesting words. Ideally, a visualization display should provide an automatic encoding for each word. One method would be to color the vowel positions in the graphical object for a word in correlation with the colors of the spectrum; that is, each 'a' position would be red, 'e' would be orange, and so on, with all consonant positions colored black. This creates an interesting (and usually different) visual pattern for a large number of words. The user can recognize repeated words by the repeated pattern in the graphical display. Over time the user should be able to recognize the patterns for certain words. This technique allows more word representations to be displayed in the same space compared to a textual representation.

2.4. Summary

The above arguments led to three goals for our information monitoring system.

- Collect information from various resources using existing tools. To account for future technologies, users must be able to configure our system to access information from any resource.
- Display a summary of the information constantly but non-obtrusively. While it is important for a user to obtain information about interesting resources at any time, the information should be provided in a way that does not interrupt the user's work.
- Graphically summarize the collected information. A graphical display of the contents of a resource will provide a quick overview in a small space with minimal effort from the user.

The next section describes the current state of our system.

3. Irwin

Irwin, the information resource watcher in a nutshell, provides users with a compact and configurable interface to information resources. Currently, Irwin can monitor email boxes, Usenet news groups, and WWW sites and can show information about them using a confluent zoom display. A configuration screen lets the user set the monitoring tools, the important names and words to find in the information, and a number of display options. When important changes occur, Irwin alerts the user by updating the confluent zoom information display. If the user desires more information, Irwin provides access to external tools. Figure 2 shows an Irwin display.

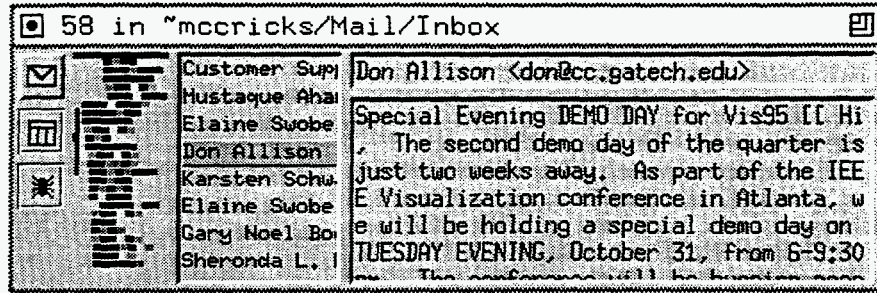


Figure 2: Irwin's display of an email file. The confluent zoom display simultaneously provides global context and local details. The graphical summary view (second from the left) contains a coded line representing each recently received message. The text views provide detailed information about a limited set of messages.

To facilitate information gathering, Irwin is designed as a collection of monitoring and display hypertools. A typical monitoring tool consists of an existing tool accessed by a Tcl/Tk front end. The front end accesses the tool at regular intervals, collects and parses information, and passes a summary to the display hypertool. The display hypertool maintains the confluent zoom that provides information about each resource. Since most of the information processing occurs in other hypertools, the display hypertool only processes user input and updates the display. This division of labor is important for slower interpretive languages like Tcl/Tk. Communication between tools is performed using Tk's *send* command. *send* was included in Tk to encourage the development of hypertools that can be connected and reused in interesting ways [6, pp.244-5]. The different levels in the confluent zoom are described in detail in Section 4.

The large amount of information combined with the limited space constraint necessitates a graphical encoding method. As a first step, we chose to focus on encoding the sender's (or author's) name by using a proportionally-sized line and the message arrival time by the position of the line. The user can create lists of senders of interest. Each list is assigned a color, and whenever a sender in a list appears in the display, the line is colored appropriately. As a default for names that do not appear in any list, Irwin colors the vowels in the object representing each sender's name. To encode the time, we listed the objects by arrival time then indented each object according to the hour at which the message was received. Thus, all messages received in the one o'clock hour are indented the same amount. If the user leaves the computer for a period of time, upon return the user can tell at a glance which messages were recently received.

4. Irwin's confluent zoom

The Irwin confluent zoom includes a number of views that inform the user on the status of selected resources. Each view (from left to right) provides more detailed information than the previous view; thus, the user can select a portion of one view and see details about the selected portion in the next. The views include a resource alert view, a summary view, a header list view, and a message information view.

- The resource alert view consists of a series of buttons whose appearance reflects the state of the resources. Based on the nature of the changes to the resource, its button will flash or change color, and an auditory cue may be sounded. When the user selects a resource by clicking on its button, its representation appears in the navigation bar view.
- The navigation bar view shows the graphical representation of the selected resource. Since a resource may have too many messages to view at full size, Irwin can shrink older messages using a fisheye display. Just as a fisheye camera lens magnifies certain objects while shrinking others, fisheye displays provide local detail about certain objects while still showing the global context in the entire data space [3, 11]. The user can click anywhere on the navigation bar to change the focus. The region in focus appears in the list view.
- The list view shows a portion of the names represented in the navigation bar. A black line next to the navigation bar indicates the graphical representation of the names shown in the list view. By including the textual representation of the names as well as the graphical one, the user eventually should be able to recognize some names in the summary view by pattern alone. The user can select a name by clicking on it in either the summary or list view, and the change is reflected in the other views.
- The message view shows information about the currently selected message. This information might include a title, author, subject, or a short summary. To obtain more information, the user can spawn a reader or browser.

The confluent zoom concept works well with Irwin's problem domain. The leftmost views provide a global overview of all resources with minimal effort from the user, while the rightmost views simultaneously can provide a closer look at recent changes of interest.

5. Future work

Input from GVU demo days, beta testers, and presentation responses have indicated a number of ways in which Irwin could provide better and easier access to information resources.

- **Improved information display techniques.** Rather than limit the graphical display style, the user should be able to choose the appearance and position of the message representations. Message attributes such as sender, content, size, and arrival time could be tied to object size, shape, color, or position at the user's discretion.
- **Improved navigation techniques.** Irwin always shows the messages in the resource in a fixed order by arrival time. Sorting and searching capabilities would provide easier access to the information desired by the user. In keeping with our goals for Irwin, these capabilities should require little effort from the user. Possibilities include document lenses [10] or gestural sorts.

- **Extended scope of information access.** Many sources of information require monitoring, including phone messages, images, faxes, shared files, and printer queues. Currently Irwin requires the author to extend their tool to produce the output in a special Tcl/Tk list format ¹. Irwin should be able to accept data in a variety of formats.

6. Conclusions

This paper discussed techniques for monitoring and visualizing the constant flood of data from information resources. We explored the advantages of hypertools, we introduced the concept of a confluent zoom, and we examined methods for graphically encoding textual information. These techniques contributed to Irwin, a compact but powerful interface to information resources. Since Irwin can be configured to use any tools to collect and browse information, Irwin will not become obsolete as tools improve over time. By limiting the size of Irwin while using a confluent zoom display, users can constantly monitor resources without disturbing their work. We continue to explore methods for graphically summarizing the collected information.

Acknowledgments

The first author wishes to thank the Oak Ridge National Laboratory Visualization and Information Sciences Group for their support during the summer of 1995. Thanks also go to John Stasko and Gregory Abowd for their helpful discussions, and to the numerous people who commented on Irwin and reviewed this paper.

7. References

- [1] Benjamin B. Bederson and James D. Hollan. Pad++: A zooming graphical interface for exploring alternate interface physics. In *Proceedings of the 1994 ACM Symposium on User Interface Software and Technology*, pages 17-26, Marina del Rey, CA, November 1994.
- [2] Stephen G. Eick, Joseph L. Steffen, and Eric E. Sumner Jr. Seesoft—a tool for visualizing line oriented software statistics. *IEEE Transactions on Software Engineering*, 18(11):957-968, November 1992.
- [3] George W. Furnas. Generalized fisheye views. In *Proceedings of the ACM SIGCHI '86 Conference on Human Factors in Computing Systems*, pages 16-23, Boston, MA, April 1986.
- [4] William C. Hill and James D. Hollan. History-enriched digital objects: Prototypes and policy issues. *The Information Society*, 10(2), April 1994.

¹See the Irwin Web page at <http://www.epm.ornl.gov/irwin/> for details of such an effort as well as more information on Irwin.

- [5] Balachander Krishnamurthy. *Practical Reusable UNIX Software*. John Wiley and sons, New York, NY, 1995.
- [6] John K. Ousterhout. *Tcl and the Tk Toolkit*. Addison-Wesley, Reading, MA, 1994.
- [7] C. Plaisant, D. Carr, and H. Hasegawa. When an intermediate view matters: A 2d-browser experiment. Technical Report CS-TR-2980, U. of Maryland, October 1992.
- [8] C. Plaisant, D. Carr, and B. Shneiderman. Image-browser taxonomy and guidelines for designers. *IEEE Software*, pages 21-32, March 1995.
- [9] Steven P. Reiss. Connecting tools using message passing in the FIELD program development environment. *IEEE Software*, 7(4), July 1990.
- [10] George G. Robertson and Jock D. Mackinlay. The document lens. In *Proceedings of the 1993 ACM Symposium on User Interface Software and Technology*, pages 101-108, Atlanta, GA, November 1993.
- [11] Manojit Sarkar and Marc H. Brown. Graphical fisheye views. *Communications of the ACM*, 37(12):73-84, December 1994.

[The page contains extremely faint and illegible text, likely bleed-through from the reverse side of the document. No specific content can be transcribed.]

INTERNAL DISTRIBUTION

- | | |
|-----------------------|--|
| 1. T. S. Darland | 24. Central Research Library |
| 2. N. W. Grady | 25. K-25 Applied Technology Li-
brary |
| 3-7. M. R. Leuze | 26. Laboratory Records - RC |
| 8. C. E. Oliver | 27-28. Laboratory Records Dept. |
| 9-13. S. A. Raby | 29. ORNL Patent Office |
| 14-18. T. H. Rowan | 30. Y-12 Technical Library |
| 19-23. R. F. Sincovec | |

EXTERNAL DISTRIBUTION

31. Dr. Fred Howes Office of Scientific Computing, ER-7 Applied Mathematical Sciences Office of Energy Research U. S. Department of Energy Washington, DC 20585
32. Dr. Gary Johnson Office of Scientific Computing, ER-7 Applied Mathematical Sciences Office of Energy Research U. S. Department of Energy Washington, DC 20585
- 33-37. D. S. McCrickard Graphics, Visualization, and Usability Center College of Computing Georgia Institute of Technology Atlanta, GA 30332
38. Dr. David Nelson Director of Scientific Computing, ER-7 Applied Mathematical Sciences Office of Energy Research U. S. Department of Energy Washington, DC 20585
39. Office of Assistant Manager for Energy Research and Development U. S. Department of Energy Oak Ridge Operations Office P. O. Box 2001 Oak Ridge, TN 37831-8600
- 40-41. Office of Scientific & Technical Information, P. O. Box 62, Oak Ridge, TN 37831