SYSTEM-WIDE MULTIRESOLUTION

Robert L. Read, Donald S. Fussell and Avi Silberschatz

Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712-1188

TR-93-04 February 1993

DEPARTMENT OF COMPUTER SCIENCES
THE UNIVERSITY OF TEXAS AT AUSTIN

AUSTIN, TEXAS 78712

System-wide Multiresolution [*]

Robert L. Read[†], Donald S. Fussell[†], and Avi Silberschatz[‡]

[†]Department of Computer Sciences,
University of Texas,
Austin, TX 78712-1188
Email: {read, fussell}@cs.utexas.edu

[‡]AT&T Bell Laboratories
600 Mountain Ave.
Murray Hill, NJ 07974
Email: silber@research.att.com

February 22, 1993

1 Introduction

A major problem of current systems is that the quantity of data outstrips the system's capacity. A number of existing systems address this problem by allowing quality to be traded for performance. For instance, in response to the volume of data generated by audio, still video, and motion video, the JPEG, MPEG, and p×64 [1, 2, 3] lossy compression protocols all offer some inherent scalability of the quality, or *resolution*, of the data. The most important benefit of lossy compression [4] is that the I/O and transmission costs are decreased. For example, Fluent Systems Inc. [5] sells software and hardware that dynamically adjust the resolution—measured in terms of bits/second—of audio/video streams on a LAN in response to changes in the available bandwidth. These applications demonstrate the utility and convenience of the resolution/performance tradeoff for digital representations of continuous functions, such as images and sounds.

Additionally, scalable resolution of geometric data is used in spatial databases [6], geographic information systems [7], and flight simulators. Even more generally, the related technique of *imprecise computation* has been suggested as an approach to meeting real-time constraints on database queries over any data type [8, 9, 10, 11]. Systematic support for scalable quality naturally complements imprecise computation by allowing larger tradeoffs of quality for time. We call systems that support this tradeoff *multiresolution* systems [12].

This paper argues that ubiquitous, systematic support for scalable multiresolution is essential to next-generation applications that access very large quantities of data. These next-generation applications are the focus of great excitement. They include HDTV [13], multimedia systems, geographic, seismology, astronomy, environmental and other scientific databases [14, 15, 16], spatial databases and geographic information systems, virtual reality systems [17], and terabyte-sized databases of traditional data [18].

---

[*] This material is based in part upon work supported by the Texas Advanced Technology Program under Grant No. ATP-024, the National Science Foundation under Grant Nos. IRI-9003341 and IRI-9106450, and grants from the IBM and Hewlett-Packard corporations. Avi Silberschatz is on leave from the University of Texas.

Section 2 demonstrates that current trends in computing technology will increase the importance of multiresolution in the future. In Section 3 we informally define the concept of multiresolution and present two brief examples. Section 4 contends that system-wide support for multiresolution will maximize the benefits provided by the concept. In Section 5 we sketch a tentative plan for research into multiresolution database technology, and mention some research direction in other subdisciplines of Computer Science. We conclude in Section 6 by presenting an ambitious target application, the *Geoscope*, as a motivation of ubiquitous multiresolution.

2 Motivation for Multiresolution

Four current trends in technology will increase the severity of the data volume problem, and therefore the importance of multiresolution in the near future:

- the trend that storage density is increasing faster than access speed,
- the growing use of multimedia and the convergence of entertainment and computer systems,
- the internetworking of heterogeneous hardware components, and
- the construction of massive databases.

The storage density of solid state memory chips, magnetic disks, and optical storage doubles every few years, leading to exponential growth in storage capacity over time [19]. However, only relatively small improvements per year in the access times and throughputs of these devices have been realized and are expected. This trend makes the access and retrieval of data more and more time expensive relative to other time costs in a computing system, such as actual computation on the processor. Thus, the payoff of saving storage access costs by retrieving lower-resolution data is of ever-increasing value.

The growing use of multimedia brings us up against bandwidth limitations. MPEG data streams at 1.5 Megabits/s [2] are expensive relative to current 10 Megabit/s Ethernet and Token Ring LANs. HDTV will contain 8 to 16 times more data than MPEG [13] and will be expensive on fiber networks even at 1,000 Megabit/s. The popularity of audio/video data and the merging of computer and entertainment systems demand multiresolution and resolution management to extract the most utility from any particular network of hardware. Inherently scalable protocols like p×64 address the problem of transmitting audio and video on low-bandwidth lines, such as cellular phone lines.

The number of portable and mobile personal computers networked via such low-bandwidth mechanisms is expected to explode [20]. Computer resources at various nodes of these inter-connected networks will be of increasingly heterogeneous capability, as supercomputers communicate with palm-tops. Much of the traffic on such networks will be audio/video data. Since we do not want to limit the resolution of data manipulated by powerful nodes to that resolution which can be processed by the less powerful nodes, data at different resolutions should be carried on such a network.

Finally, the construction of ever-larger banks of data, exemplified by the Earth Observing System [14], necessitates multiresolution. So much data is being collected that it cannot be examined in detail. While all of the data is precious, only a fraction of it can be computed against or examined by a human at any particular moment. Already, image databases use low resolution "browse images" [21, 22] to allow rapid preliminary examination of the data. After seeing low resolution overviews of data, scientists may then examine a certain feature at maximum resolution.

We believe these trends will necessitate multiresolution across every component of a system and for all kinds of data.

3 Multiresolution

Multiresolution is the concept of viewing data at different levels of information content. The fields of denotational semantics [23, 24] and information theory [25, 26] provide an intuitive and a formal definition of *information content*. We repeat the informal definition of this concept here to provide the reader with the necessary intuitions.

Data describe the real world. Some data are more descriptive than other data. For instance, the daily list of stock volumes, opening prices and closing prices describes market history better than averages and indices computed over many stocks. Similarly, a high quality audio recording is more descriptive of music played than a poor, scratchy recording. The more descriptive data are, the more *information* they contain. We use the term *resolution* synonymously with "information content." Thus, lower-resolution data is less precise and less informative than higher-resolution data. The meaning of data, and hence the notion of resolution, is always application dependent. For example, the form of multiresolution supported by JPEG has no meaning for non-raster data.

The notion of multiresolution has previously been treated more formally [12]. We provide informal examples of two distinct multiresolution data types here. We seek to apply multiresolution concepts to all kinds of data.

Example 1: p×64 [3] is an inherently scalable motion video protocol used for teleconferencing over limited bandwidth lines. It could serve as the basis for a multiresolution data type of "motion-video" consisting of motion pictures at different qualities. A poor quality movie would be "low resolution" and would approximate "higher-resolution" movies of the same film.

This motion-video type, or one based on different protocols, could be used in an on-demand video-server [27] that plays any number of movies to viewers on a digital network. When the number of movies currently being viewed is very high, the server might provide relatively low resolution video, allowing its storage devices and network capability to meet the demand. As the demand eases, higher-resolution would be provided. In this way, the server would manage the resolution provided to the users. □

Example 2: A histogram can be used as a low resolution version of a set. The number of bins determines the resolution of the histogram. When there are few bins and many data are lumped together, the histogram is a low resolution version of the set. Such a histogram can be efficiently encoded in a fraction of the space of the set it represents. When there are a large number of bins, the histogram provides a great deal of useful information about the set, and is higher-resolution. Under certain conditions, a histogram may completely determine the membership of a set, which is the highest resolution possible. A generalization of histograms called the *sandbag* has been suggested as a general purpose approach to multiresolution sets [12]. □

These two types demonstrate a particular property of approximations that is generally true, though not universally obtained in practice.

> If X is a low resolution version of Y, then X requires less space to be represented by a computer than Y.

Accessing a large (high resolution) object requires many accesses to main memory and/or many expensive I/O operations. Multiresolution systems use this general relationship between low resolution data and space to improve performance by computing against lower-resolution data when possible.

4 System-wide Multiresolution is Essential

Every component of a software system should support multiresolution. For example, if rasters are retrieved from a DBMS at various resolutions, but the graphics subsystem can only render rasters at one resolution, no time will be saved. There may be some utility in retrieving a high resolution image from a storage structure and applying an algorithm that reduces its resolution before it is provided to a different software subsystem. However, the entire system will be more efficient if it employs multiresolution storage structures that provide the savings of reduced I/O and memory access.

In order to take advantage of heterogeneity of hardware in networks, every software component in the system must support scalable resolution. In a situation where a service is being sold over a network [27, 28], the utility of the data, and therefore the demand for the data, is based on resolution. The total utility provided to clients (and the income of the server!) is maximized by systems that can both provide and utilize data at different levels of resolution. For instance, one can easily imagine paying to view a film based not only on the popularity of the film, but also on the quality of resolution that one receives. Someone with a slow CPU and limited memory and display capability should see the same data that the rich person with a state-of-the-art computer and fancy display technology sees, "but through a glass, darkly." Certainly, a data vendor wishes to sell data both to the poor and to the rich–though not necessarily at the same resolution or the same price. Multiresolution specifically raises interesting questions of market-based allocation of resources [29], especially network bandwidth.

The more resolution scalable each component of a software system is, the more control the client user or client application can have over the result resolution. This control is essential to the utility of a multiresolution system because different situations have different performance and resolution requirements. The *ad hoc* multiresolution systems in existence today either dictate resolution to the client or offer a very small set of resolution choices. Designing as much resolution scalability into the system as possible allows for easy updgrades as improved individual software and hardware components become available.

This scalability principle extends to federations of autonomous, heterogeneous components. A system that interacts with systems designed by or owned by others should be prepared to offer as much flexibility in resolution as possible, since no assumptions can be made about the requirements or capabilities of the other systems.

System wide scalable resolution is not an automatic advantage of an object-oriented design. Designing a notion of resolution for a particular kind of data in a specific application requires careful consideration of the nature of human or client interaction with that data type and the capabilities of the hardware. For instance, object-oriented design may allow motion video clips to be passed back and forth between subsystems easily, but it cannot guide the construction of a notion of resolution of video clips appropriate for a particular application.

5 A Tentative Research Program for System-wide Multiresolution

System-wide multiresolution motivates work in many systems-oriented subdisciplines of Computer Science, such as graphics, databases, and computer-human interaction (CHI), as well as more theoretical work in data structures and algorithms. Our own research has focused on database technology. After outlining the important multiresolution issues in that area, we suggest some tentative research directions in other fields.

The construction of data retrieval systems that systematically support multiresolution for very large multimedia databases and virtual reality systems is a challenging problem with an enormous payoff. For several reasons, this is properly a database research issue, as opposed to pure research in graphics.

- Systematic multiresolution requires a multiresolution data model that organizes the data.
- The data involved are so voluminous that efficient and comprehensive management of the storage hierarchy is essential.
- Issues of autonomy, security, and efficient concurrent access of these databases are germane.

We have proposed a multiresolution data model [12] that is a framework for further research. We believe the most pressing mutliresolution database research issues are:

- the development of algorithms that implement an efficient multiresolution algebra,
- an extension of “value functions” used in real-time systems that allow users to assign a value to the resolution as well as the timeliness of a query response, and
- a query optimizer for the multiresolution algebra and extended query language.

Although we are working on these problems, research on a larger scale is needed.

The field of graphics has used multiresolution concepts from its beginning. Additional systems-level research into the integration of this technology with database and network technology will be extremely fruitful. Further, we believe cyberspace [30] and virtual reality systems will require similar lossy compression technology for geometric rather than raster based graphic representation, and that this technology also must carefully consider the general ideas of multiresolution transmission and calculation [31, 32].

In terms of multiresolution, the most important CHI research concerns the question:

> How can we provide low resolution data that maximizes usefulness to the user at a minimum cost to the computing resources?

The design of lossy compression algorithms for multimedia data carefully considers human psycho-acoustic and psycho-visual models, so the line between CHI and graphics already has been blurred deeply by the notion of multiresolution. However, the applications of similar notions of lossy compression to other data types, such as maps and solid geometric scenes, has not been fully researched.

In additional to the systems-oriented work already mentioned, some deep, interesting theoretical work can be outlined. Primarily, this involves extending the notion of multiresolution to more data types than spatial representations and digital samplings of continuous functions.

Computing against low resolution data is a special case of computing with limited information. It has long been recognized that computing in the absence of complete information poses special problems, often increasing computational complexity while greatly decreasing precision in the results. However, in a multiresolution environment the programmer designs the intentional loss of information, so that this imprecision, traditionally disadvantageous, may be used advantageously. The cost-saving advantages of performing multiresolution (i.e. low resolution) computation offset the problems caused by imprecision. These savings are obtained for all data types and computing problems to some degree, although, of course, the largest savings are in those kinds of data for which lossy compression has already been explored.

Multiresolution systems will provide the greatest benefit when every data type supports notions of multiresolution. We are currently developing a notion of multiresolution set, called the *sandbag* [33]. The multiresolution set is a fraction of the data types that need to be considered. Some of the general issues involved in the construction of any multiresolution data type are:

- the development of notions of resolution allowing both useful computation on low resolution data and compact representations that obtain the performance advantages of multiresolution,
- the constructions of metrics for the resolution, and
- the constructions of incremental algorithms that efficiently utilize incrementally improving resolution in their inputs.

This is a fun and challenging area to work in because various branches of mathematics, engineering and science may be applicable in surprising ways. For instance, we have been attempting to use information theory to develop a useful metric for the resolution of the sandbag.

6 The Geoscope

In this section we resurrect an architectural vision discarded in 1953 and project it into the future, and use it to illustrate the utility of multiresolution and the research we have suggested. Our long-term plan is to develop the multiresolution technology needed to realize this vision.

R. Buckminster Fuller (1895-1983), scientist, inventor, philosopher, futurist, cartographer, and poet, was very concerned with mapping the earth and understanding global trends. In 1953 he proposed the construction of a 60-meter diameter sphere to be suspended in space next to the United Nations building in New York [34]. This enormous ball was to be a precise simulacrum of the planet Earth, on a such a scale that a tourist could observe on its surface the house in which he or she lived with the aid of powerful binoculars. Colored lights were to be projected onto its surface to show the distribution of the Earth's resources, including its population and money. By changing these lights over time in accordance with the changing geographic distribution of resources, global trends could be graphically displayed on the "Geoscope." This would deepen our understanding of "Spaceship Earth," a term coined by Fuller.

Like many of Fuller's ideas, the Geoscope is still ahead of our time, and the 60-meter version has not yet been constructed. However, computer technology now allows us to contemplate the implementation of a Geoscope in software. A software Geoscope accessed by portable personal computers connected via cellular phones has the advantage that any number of people can observe it at the same time. The precision of such a Geoscope is limited only by technology and the available data. For instance, the floor plan of every building in Paris might be accessible, and the traffic patterns of Los Angeles projected through it. If the appropriate historic data is available, it can simulate the past as well as the present. For instance, the flow of money, material, and men associated with World War II could be graphically depicted.

The implementation of the Geoscope, like many less ambitious projects, benefits from comprehensive multiresolution. Since a Geoscope will use raster data in the form of satellite photos, spatial data in the form of maps, and general data to describe the flux of money, resources, and people, it is an excellent target application. If a continent is viewed, satellite photos provide much greater resolution than the human eye can perceive, so it is clearly reasonable to construct lower-resolution images of large data objects, such as countries. Similarly, although every road in the world may be available in the Geoscope, a view of even a small city may be too busy unless a low resolution map showing only the main arteries is displayed. A somewhat different form of low resolution is required for describing resources and population. Although a database of the over five billion people may be constructed, there is little point in locating them all precisely for some applications. At least one approach to this kind of resolution has been suggested [12]. The transmission of these three different kinds of data over limited bandwidth lines to and from computers of greatly varying capacities demands multiresolution.

The applications of the Geoscope boggle the mind. It will be the ultimate Atlas. Who will buy real estate without consulting it? The teaching of history, anthropology, ecology, and, of course, geology and geography will be revolutionized. For instance, a user of the Geoscope will be able to walk through ancient Jerusalem before the Crusades, Rome at the time of Nero, or the Library of Alexandria before its destruction. The Geoscope will be a Modern Wonder of the World – and may contain all the Ancient Wonders in their ideal condition.

The Geoscope as we have described it is many years away. However, similar but less ambitious projects requiring multiresolution are imminent. Foreseeable trends in hardware technology will continue to increase the demand for multiresolution technology. We therefore conclude that systematic support for multiresolution is critical to next-generation information systems.

Acknowledgements

Sarah Chodrow, Mary C. Reed, Bert Rothenbach, and Thomas Woo provided helpful comments. Michael Benedikt and the writings and life of Buckminster Fuller were inspirational.

References

[1] Gregory K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):31–44, April 1991.

[2] Didier Le Gall. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):47–58, April 1991.

[3] Ming Liou. Overview of the p×64 kbit/s video coding standard. *Communications of the ACM*, 34(4):60–63, April 1991.

[4] M. Nelson. *The Data Compression Book.* M & T Publishing Inc., 501 Galveston Drive, Redwood City, CA 94063, 1991.

[5] Fluent Systems, Inc. 594 Worchester Rd., Natick, Massachussetts, 01760-1827.

[6] Hanan Samet. *The Design and Analysis of Spatial Data Structures.* Addison-Wesley, 1989.

[7] W. E. Huxhold. *An Introduction to Urban Geographic Information Systems.* Oxford University Press, 200 Madison Avenue, New York, NY 10016, 1991.

[8] S. V. Vrbsky and J. W. S. Liu. An object-oriented query processor that returns monotonically improving answers. In *Proceedings of the 7th IEEE Conf. on Data Engineering*, Kobe, Japan, April 1991.

[9] K. Liu and R. Sunderraman. On representing indefinite and maybe information in relational databases. In *Proceedings: The Fourth International Conference on Data Engineering*, pages 250–257, February 1988.

[10] K. J. Lin, S. Natarajan, and J. W. S. Liu. Imprecise results: utilizing partial computations in real-time systems. In *Proceedings of the IEEE 8th Real-Time Systems Symposium*, San Jose, California, December 1987.

[11] K. B. Kenney and K. J. Lin. Structuring large real-time systems with performance polymorphism. In *Proceedings of the IEEE 11th Real-Time Systems Symposium*, pages 238–246, Orlando, Florida, December 1990.

[12] Robert L. Read, Donald S. Fussell, and Avi Silberschatz. A multi-resolution relational data model. In *18th International Conference on Very Large Data Bases*, pages 139–150, Vancouver, Canada, August 1992.

[13] Karen A. Frenkel. HDTV and the computer industry. *Communications of the ACM*, 32(11):1300–1312, November 1989.

[14] Jeff Dozier. Access to data in NASA's Earth Observing System. In *SIGMOD Conference Proceedings*, page 1, 1992. Keynote Address.

[15] A. Shoshani and H. K. T. Wong. Statistical and scientific datbase issues. *IEEE Transactions on Software Engineering*, pages 1040–1046, October 1985.

[16] M. A. Bassiouni. Data compression in scientific and statistical databases. *IEEE Transactions on Software Engineering*, pages 1047–1058, October 1985.

[17] Ken Pimentel and Kevin Teixeira. *Virtual Reality: Through the new looking glass.* Windcrest Books, 1993.

[18] A. Silberschatz, M. Stonebraker, and J. Ullman. Database systems: Achievements and opportunities. *Communications of the ACM*, 34(10):110–120, October 1991.

[19] J. L. Hennesy and D. A. Patterson. *Computer Architecture: A Quantitative Approach.* Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1990.

[20] T. Imielinski and B. R. Badrinath. Querying in highly mobile distributed environments. In *18th International Conference on Very Large Data Bases*, pages 41–52, Vancouver, Canada, August 1992.

[21] Personal communication with Dr. Anita Cochron about the Halley Watch Archive CD image library.

[22] A. Turtur, F. Prampolini, M. Fantini, R. Guarda, and M. A. Imperato. IDB: An image database system. *IBM Journal of Research and Development*, 35(1/2):88–96, January/March 1991.

[23] P. Buneman, A. Jung, and A. Ohori. Using powerdomains to generalize relational databases. *Theoretical Computer Science*, pages 23–55, 1991.

[24] J. E. Stoy. *Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory.* MIT Press, Cambridge, Massachusetts, 1977.

[25] H. L. Resnikoff. *The Illusion of Reality.* Springer-Verlag, 1989.

[26] R. W. Hamming. *Coding and Information Theory.* Prentice-Hall, Englewood Cliffs, NJ, 07632, 1986.

[27] P. Venkat Rangan, Harrick M. Vin, and Srinivas Ramanathan. Designing an on-demand multimedia service. *IEEE Communications Magazine*, pages 56–64, July 1992.

[28] Joe Sutherland and Larry Litteral. Residential video services. *IEEE Communications Magazine*, pages 36–41, July 1992.

[29] C. A. Waldspurger, T. Hogg, B. A. Huberman, J. O. Kephart, and W. S. Storne. Spawn: A distributed computational economy. *IEEE Transactions on Software Engineering*, 18(2), Feb. 1992.

[30] Michael Benedikt, editor. *Cyberspace: First Steps.* Cambridge, MIT Press, 1991.

[31] L. Bergman, H. Fuchs, E. Grant, and S. Spach. Image rendering by adaptive refinement. *Computer Graphics*, 20(4):29–38, August 1986.

[32] M. F. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenberg. A progressive refinement approach to fast radiosity image generation. *Computer Graphics*, 22(4):75–84, August 1988.

[33] R. L. Read, D. S. Fussell, and A. Silberschatz. The fact set approach to the sandbag imprecise set representation and its complexity. Submitted to *PODS*.

[34] R. Buckminster Fuller. *Critical Path.* St. Martin's Press, 1981. Chapter 5, Part II.