



Calhoun: The NPS Institutional Archive
DSpace Repository

Faculty and Researchers

Faculty and Researchers Collection

1998-01

Multimedia Systems by Neil C. Rowe

Rowe, Neil C.

Monterey, California. Naval Postgraduate School

<http://hdl.handle.net/10945/36950>

Downloaded from NPS Archive: Calhoun



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>

Multimedia Systems

Neil C. Rowe

U.S. Naval Postgraduate School

Multimedia systems combine the digital form of images, graphics, audio, electronic signals, or video with traditional text data. Multimedia systems provide many fruitful applications for distributed and parallel processing for several reasons. First, multimedia data can be bulky: A traditional television picture has almost a million bytes of data, and video needs at least 24 of those per second. Thus even simple operations on data can significantly benefit from smarter processing methods. Second, many important multimedia applications like video delivery have difficult real-time constraints. Third, multimedia data is often easily partitionable for processing.

Intramedia and intermedia parallel and distributed processing

Much multimedia processing lends itself well to parallel methods (see Parallel Algorithms and Multimedia Data Objects). For images for instance, the operations of rotation, contrast enhancement, finding of sharp contrast boundaries, and searching for particular shapes can be done in parallel on the different parts of the image because of their basically local nature [5]. Similar partitioning can be done for audio and video with respect to time: Frequency or color analysis at one time is generally independent of that at another. Compression and decompression methods like JPEG for images and MPEG for video [2] provide good opportunities for parallelism because of their standardization and importance in making networks efficient. Parallelism is especially valuable for ray-tracing methods in rendering graphics for three-dimensional scenes because the paths of individual light rays do not interact.

Distributed processing can also be appropriate between the different media comprising a multimedia data object because each has its own appropriate compression and processing methods. For instance, television has a video track, an audio track, often a closed-caption track, and an overall textual description; each can be considered a separate data "stream." The speedup possible in distributed processing between media is usually small because of the considerable speed differences: Text processing is much faster than audio processing, and audio is much faster than video. Larger speedups are more commonly obtained by distributed processing of different tasks within media, as when each of 1000 processors searches for one of 1000 shapes within an image in parallel.

Multimedia delivery

The main problem with multimedia data is just delivering it where needed [1, 3]. Because of its bulkiness, multimedia is often best archived at a few sites and retrieved from there. That can entail logistical problems in delivering all those bits efficiently since networks are generally optimized for text data. Nonetheless, important applications for multimedia delivery systems are being explored, including digital multimedia libraries (see Digital Libraries), video meetings (see Videoconferencing), video medicine (see Telemedicine), distance learning, multimedia mail, and interactive television.

Multimedia delivery is typically intensive in input/output and communications resources as opposed to processor resources. Network architectures and software need to be optimized for large transmission "bandwidths" (bits per second). This has several implications:

1. System architectures should be chosen for fast input/output (see Input/Output). Parallel ports are desirable (although not currently common).
2. Star and fully connected topologies are desirable (see Network Topologies). That may only be feasible with local-area networks for many applications.

3. Switches should be preferred to routers on network connections since the routers have lower bandwidths and higher delays (see Networking Technologies).
4. Experimentation with the packet size for multimedia data may improve performance since the best size is hard to predict.
5. Data-compression techniques should be used where possible to minimize the necessary bandwidth; typical compression ratios range from 2:1 for audio to 20:1 for images and 50:1 for video.
6. Caching (see Cache Memories) of frequently-used data can help efficiency (as when data is multimedia Web pages (see World Wide Web), some of which are much more popular than the others), but not all multimedia applications so benefit. A simple and popular alternative to caching is a CD-ROM optical disk at the end-user site containing commonly-needed media data.

Real-time multimedia delivery

Many multimedia applications have real-time constraints, such as those delivering audio and video [4] and those supporting real-time simulations. But bandwidth over networks is a problem. A single MPEG-1 compressed video of current television-picture quality needs around 2 megabits per second (though videoconferencing and video interviews can be adequate with less); compact-disk music audio requires 1.4 megabits/second. Standard (ISDN) telephone lines provide 0.064 megabits/second, nowhere near adequate for either (though several lines can be used in parallel, what is called "bonding"). T-1 lines can improve this to theoretically 1.5 megabits/second, but that is still inadequate for video. Ethernet connections run theoretically 10 megabits/second but are only appropriate for local-area networks. Several new network technologies such as the increasingly popular ATM (see Asynchronous Transfer Mode) should provide still higher bandwidths, but they may still be insufficient for video if heavily loaded. One could download media completely before "playing" them so that bandwidth is less important, but that requires a large storage capacity.

Thus without careful design there can be serious problems in quality of service (see Quality of Service) in real-time multimedia delivery, problems suggesting negotiation both beforehand and in real time. Generally speaking, worst-case or deterministic metrics for quality of service are more important than average-case or statistical ones for real-time multimedia since users have absolute acceptability limits. For instance, video becomes unwatchable if its frame rate is too uneven, and audio is even more sensitive.

Transmission bandwidth is bounded by the bandwidth with which multimedia data can be supplied from an archive. Real-time data like live video may be able to bypass storage and go directly onto the network as fast as the network can handle (which also can save considerable storage: A two-hour movie needs around 2,000 megabytes). But in general the multimedia data must be archived. It must be in blocks for efficient memory management, though the blocks can be larger than those typical for traditional data. Magnetic tape and optical disks are less flexible in manipulating blocks than magnetic disks, so the latter is preferable for multimedia applications. The seek time of the magnetic disk head between blocks can significantly lower attainable average bandwidths, so special disk-allocation schemes for multimedia have been proposed. Successive blocks can be put on different disks so that a block from one disk can be transmitted during the seek time for a block from another ("striping"). Figure 1 shows an example where an image is striped in four pieces across two archival sites, then combined at the delivery site. Block transfers themselves are not usually a bottleneck since magnetic disks in 1998 can deliver data at 40 megabits per second and this rate is steadily increasing every year.

Archive site 1



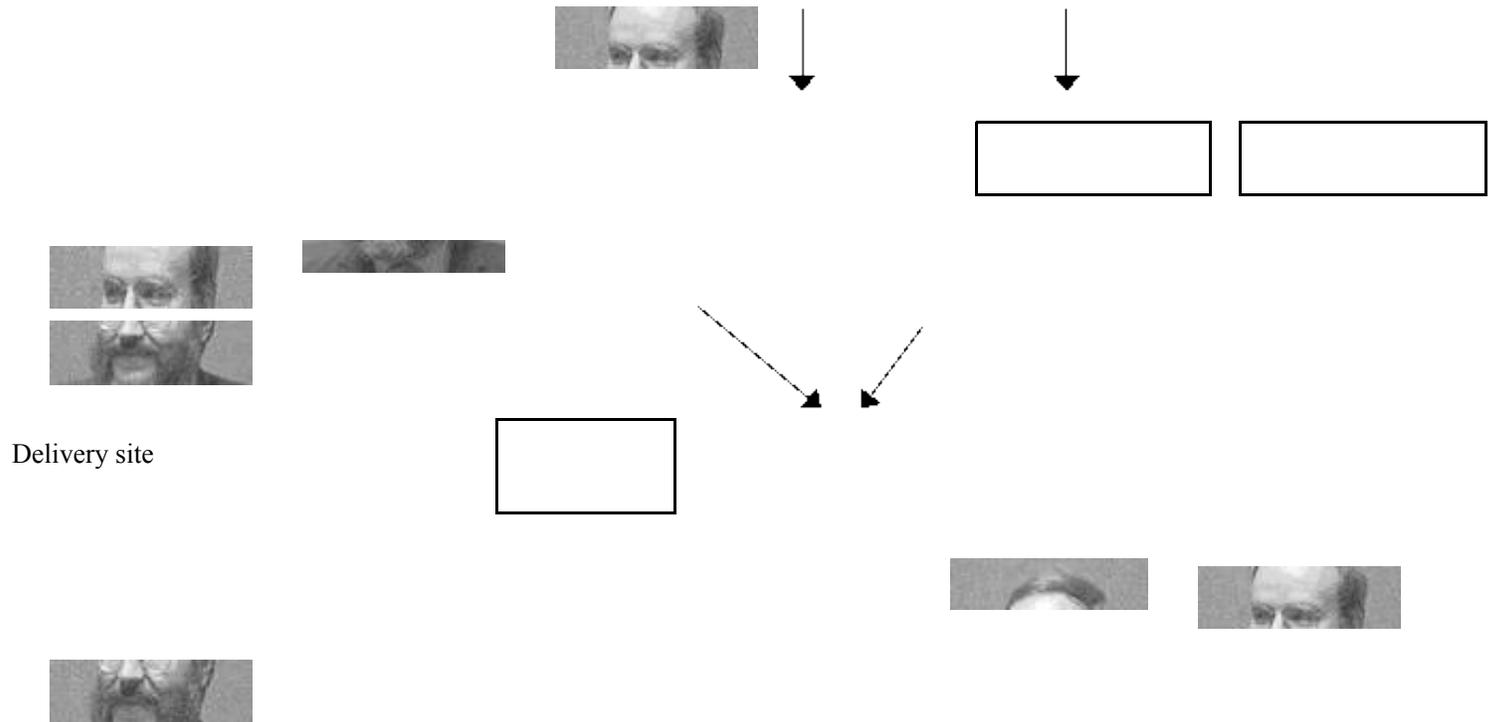


Figure 1: Example of striping and buffers for image-data delivery.

Network transmission delay (especially important for real-time interactive applications like videoconferencing) can be aided by a good network routing. Use of multiple paths relaying multimedia data from source to destination can reduce the effect of any one bottleneck. If it is important that data not be lost in case of loss of a node, redundant data can be sent over the multiple paths. But many multimedia applications like video playback are tolerant of occasional data loss.

Another transmission issue is the evenness of the arrival of data at a destination site ("jitter"), which for bulky data is significantly affected by the network load. Too-large jitter causes jumpiness in the output. In addition, the variation in data volume sent per unit time ("burstiness") may be considerable, as when data redundancy changes for compressed data when a static video scene suddenly becomes active. Video and audio need uniform speeds to work correctly (usually video jitter delays cannot exceed 0.1 seconds and music audio delays 0.0001 seconds), so this is a key quality-of-service issue. Accumulation of data in storage buffers along the transmission path is the traditional remedy for jitter and burstiness (and also data packets received out of order), but effective multimedia buffers must be big. Figure 1 shows an example of buffers accumulating at intermediate nodes as well as the delivery node for transmission of an image; only the upper two pieces and part of a third piece of the image have reached the delivery site. Overall multimedia transmission delay is often unimportant (we do not care whether a movie takes one second or one minute crossing the network as long as this remains constant), so output presentation to the end-user can wait until big buffers are reasonably full. If the transmission rate is too fast for the end-user and buffers are overflowing, the system can drop frames of video or short segments of audio, evenly spaced ones to minimize the jitter, or order the archive to slow the rate if possible. If the transmission rate is too slow and buffers are near-empty, the archive can be ordered to reduce the number of bits transmitted for each image location (pixel) or audio-signal height, to provide reduced-quality output at a proper speed; or ordered to reduce the number of data items per unit time, by reducing the size of an image or the sampling rate of audio. When multimedia data structures are partitioned for efficiency, there are also difficult issues in coordination of separate data streams at a destination (see Synchronization, Multimedia).

Object-oriented distributed multimedia systems

Object orientation can mean several things for distributed multimedia systems. It can refer to a programming or query language

supporting multimedia. A variety of abstractions are helpful, including classes for different media types and classes distinguishing storage, transmission, and presentation media. Object orientation can also refer to tools for performance analysis (see Performance Modelling) and design of complicated multimedia presentations. For instance, an artist who wishes to create a show with coordinated text, audio, and video needs to specify something like a Petri Net (see Petri Nets) involving the objects, and some tools provide this.

Object orientation can also mean the multimedia data itself is object-oriented, as in distributed simulation systems (including virtual realities) where software objects represent physical objects in the world (see Object Computing, Distributed). Then network data traffic is primarily parameterized updates to software objects (like vehicles in a combat simulation) rather than whole scenes (though there still may be image traffic of things like faces that are not easily parameterized). This results in a cleaner software design as well as less data traffic (as an intelligent form of compression), but requires more work assembling a scene. Distributed simulation systems differ from other multimedia applications in that they often can use multicasting (see Multicast Communications Systems), unreliable protocols (since information will be updated again shortly), and protocols with minimal network delays (especially when interactive).

References

- [1] --, Special Issue on Distributed Multimedia Systems and Technology. *IEEE Journal on Selected Areas in Communications*, 14 (7), September 1996.
- [2] S. M. Akramullah, I. Ahmad, and M. L. Liou, A Data-Parallel Approach for Real-Time MPEG-2 Video Encoding. *Journal of Parallel and Distributed Computing*, 30, 129-146, 1995.
- [3] F. Flickner, *Understanding Networked Multimedia: Applications and Technology*. Hemel Hempstead, UK: Prentice-Hall International, 1995.
- [4] D. Jadav and A. Choudhary, Designing and Implementing High-Performance Media-on-Demand Servers. *IEEE Parallel and Distributed Technology*, 3 (2), 29-39, Summer 1995.
- [5] I. Pitas (ed.), *Parallel Algorithms for Digital Image Processing, Computer Vision, and Neural Networks*. Chichester, UK: Wiley Professional Computing, 1993.

Cross Reference

Computer Supported Cooperative Work see Multimedia Systems.

Digital Libraries see Multimedia Systems.

Multimedia Data Objects see Multimedia Systems.

Simulation and Modeling see Multimedia Systems.

Synchronization, Multimedia see Multimedia Systems.

Telemedicine see Multimedia Systems.

Videoconferencing or Distributed Office see Multimedia Systems.

World Wide Web see Multimedia Systems.