Requirements for Multicast in a Technical Workstation Environment

Bert J. Dempsey and Alfred C. Weaver

Computer Science Report No. TR-90-04
March 19, 1990

Requirements for Multicast in a Technical Workstation Environment

Bert J. Dempsey and Alfred C. Weaver

Department of Computer Science
Thornton Hall
University of Virginia
Charlottesville, Virginia 22903
(804) 924-7605
bjd7p@virginia.edu, weaver@virginia.edu

Introduction

From a survey of the technical literature and conversations with knowledgeable professionals, we identified seven requirements for a Transport Layer multicast in the Technical Workstation Environment:

- group querying and reporting
- reliable, fast communication for distributed processing
- concurrency control for distributed databases
- efficient voting algorithms for server pools
- collaborative development
- rapid distribution of real-time digital images
- distributed synchronization

These requirements are each discussed in turn below.

1. Group Querying and Reporting

Requirement: A multicast capability is required to shield user applications and system monitors from knowledge of the physical location of resources available on the network.

Multicast addressing provides location-independent addressing by allowing for a run-time binding of a logical grouping of servers (a functional group) with actual physical resources. For example, a diskless workstation may use, instead of a hard-wired unicast address, a multicast address for the group of boot servers [CHER85]. The multicast request solicits a number of responses, which are serialized by the network medium, and the workstation uses the first response. The number and location of the servers are unknown at the workstation and possibly changes with time.

In the Technical Workstation Environment we see a trend of increasing system management functionality, of which network management represents a subset. From the system management console, physical devices will be interrogated and parameters set. This management facility will require a Transport Layer multicast as broadcasting becomes increasingly expensive due to a growing number of management activities and network nodes. Multicasting enables a de-coupling of network addressable entities and their physical locations such that these management facilities can distribute information to and collect information from a changeable, and possibly unknown, set of distributed processing entities.

2. Fast, Reliable Communication for Distributed Processing

Requirement: Reliable multicast mechanisms are required to simplify and facilitate group communication within distributed and parallel applications.

In the Technical Workstation Environment we identify a need to support distributed and parallel applications that have, until recently, relied upon point-to-point remote procedure call (RPC) communication [BIRR84]. Recent work has focused on a two-step delivery mechanism whereby all multicast messages to a group are sent first to a central *sequencer* node that orders the messages and reliably sends them to the group's members. When a process needs to distribute a global update to more than one other process, this reliable multicast facility can achieve lower message latencies than RPC and transparently mask communication failures [KAAS89].

3. Concurrency Control for Distributed Databases

Requirement: A multicast capability is required to support a simple, but powerful, concurrency control mechanism for distributed databases.

In distributed database applications, reliable multicast mechanisms that use a central sequencing node to order multicast messages provide, as an alternative to locking mechanisms or more complex timestamping mechanisms, a pseudo-timestamp for the purposes of concurrency control. Also, in very large databases where searching is computationally intensive (as might be expected of future databases in the life sciences and as exist in certain poorly indexed archival databases today (e.g. the patent office)), a sequencing node performing reliable multicast would enable a horizontal decomposition of the database for parallel searching on multiple workstations. In this case parallel searching proves very valuable since (1) it bounds search time since sections are of fixed size; (2) it eliminates the need for mainframes to handle very large databases and thus pulls these databases into the realm of networked workstations; (3) it allows for tuning the search process by adjusting the size of sections and duplicating often-searched sections [TSEU89].

4. Efficient Voting Algorithms for Server Pools

Requirement: A multicast capability is required to exploit effectively pools of distributed processors for dynamic load balancing or fault-tolerant programming.

In the Technical Workstation Environment we identify a need for efficient and effective mechanisms to allow pools of clients to share pools of servers [NI85] [STAN84]. A multicast facility allows server-driven scheduling by having each server multicast its current status to the pool of clients, each client send its tasks, and the server select a client. Alternatively, in client-driven scheduling, a client multicast its needs to the servers, available servers respond with their loads, and the client selects a server [LIAN90]. Without a multicast facility, the posting of clients' jobs or servers' loads (that is, contract negotiation) will be prohibitively slow in many situations. Cooper has shown how pools of servers can underpin fault-tolerant programming using replicated procedure modules and that a multicast communication facility would dramatically improve the efficiency of this scheme [COOP85].

5. Collaborative Development

Requirement: A multicast capability is required for a collaborative development environment.

In the Technical Workstation Environment we identify a need for multicast in support of substantial increases in connectivity. A proposal being studied by the National Science Foundation for a National Collaboratory foresees the need for a very rich interconnection between multi-disciplinary scientists in order to accelerate the pace and quality of research projects such as mapping the human genome and global change

[WULF88]. In the realm of application development tools, plans are now underway to move up the next step from distributed software development to *collaborative development* in which a number of contractors spread over a wide area will interact daily in the concurrent planning and developing of large software projects. This new software development environment will require multicast in at least three ways. First, there must be rapid file sharing among a number of physically dispersed sites. Second, the substantial increase in the total number of nodes on which project resources will reside will have a dramatic impact on Directory Services. In particular, the need for inquiries to distributed name and route servers will rise. Thirdly, collaborative development will require on-line electronic conferencing and electronic mail distribution lists to which interested parties can subscribe. Both of these applications are most naturally supported by a multicast mechanism. Existing projects such as Grapevine [BIRR82] and Enchere [BANA86] represent first steps toward designing powerful distributed systems that provide the full range of services required for collaborative development.

6. Rapid Distribution of Real-Time Digital Images

Requirement: A multicast capability is required to support rapid multidestination distribution of graphics images.

In the Technical Workstation Environment we identify a need to move large (> 1 Megabyte) files, in particular digital images, rapidly to multiple users. In shipboard or ground-based command and control environments, signal processing techniques are applied to raw data from sensors and the processed data distributed across high-performance networks to display workstations for human operators [COHN89]. In

[MARL89] a scenario depicting the needs of future Navy platforms, specifically a Tactical Console Display subsystem, is discussed in detail. Here twenty display workstations receive multiple data streams, one being a periodic update of the ship's primary track file in which various types of sensor data have been merged. The real-time constraints present in a command and control environment make multicasting crucial since time does not permit a series of unicasts.

7. Distributed Synchronization

Requirement: A multicast capability is required for global synchronization in certain simulations.

Within the parallel discrete-event simulations community, certain simulation strategies call for explicit synchronization points between logical processes wherein the global synchronization requires communication [NICO90]. The development, testing, and analysis of these simulation techniques in a Technical Workstation Environment requires a multicast facility. Otherwise, the communications overhead of using a series of point-to-point messages to negotiate at synchronization points will dominate the simulation's behavior and distort analysis.

REFERENCES

BANA86 Banatre, J.P., Banatre, M., Lapalme, G., and Ployette, F., "The Design and Building of Enchere, A Distributed Electronic Marketing System", *Communications of the ACM*, Volume 29, Number 1, pp. 19-29, January 1986.

BIRR82 Birrell, Andrew, Levin, R., Needham, R., and Schroeder, M., "Grapevine: An Exercise in Distributed Computing", *Communications of the ACM*, Volume 25, Number 4, pp. 260-274, April 1982.

BIRR84 Birrell, Andrew, and Nelson, B.J., "Implementing Remote Procedure Calls", *ACM Transactions on Computer Systems*, Volume 2, Number 1, pp. 39-59, February 1984.

CHER85 Cheriton, D. and Deering S.E., "Host Groups: a Multicast Extension for Datagram Internetworks", *Proceedings of the Ninth Data Communications Symposium*, IEEE/ACM, Whistler Mountain, BC, Canada, pp. 172-179, September 1985.

COHN89 Cohn, Marc, "Functional Addressing: Another Way of Looking at Multicast", *TRANSFER*, pp. 13-15, November/December 1989.

COOP84 Cooper, E.C., "Circus: A Replicated Procedure Call Facility", Fourth Symposium on Reliability in Distributed Software and Database Systems, 1984.

KAAS89 Kaashoek, M.F., Tanenbaum, A.S., Hummel, S.F., and Bal, H.E., "An Efficient Reliable Broadcast Protocol", *Operating Systems Review*, Volume 23, Number 4, October 1989.

LIAN90 Liang, L., Chanson, S.T., and Neufield, G.W., "Process Groups and Group Communications: Classifications and Requirements", *IEEE Computer*, Volume 23, Number 2, pp. 56-66, February 1990.

MARL89 Marlowe, David, "Requirements for a High Performance Transport Protocol for Use on Naval Platforms— Revision 1", PEI Document 89-104, July 1989.

NI85 Ni, L., and Hwang, K., "Optimal Load Balancing in a Multiple Processor System with Many Job Classes", *IEEE Transactions on Software Engineering*, Volume SE-11, Number 5, pp. 491-496, May 1985.

NICO90 Nicol, David, "Analysis of Synchronization in Massively Parallel Discrete-Event Simulations", *SIGPLAN Symposium 1990*, March 1990 (upcoming).

STAN84 Stankovic, J.A., "Simulations of Three Adaptive, Decentralized Controlled, Job Scheduling Algorithms", *Computer Networks*, Volume 8, Number 3, pp. 199-217, June 1984.

TSEU89 Tseung, L.C.N., "Guaranteed, Reliable, Secure Broadcast Networks", *IEEE Network*, pp. 33-37, November 1987.

WULF88 Wulf, William, "The National Collaboratory— A White Paper", NSF, December 1988.