to appear in
Proceedings of the Eighth IEEE Distributed Simulation – Real Time Applications Workshop, 2004

Implementation of Host-based Overlay Multicast to Support of Web Based Services for RT-DVS

Dennis M. Moen, J. Mark Pullen and Fei Zhao
George Mason University
{dmoen,mpullen,fzhao}@gmu.edu

Abstract

Growing demand for use of Internet/Web-based services in real-time distributed virtual simulation (RT-DVS) and other real-time applications is fueling extensive interest in overlay multicast protocols. These applications demand Quality of Service (QoS) and many-to-many multicast services that are not available in underlying Internet services today. This paper describes an early implementation of an overlay multicast protocol designed to support many-to-many multicast for RT-DVS applications called Extensible Modeling and Simulation Framework Overlay Multicast (XOM). We first describe the architecture and key design considerations of XOM. We then provide preliminary results from lab experiments with our prototype. Our results indicate that we can achieve performance objectives for support of web-based services for RT-DVS.

1. Introduction

Recent increased interest in Web based distributed simulation has prompted examination of the adequacy of existing network protocols to support the demanding data distribution needs of this approach. In particular, providing reliable multicast services to real time distributed virtual simulation (RT-DVS) is an important requirement to enable use of Web based services for RT-DVS across open networks such as the Internet. These services must include support for quality of service (QoS) for reliability and bounded latency and also support for many-to-many multicast communications. Internet multicast protocols have been developed over several years to support group communications. Historically, these protocols have focused on supporting applications, such as streaming audio and video that typically have one-to-many type data distributions. These early protocols have exhibited limitations in support of more demanding types of applications [1] and have proved too complex in operation for acceptance by most network operators. For RT-DVS using Web services, available network services are inadequate to support the many-to-many multicast requirement as well as their need for improved QoS.

RT-DVS applications using visual space management in real-time distributed simulation and supporting communications systems are evolving to Web based services using XML tagged objects. This new Web based services model is called the Extensible Modeling and Simulation Framework (XMSF) [2]. The facilities and performance provided by underlying networks represent an important constraint in deployment of XMSF. As an example, the throughput required for communicating object status updates is potentially extremely large, consisting of thousands of state updates per second from simulation objects. These object updates typically have message sizes of at least 110 bytes without tag or other protocol overheads. Web-based RT-DVS are run on heterogeneous sets of workstations with differing processing and display capabilities, traveling over a heterogeneous network with capacities varying by many orders of magnitude between the initial connecting link and the slowest end user. We have therefore expanded our previously proposed approach [3] to meet the needs of XMSF and given it the name XMSF Overlay Multicast (XOM). The XOM overlay network concept is presented in figure 1. We propose the XOM as an overlay multicast protocol designed to support efficient, reliable multicast transmissions on top of existing network protocols such as UDP/IP. It is based on the notion of a multicast relay agent (XOM) in each subnet, which uses unicast protocols to communicate with its peers on other subnets and controls all aspects of the subnet's group communications for a given multicast group. The XOM provides reliable services over UDP by offering two classes of services, urgent and best-effort, and by actively managing the end-to-end path stability and delay bounds between XOMs.

2. Network requirements for RT-DVS

Widely deploying RT-DVS across many organizations with large numbers of applications requires robust multicast networking services that are transparent to end users. Our approach to XOM recognizes that underlying networks may have a wide range of network capacities and may include wireless media and low capacities to modern broadband networks operating at gigabit speeds. We also recognize that, as RT-DVS applications move toward advanced

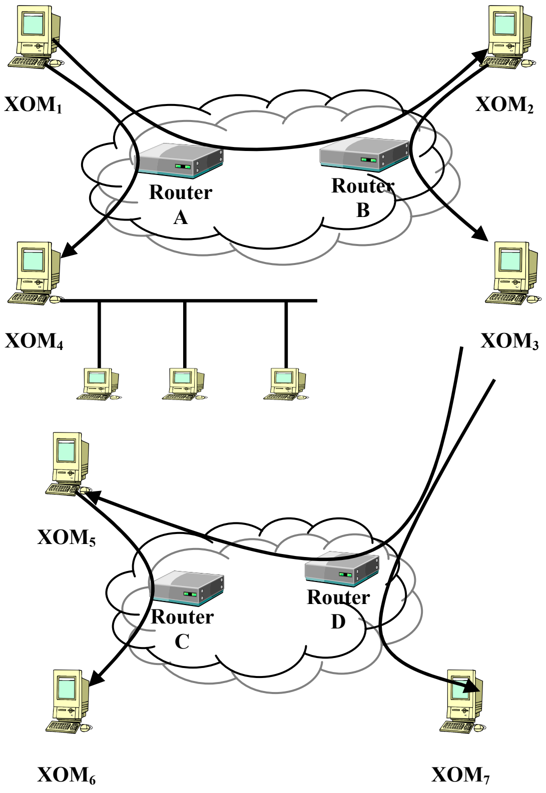


Figure 1. XOM overlay

technologies such as XML-oriented Web services and agent-based distributed simulations [2], there will be a growing need for advanced networking services that are not likely to be available in all parts of the Internet [3]. These new services also must accommodate higher layer services designed to improve data management such as those described by Boukerche [4], Morse [5], and data matching algorithms proposed by Tan [6].

As an example, the interest management (IM) approach described by Wang [7] is used in RT-DVS for efficiently matching entity interests with data. Wang proposes the idea of entities subscribing to a communications channel based on interest. Our approach to XOM provides this "channel" type communication service as part of the XOM multicast service. The XOM uses a convention of a source S and a group G to describe a communications channel or path, (S, G). A user subscribes to a data source S by joining a group G. The XOM manages efficient delivery of the data to all users via overlay multicast. An instance of the XOM on the LAN hosting the agent application provides the service.

A second major goal is to use Web based services with XML data models [8] to not only make the RT-DVS application environment more open, but expand the user base through standards based data exchange. This is exemplified by the XMSF [2] and associated data models such as the Command and Control Information Exchange Data Model (C2IEDM) [9]. These initiatives are dependent on a multicast network service for group exchange of datagrams with real-time response and predictable guarantees of service quality.

3. XOM architecture

The XOM solution must provide the RT-DVS real-time response and predictable network services in order for the end simulation systems to interact within specific delay bounds. Simulation users deployed across the Internet and/or Intranets require low latency, including stringent jitter requirements and high capacity demands. In this section, we address design factors in our approach to XOM.

The architecture for the XOM presented in figure 2 is designed to be highly modular so that module optimization and alternative strategies for each module can be prototyped for evaluation. The proposed approach for the XOM overlay is to use the Internet User Datagram Protocol (UDP) as the underlying network protocol and offer services to the application layer for two different classes of services: Class 0:best effort and Class 1:urgent traffic.

We employ a priority queuing strategy to give precedence to class 1 traffic and mark class 0 traffic as "discard eligible" in the event of network congestion. This approach is consistent with our previous efforts in development of the Selectively Reliable Multicast Protocol (SRMP) [10, 11]. Since simple precedence does not provide error control, any form of desired error control must be added to the client application. The design assumption is that messages are relatively small (less than 120 bytes) and the underlying network is able to deliver message guarantees greater than 99% and has reasonable routing path stability on the order of minutes. Reliable transport can also be provided using such protocols as SRMP, shown in figure 2 as an example interface, where a more desirable reliability is sought but not available to the client application.

Allocation of a specific XOM IP address, or network service access point, is not considered part of the XOM protocol and requires the use of an outside addressing/registry authority to establish an XOM host. The registry maintains a list of all XOMs in the overlay and registered RT-DVS application users. This use of an external registry provides a level of security by establishing an authority that authorizes a source to be a sender which can be used by networked XOMs receivers for recognition of authorized senders in the network. The registry also will maintain the public routable IP addresses of all active XOMs and subsequently used by the XOMs to establish efficient overlay multicast paths between XOMs, and will maintain multicast group membership information. Once an XOM host is established, internal protocol mechanisms provide for path optimization between XOM hosts and manage multicast group membership.

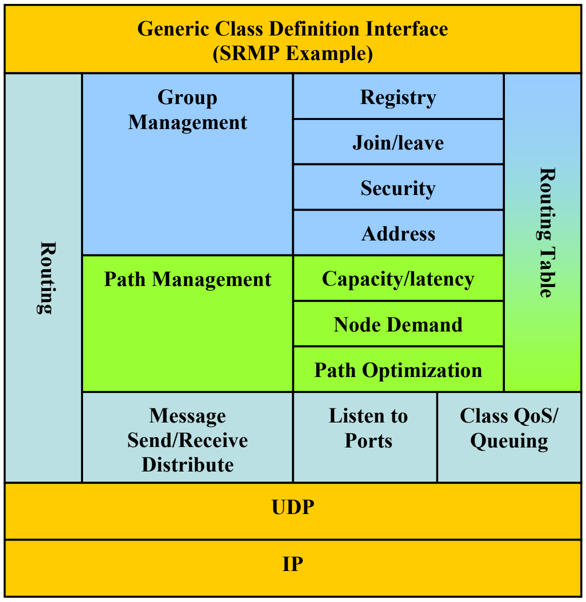


Figure 2. XOM architecture

Allocation of a specific XOM IP address, or network service access point, is not considered part of the XOM protocol and requires the use of an outside addressing/registry authority to establish an XOM host. The registry maintains a list of all XOMs in the overlay and registered RT-DVS application users. This use of an external registry provides a level of security by establishing an authority that authorizes a source to be a sender which can be used by networked XOMs receivers for recognition of authorized senders in the network. The registry also will maintain the public routable IP addresses of all active XOMs and subsequently used by the XOMs to establish efficient overlay multicast paths between XOMs, and will maintain multicast group membership information. Once an XOM host is established, internal protocol mechanisms provide for path optimization between XOM hosts and manage multicast group membership.

The core of the XOM is provided in the Group and Path management modules as the result of these activities provide the information for the routing table to be used in making message forwarding (routing) decisions. We are considering a number of approaches for managing the multicast path overlay and associated groups:

1. The XOM servers can provide a service that is independent of group management and essentially provide a path(s) overlay optimized across an open network to other registered XOMs. Under this model, the path overlay behaves like a closed network with all group communications provided as single multicast network similar to Protocol Independent Multicast – Sparse Mode (PIM-SM) specification [12]. Each user application of the XOM then listens for desired group identification communications broadcasted on the local area network hosting the XOM and discards/ignores unintended traffic.
2. The XOM can provide a service that recognizes group membership dynamics (registered XOMs that host users/applications identified by group) and provide an overly path optimized for each group. This approach is generally referred to as source-based tree multicast [13]. This approach implies management and optimization of multiple paths, maintaining a path structure for each registered multicast group.
3. The XOM can provide a service that aggregates group traffic across optimized paths between XOMs as presented in figure 3. These optimized paths are essentially shared trees and can be optimized for capacity and delay to support aggregated group traffic [14]. This approach is also similar to aggregated group multicast over MPLS [14] and with added features for group management.

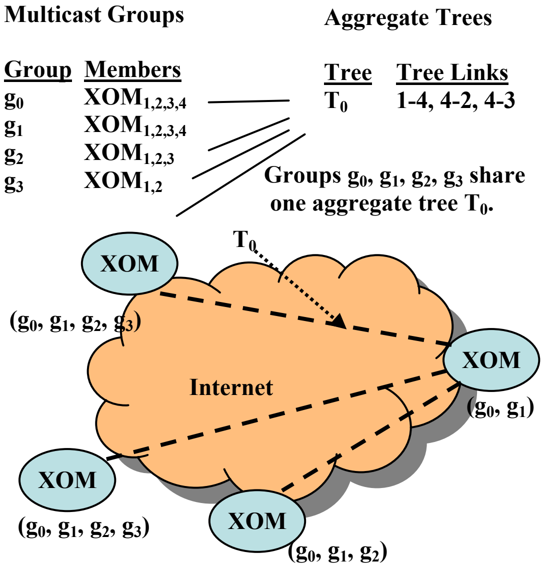


Figure 3. Group aggregation overlay

In all cases, once an XOM with an associated address is established, receivers issue a request to join existing groups using a unique connection identifier that is pre-assigned by the registry. Using this approach, the RT-DVS application is able to control or specify which sources of information are of interest. This is important, as we expect that many disparate RT-DVS applications could conceivably be running through a local supporting XOM. Group membership control can help protect an individual RT-DVS application from receipt of unspecified or undesired information flows and also aid in minimizing overall network traffic load. At this time, we have not specified how the XOM identifier is allocated or how the receivers learn its value, but is assumed to be handled through the external registry service.

To manage this process at the local network, behind the XOM, we propose to use services of IGMPv3 [15]. Version 3 adds support for "source filtering". This feature allows for a system to report interest in receiving messages only from specific source addresses and therefore, aids in the overall management of group membership and optimization of traffic loads on the network. This approach also adds a level of security to the RT-DVS application as the application is able to discard or ignore unauthorized source information.

We allow for the join request from a RT-DVS application to specify whether the receiver wishes to be a producer of information as well as a receiver, whether the connection should be able to provide the two classes of service (best effort or urgent), whether the receiver is able to accept multiple senders of information, the minimum throughput desired, and the maximum data message size. This request information is presented to the supporting XOM and used in the connection set-up process to support negotiation of the path capacity and latency parameters between the sender XOM, intermediate XOMs, and receiver XOMs. If the needed resource to support the request cannot be made available, the sender is given the option of either accepting what is available or canceling the connection request.

An RT-DVS application request for terminating membership in a group is managed locally using IGMP. This approach also simplifies global group membership management as it does not require a global group list. Local group membership records are exchanged between XOMs using the IGMP group record management features. This allows for simple coordination between XOMs for their use in allocation of path resources. For example, when the last receiver along a path has been removed, the resources allocated over that path can be released.

We have not included in the specification what action the local XOM should take when the RT-DVS application group is reduced to a single member, but a logical action would be to stop transmission if there are no active receivers and announce this to the registry service. A likely result of this action is a re-calculation of optimum path overlay to determine whether the XOM is required to stay in the network or not. The XOM leaving the network implies that it is not on any other optimum path for the remaining XOMs. In the case where the XOM stays in the network, the implication is that it is on a critical path for overall network optimization.

Our group membership approach assumes that a group definition is based on a specific RT-DVS application running behind an XOM on the local area network. Multiple instances of a RT-DVS application are supported behind each XOM, each of which may have different group membership characteristics to include membership in multiple groups. It is also feasible for a RT-DVS application to have membership in more than one group. We present an example in figure 4. Notice that application B has membership in group 1 and in group 2. In order to maintain efficiencies in message transmission, we form a new group 3 that is the union of the two groups. We also imply no explicit set-up processing between the sender and the receivers prior to the establishment of group communications. The XOM mechanism is required to pass the multicast group (IP/group tag) address to the XOM of the associated receivers. The receivers' XOM must have established group membership for the address prior to transmission in order to receive the data.

We use a decentralized algorithm to construct an overlay by searching for potential existing XOMs to become the child of in the overlay. This algorithm relies on information provided by the registry to the joining XOM as an aide in the establishment of a connection. In this process, we employ two mechanisms that contribute to global service guarantees and help manage the construction of path overlays. The first is that we put a limit on the out degree of an XOM using Bollobas [16] definition of the degree of a vertex. That is, we do not allow the construction of more than *n* connections to other XOMs where *n* is determined by the resources of the XOM and access capacity of the underlying network. This serves to limit the processing demands and network access capacity of individual XOMs in the overlay and therefore establishes lower bounds on the performance. The second is that we create a threshold for the end-to-end overall path delay from a sending XOM to a destination XOM and only offer best effort above this threshold to joining XOMs that do not successfully find an existing XOM node that is adequate to maintain end-to-end path delay thresholds.

4. Other XOM design considerations

In our earlier work [3] developing the basic concepts of overlay multicasting for RT-DVS, we presented four major factors to be considered; traffic load, resource demand, path characteristics, and path convergence. These represent routing constraints that must be managed by the XOM routing function so that the RT-DVS can rely on the network for services to meet the real-time response and predictable behavior required for the end systems. Our performance model for routing function design and development is presented in figure 5 and embraces the four factors previously defined.

The XOM routing protocol must have knowledge of the offered traffic load, the total demand for access capacity of the XOM (includes local generated traffic and routed traffic), and the relative geographic topology of the overlay network in terms of path latencies between XOMs in order

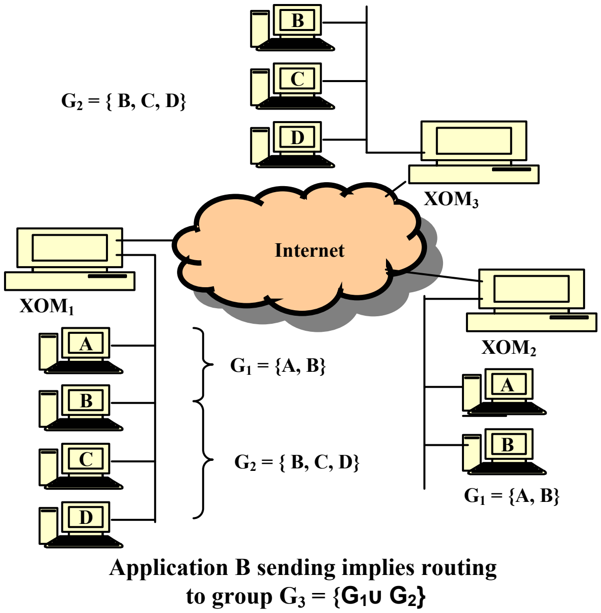


Figure 4. Group membership

to perform overlay path optimization. Optimization is accommodated by trading quality of service (defined by end-to-end latency) with the capacity of the access link, essentially rate control based on the access capacity and priority queuing. Multicast support for QoS alternatives can be realized by combining use of different multicast groups with prioritization of access capacity in the overlay. As part of our early test results discussion in section 4 below, we will further discuss how QoS can be controlled by controlling the number of paths originating from a XOM node.

The XOM, as an overlay, provides a level of guaranteed end-to-end for capacity and latency. The guarantee is provided by managing sender rate control at the link access for all connections, which are made at overlay set-up time based on statistical analysis of path and service demand characteristics shown in figure 6. XOM dynamically manages the path, ensuring that the available capacity is managed at optimum for the overlay. The enforcement policy ensures that the same path is followed for all transmissions, and prohibits new connections over the network unless there is sufficient capacity to accommodate the expected traffic. This is accomplished by maintaining the statistical state of all connections in the XOM hosts. If the underlying network is unable to provide the minimal level of service to meet local requested services, then the XOM can join the overlay but provides feedback to the application that service levels are not guaranteed. This allows the application to make a decision on whether to accept the reduced service level or decline to join.

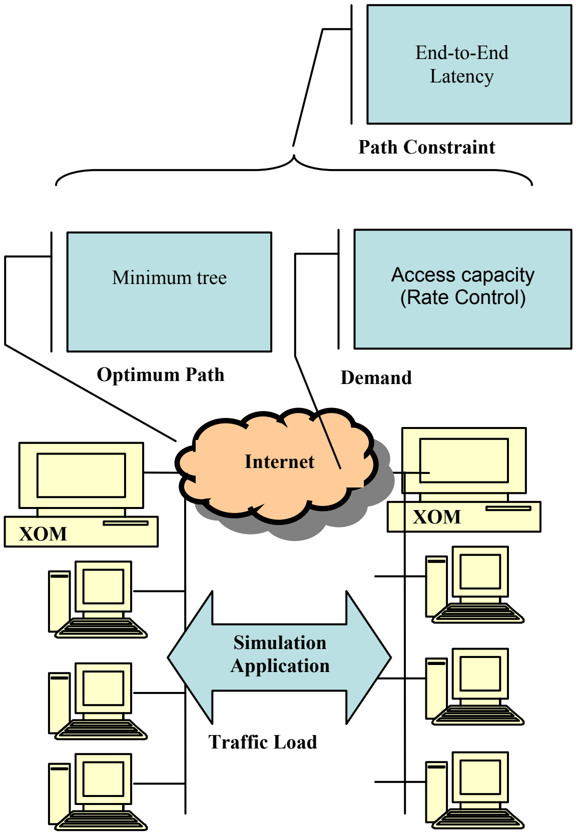


Figure 5. Overlay routing constraints

Reliability for multiple classes of RT-DVS traffic can be accomplished through use of a protocol such as SRMP over the multicast overlay or as a service provided by the XOM. The XOM service could be a weighted queuing model or message congestion marked-for-discard scheme similar to that used in Frame Relay oversubscription congestion control. In any case, our approach is to allow for two classes of service as described above.

In support of reliable transport, communication services also must support adaptation to network congestion using flow control mechanisms. SRMP provides a flow control mechanism to regulate the quantity of data placed on the network using application sender rate control based on feedback per lost messages in the network, thereby increasing network efficiency by reducing the number of dropped messages and avoiding the need for large numbers of retransmissions.

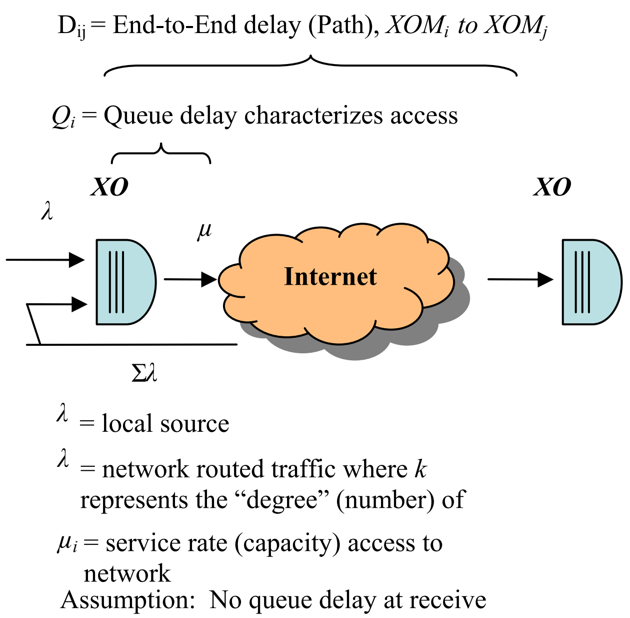

D_{ij} = End-to-End delay (Path), XOM_i to XOM_j
Q_i = Queue delay characterizes access
XO
XO
λ
μ
Internet
Σλ
λ = local source
λ = network routed traffic where k represents the "degree" (number) of
μ_i = service rate (capacity) access to network
Assumption: No queue delay at receive

Figure 6. Access model

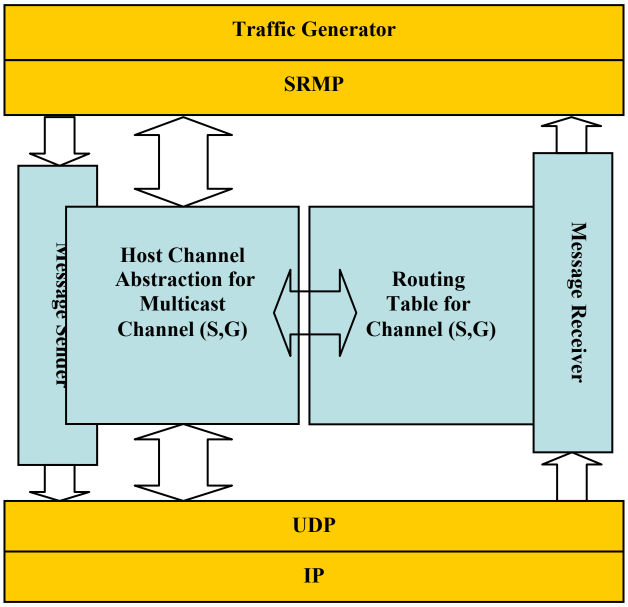

Traffic Generator
SRMP
Message Sender
Host Channel Abstraction for Multicast Channel (S,G)
Routing Table for Channel (S,G)
Message Receiver
UDP
IP

Figure 7. XOM functional model

5. Preliminary XOM performance evaluation

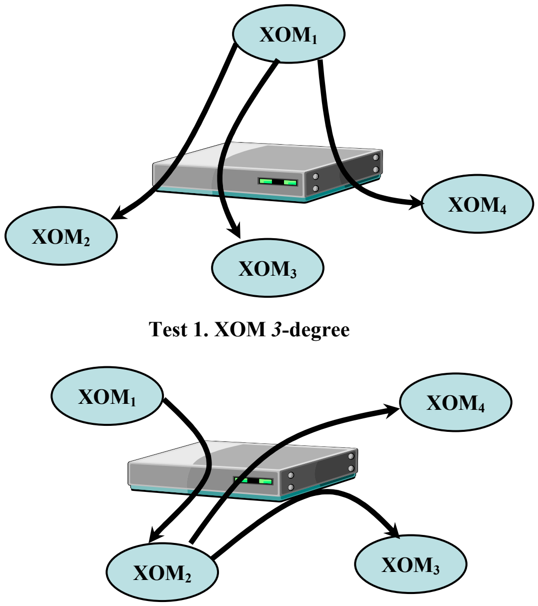
We have conducted a preliminary evaluation of our XOM approach with the objective to demonstrate message throughput of the basic functions presented in figure 7. Our specific performance interests were to measure message throughput of the send/receive functions and to gain an understanding of the impact of the *n*-degree or the number *n* of reflect messages that might be achievable from an XOM. We deem the *n*-degree factor a significant parameter as it represents the ability of the XOM to replicate messages to multiple paths or channels, therefore a driving force behind overall path construction and path optimization.

The initial test environment was established in the George Mason University C3I Networking and Simulation Laboratory. The test configuration consisted of four XOMs operating on each of their own LAN segments. These four segments were connected together via a single router to represent a WAN. Two test scenarios were conducted as represented in figure 8. In Scenario 1, we tested the XOM as 3-degree routing function where degree means number of paths required for the XOM to replicate and forward messages to. In Scenario 2, we tested the XOM as a 2-dgree routing function. Notice that we also, in Scenario 2, required the data path to traverse the WAN router twice, which represents the typical type of access architecture that we would expect for the XOM. This scenario also gives us a measure for maximum throughput based on a single channel.

The initial test environment was established in the George Mason University C3I Networking and Simulation Laboratory. The test configuration consisted of four XOMs operating on each of their own LAN segments. These four segments were connected together via a single router to represent a WAN. Two test scenarios were conducted as represented in figure 8. In Scenario 1, we tested the XOM as 3-degree routing function where degree means number of paths required for the XOM to replicate and forward messages to. In Scenario 2, we tested the XOM as a 2-dgree routing function. Notice that we also, in Scenario 2, required the data path to traverse the WAN router twice, which represents the typical type of access architecture that we would expect for the XOM. This scenario also gives us a measure for maximum throughput based on a single channel.

We used the source traffic generator described in [11] and varied the number of source objects in order to map performance against offered load. During test run start up, there is a high message rate as objects execute join commands. We executed each test run for 1800 seconds to ensure that we reached steady-state. At steady-state, each object represents approximately 3.1 messages per second of offered load with a message payload size of 120 bytes. We used SRMP [11] as the interface to the XOM.

The results of our experiments are presented in figure 9 (Mean delay) and figure 10 (Loss ratio). Our area of interest for the results is for loss ratios of less than 1%. Under both scenarios, the XOM message throughput was approximately 1300 messages per second. For the 2-degree network this represented slightly greater than 400 simulated objects supported with delay averaging 30 msec. In the case of the 3-degree network, greater than 300 source simulated objects were supported with message delay averaging 32.2 msec.



Test 1. XOM *3*-degree

Test 2. XOM 2-degree

Figure 8. XOM test scenarios

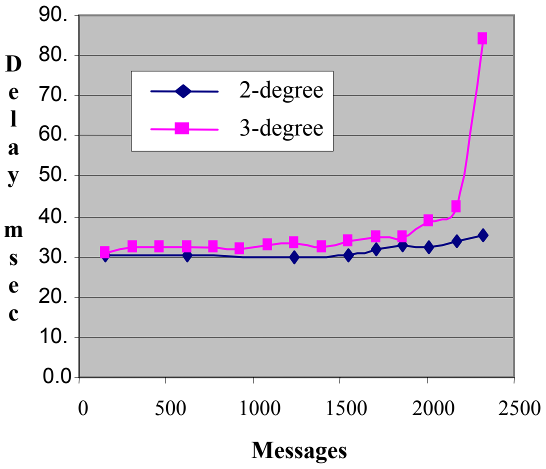


Figure 9. Mean delay

In another system we have studied, reported in a conference companion paper by Morse, et al, [17], 4000 messages per second supported 10,000 objects. We have not yet completed code optimization of the existing XOM, which is written in Java. Our initial results, reprogramming the performance-critical module with C++, have achieved a factor of three performance improvement over the results shown here. Thus our experiment has given us confidence that we are able to achieve desired throughput capabilities to support Web based services using overlay multicast.

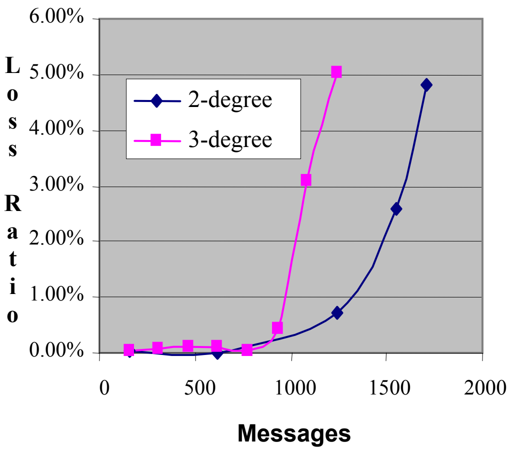


Figure 10. Loss ratio

6. Conclusions and future work

This paper describes an initial overlay multicast implementation that provides a many-to-many multicast service for distributed virtual simulation environments. We have described the architecture of the XOM and also the key design considerations employed during the XOM development.

We deployed the XOM prototype in our Lab using a network of four XOMs and evaluated its throughput characteristics under various loading conditions. We measured throughput in two different scenarios in order to evaluate throughput under routing conditions where the XOM was required to perform message replication. Our results indicated performance levels that show us great promise that overlay multicast can meet the needs of the RT-DVS environment, particularly for web-based distributed simulations. This gives us confidence that our approach can provide a networking capability that allows for RT-DVS to be independent of underlying network protocols.

As we move to the next level in our experiments and development, we recognize there are many challenges. We must remain focused on efficiency and scalability as well as overlay management mechanisms that respond to the dynamic nature of underlying open networks. We need to better understand the network characteristics of the collaborative and distributed virtual simulation environments, as these characteristics must be accommodated in the XOM. Our planned next steps are:

- Move from the laboratory to conducting an experiment across an open WAN network.
- Continue refining the protocol by prototyping repeatedly at increasing levels of sophistication and optimization.
- Integrate with early prototypes of RT-DVS Web services development projects.

7. Acknowledgement

This work was supported in part by the US Defense Modeling and Simulation Office.

8. References

[1] R. Braudes and S. Zabele, “Requirements for Multicast Protocols”, *IETF RFC 1458*, 1993.

[2] D. Brutzman, M. Zyda, M., J. Pullen, and K. Morse, “Extensible Modeling and Simulation Framework (XMSF): Challenges for Web-Based Modeling and Simulation”, US Naval Postgraduate School, 2002

[3] D. Moen and J. Pullen, “Enabling Real-Time Distributed Virtual Simulation over the Internet Using Host-based Overlay Multicast”, *Proceedings of the Seventh IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2003, pp. 30-36.

[4] A. Boukerche and C. Dzermajko, “Performance Comparison of Data Distribution management Strategies”, *Proceedings of the Fifth IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2001, pp. 67-75.

[5] G. Tan, Y. Hu, and F. Moradi, “Automatic SOM Compatibility Check & FOM Development”, *Proceedings of the Seventh IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2003, pp. 60-67.

[6] K. Morse and M. Petty, “Data Distribution management Migration form DoD 1.3 to III 1516”, *Proceedings of the Fifth IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2001, pp. 58-65

[7] L. Wang, S. Turner, and F. Wang, “Interest Management in Agent-based Distributed Simulations”, *Proceedings of the Seventh IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2003, pp. 20-27.

[8] Y. Wang and Y. Liao “Implementation of a Collaborative Web-based Simulation Modeling Environment”, *Proceedings of the Seventh IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2003, pp.150-157.

[9] A. Tolk, “Moving Towards a Lingua Franca for M&S and C3I – Developments concerning the C2IEDM”, *2004 Euro Simulation Interoperability Workshop*, June 2004.

[10] J. Pullen, "Reliable Multicast Network Transport for Distributed Virtual Simulation", *Proceedings of the 1999 IEEE Workshop on Distributed Simulation and Real-Time Applications,* 1999.

[11] D. Moen and J. Pullen, "A Performance Measurement Approach for the Selectively Reliable Multicast Protocol for Distributed Simulation", *Proceedings of the Fifth IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2001, pp. 30-34.

[12] D. Estrin *et. al.*, “Protocol Independent Multicast-Sparse Mode (PIM-SM) Protocol Specification”, *IETF RFC 2362*, 1998.

[13] A. Fei, D. Zhihong, and M. Gerla, “Constructing shared-tree for group multicast with QoS constraints”, *IEEE GLOBECOM '01*, Volume: 4, Nov. 2001 pp. 2389–2394.

[14] J. Cui, M. Faloutsos and M. Gerla, “An Architecture for Scalable, Efficient, and Fast Fault-Tolerant Multicast Provisioning,” *IEEE Network Magazine*, March/April 2004. pp. 26-34.

[15] B. Cain *et. al.*, “Internet Group Management Protocol, Version 3”, *IETF RFC 3376*, 2002.

[16] B. Bollobas, *Random Graphs*, 2nd Edition, Cambridge Press, NY, 2001.

[17] K. Morse *et, al.*, “An Architecture for Web-Services Based Interest Management in Real Time Distributed Simulation”, Companion paper in the *Proceedings of the Eighth IEEE Workshop on Distributed Simulation and Real-Time Applications,* 2004.