LCMON Network Traffic Analysis

Adam Black
Centre for Advanced Internet Architectures, Technical Report 071029A
Swinburne University of Technology
Melbourne, Australia
adamblack@swin.edu.au

***Abstract*—The Swinburne supercomputer (as of May 2007) consists of 145 x Dell Power Edge 1950 nodes, each containing 2 x quad core Clovertown processors @ 2.33 GHz, 16 GB of RAM and 2 x 500 GB hard disks. LCMON 1.0 is a utility based on L3DGEWorld 2.1 which allows for real-time visualisation of the supercomputer cluster.**

In this report we describe several experiments conducted to model the incoming and outgoing traffic at a LCMON 1.0 server as a function of the number of clients connected. We also compare the LCMON traffic when the clients move around the LCMON world as opposed to when they are stationary.

Some latter experiments revealed we can reduce the time required to update all 145 avatars in the LCMON world from 63 seconds to 16 seconds by sending metrics to the LCMON (L3DGEWorld) server at a faster rate. We also demonstrate that changes in avatar's metrics are responsible for increases in the LCMON server to client data rate, however after the clients have been informed of the changes the data rate reduces.

I. Introduction

LCMON 1.0 (L3DGEWorld Cluster-node Monitoring) is a utility designed to visually represent activity in the Swinburne supercomputer clusters using L3DGEWorld 2.1 [1]. The various metrics of each cluster node such as CPU utilisation, memory usage and network traffic are represented in LCMON 1.0 by avatars spinning, varying in colour / size and bouncing. L3DGEWorld 2.1 is based on the OpenArena engine [2] which is an open source clone of Quake 3.

In order to view the supercomputer activity with LCMON you must connect to a LCMON server. Broadly speaking, the purpose of the LCMON server is to collect statistics from the supercomputer and update the avatars accordingly in the 3D environment provided by OpenArena.

This document analyses the network traffic generated between a LCMON 1.0 server and one or more clients.

Section II briefly discusses the software and equipment used in our testbed. In Section III we consider the traffic load at the LCMON 1.0 server when several clients connect. We also consider the two cases where the clients remain idle and when they are actively moving around the 3D LCMON world. In Section IV we consider how changing the avatars during gameplay can affect the server to client traffic load. We also investigate the time required to update all the avatars in the LCMON world and determine if this time can be decreased and what affect this has on bandwidth.

II. LCMON 1.0 Traffic Monitoring Testbed

A. Software Configuration

All tests presented in this document were conducted using LCMON 1.0 on the server and client PCs. The LCMON clients were configured to use the 'Fastest' graphics mode in the OpenArena engine such that the client to server packet rate would not be compromised by lower end video cards.

B. Hardware/Software Platforms

The specifications of the PCs we used is displayed in Table I.

TABLE I
Specifications of the Testing PCs

Name	Hardware	OS
Server	P4 (2.66GHz), 512MB RAM	FreeBSD 6.2
Client 1	Celeron (2.40GHz), 512MB RAM, GeForce 6600 video	Windows XP
Client 2	P4 (3.00GHz), 512MB RAM, Intel 82915G video	Windows XP
Client 3	P4 (2.80GHz), 512MB RAM, Intel 82865G video	Windows XP
Client 4	Conroe T2600 (2.16GHz), 2GB RAM, Intel GMA 950 video	Mac OS X 10.4.10
Client 5	P4 (2.80GHz), 768MB RAM, GeForce 5200 video	Windows XP
Client 6	Conroe T2300 (1.66GHz), 1GB RAM, Quadro NVS 110M video	Debian GNU/Linux

III. Typical LCMON 1.0 Traffic With Multiple Clients

In this section we will analyse the typical server to client and client to server traffic when one or more LCMON clients connect to a server. The goal is to establish a relationship between the throughput at the server and the number of clients connected.

We will examine two different scenarios which are discussed in Section III-A and III-B, then in Section III-C we will estimate the server to client traffic for an arbitrary number of LCMON clients.
NB: All the traffic traces (described below) were only recorded during gameplay, and do not include the initial connection setup packets.

A. Stationary Clients

In the first scenario a single client connected to the LCMON 1.0 server and remained stationary in the 3D OpenArena world while a traffic trace was recorded on the server over a two minute period, using tcpdump. This test was repeated while two, three and four stationary clients were connected to the server. The LCMON client-server configuration is shown in Figure 1.

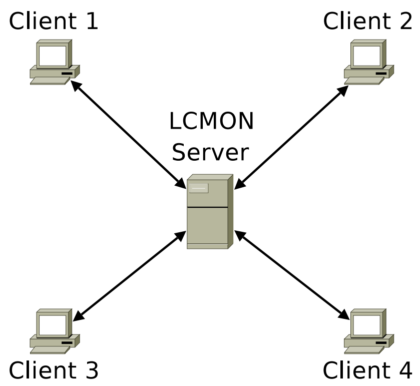

Fig. 1. LCMON client-server testbed configuration when all four clients were connected and stationary in the 3D world

Figure 2 and 3 show the average data rate and packet rate of the aggregate LCMON 1.0 traffic in the server to client direction, while Figure 4 and 5 show similar graphs, except when the traffic is flowing in the client to server direction. Figure 6 and 7 show the server to client and client to server packet length distributions[1].

In Figure 2 we can see the server to client data rate seems to increase in a non-linear fashion as more

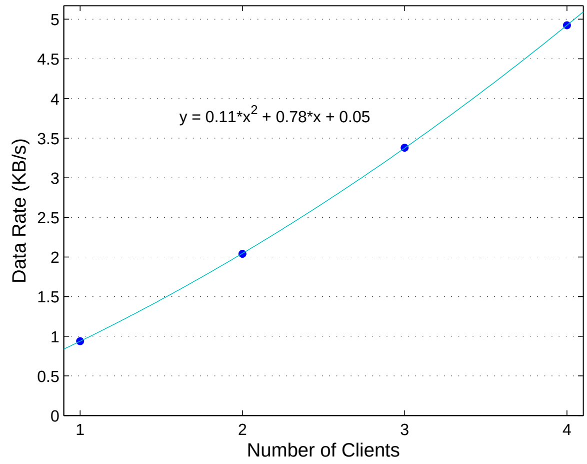

Fig. 2. Aggregate LCMON server to client data rate with stationary clients

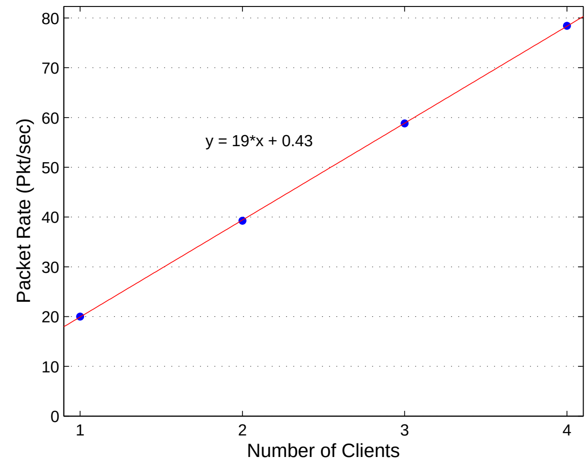

Fig. 3. Aggregate LCMON server to client packet rate with stationary clients

clients connect to the LCMON 1.0 server, however notice in Figure 3 the packet rate increases linearly. This is because the OpenArena engine (used by L3DGEWorld 2.1) sends snapshots to clients every 50 ms by default, however as the number of clients increases the size of the snapshots must also increase. This is confirmed in Figure 6.

Figure 3 shows as more clients connect to the LCMON server, the packet rate per client drop below the expected 20 pkts/sec. In order to better understand this we graphed the inter-arrival time of packets flowing from the Server to Client 1 when four clients were connected to the LCMON server. This plot has been shown in Figure 8 and the distribution percentage was calculated on a total

[1]The packet lengths in all calculations are based on the IP datagram size

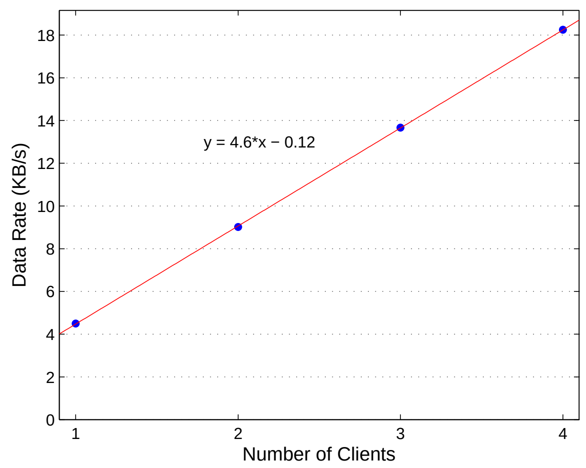

Fig. 4. Aggregate LCMON client to server data rate with stationary clients

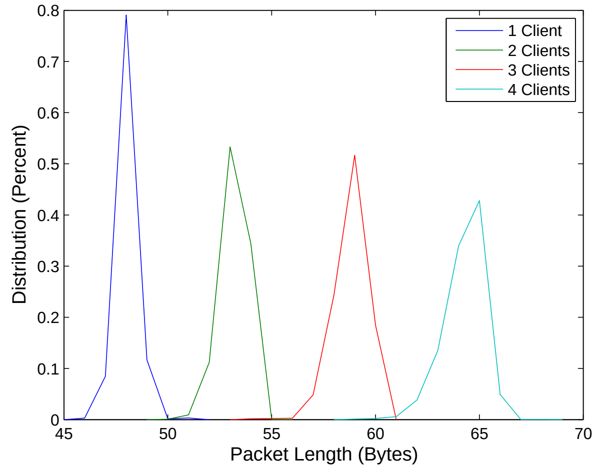

Fig. 6. Aggregate LCMON server to client packet length distribution with stationary clients

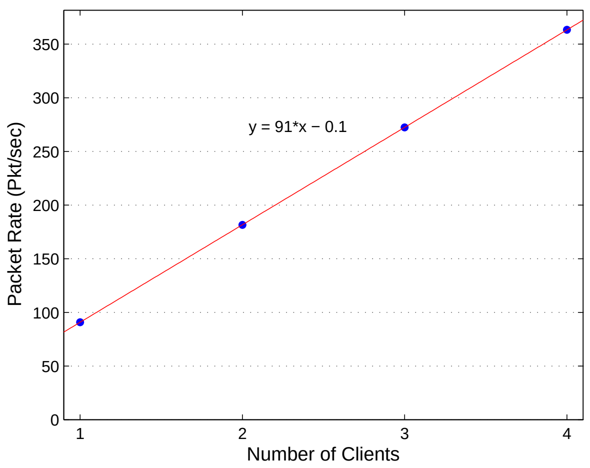

Fig. 5. Aggregate LCMON client to server packet rate with stationary clients

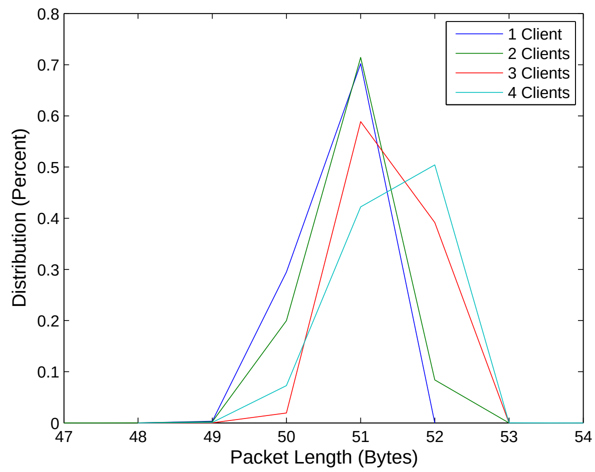

Fig. 7. Aggregate LCMON client to server packet length distribution with stationary clients

of 2348 captured packets. In Figure 8 the inter-arrival times range from 50.67 ms to 63.01 ms, we can see however that packets are mostly distributed around 51 ms. The fact that no packets have an inter-arrival time of: $50 * x$ milliseconds, where $x = 2, 3, 4, \ldots$ suggests that no outgoing LCMON snapshots were lost by the tcpdump filter and any discrepancies in the packet rate were caused by the OpenArena engine.

Figure 4 and 5 show that the bulk of the LCMON 1.0 traffic is in the client to server direction. Notice the packet rate per client is approximately 90 pkts/sec. Interestingly enough the clients used the default maximum frame rate setting of 85 frames/sec.

Figure 7 suggests the packet length of client to server traffic is not affected by additional clients connecting to the LCMON server.

B. *Moving Clients*

This section is similar to Section III-A, however the tests performed here involved the clients moving around the LCMON world and 'shooting' entities to reveal various statistics about the supercomputer nodes. The purpose of this scenario was to simulate realistic usage of LCMON 1.0. The LCMON client-server configuration is shown in Figure 9.

Figure 10 and 11 show the average data rate and packet rate of the aggregate LCMON 1.0 traffic in the server to client direction, while Figure 12 and 13 show

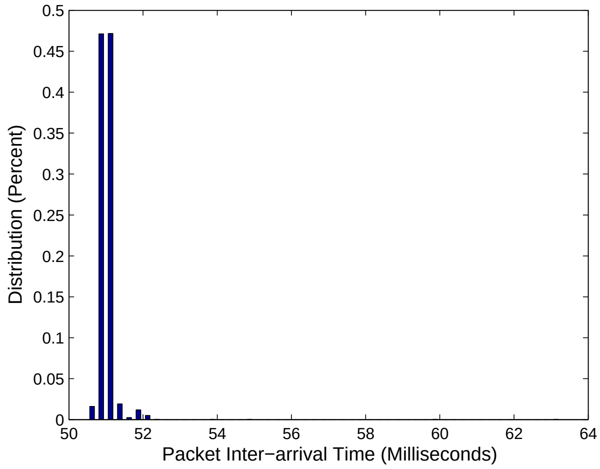

Fig. 8. LCMON Server to Client 1 packet inter-arrival time with four stationary clients

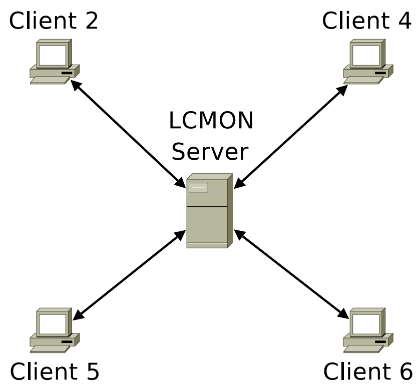

Fig. 9. LCMON client-server testbed configuration when all four clients were connected and moving around in the 3D world

similar graphs, except when the traffic is flowing in the client to server direction. Figure 14 and 15 show the server to client and client to server packet length distribution[1].

Figure 10 shows that moving clients cause a larger server to client traffic load compared with stationary clients, especially as the number of clients becomes quite large. In Figure 11 we can see the outgoing LCMON packet rate at the server is still ∼20 pkts/sec per client.

Comparing Figure 12 and 4 we can see that moving clients cause a larger client to server data rate compared with stationary clients, however the relationship between the number of clients and data rate is still linear. Also notice in Figure 13 the client to server packet rate is still

[1]The packet lengths in all calculations are based on the IP datagram size

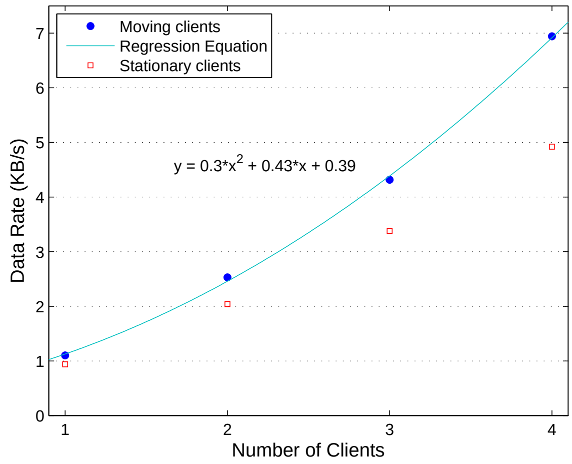

Fig. 10. Aggregate LCMON server to client data rate

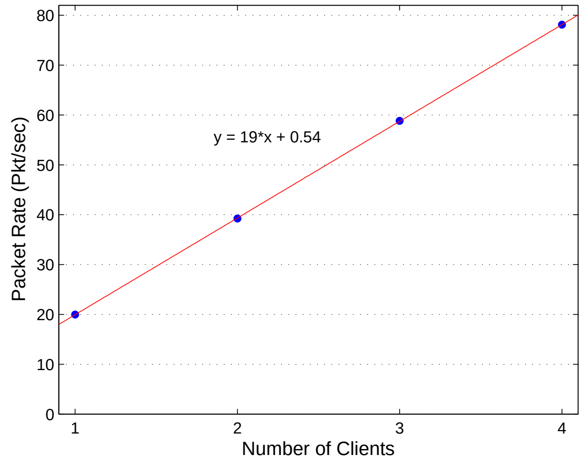

Fig. 11. Aggregate LCMON server to client packet rate with moving clients

approximately 90 pkts/sec per client.

In Figure 14 we can see the server to client packet length distribution has a large spread of values, ranging from approximately 50 to 350 bytes. It can be seen that as the number of clients increase, the server to client packet lengths generally increase as well.
Figure 15 suggests the client to server packet length distribution is not affected by the number of clients connected and the packet lengths generally fall in the range of 50 to 85 bytes.

C. *Traffic Projection*

In this section we will use the regression model seen in Figure 10 to estimate the server to client data rate (for LCMON traffic) when more than four clients connect to

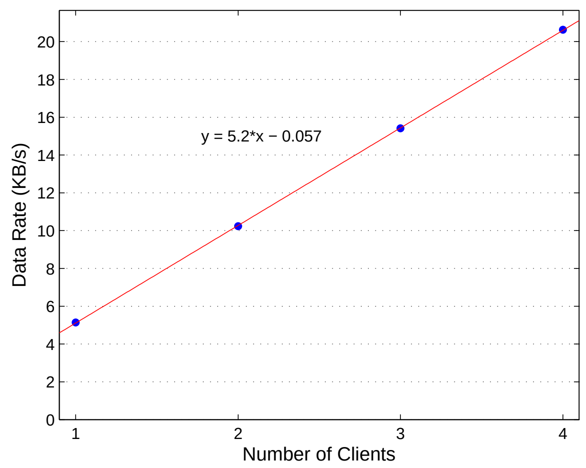

Fig. 12. Aggregate LCMON client to server data rate with moving clients

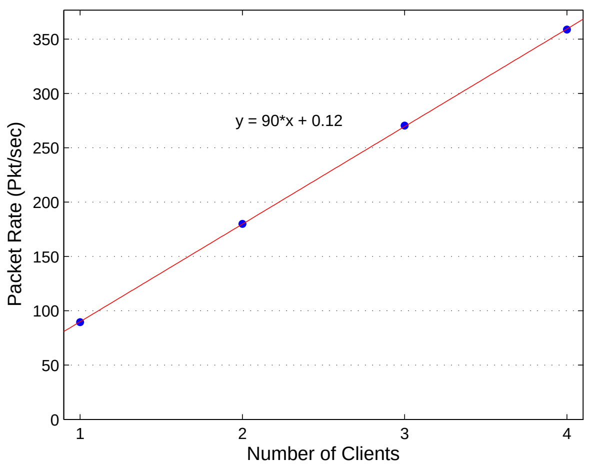

Fig. 13. Aggregate LCMON client to server packet rate with moving clients

the LCMON server and move around the 3D world in a realistic fashion. In Table II the first column contains the number of clients and the second column contains the estimated aggregate server to client data rate.

The server to client traffic load quickly increases as the number of clients grow, however this model may not be an ideal indication of the exact traffic load to expect with N clients. This is because the regression equation is based on only four data points.

D. Conclusion

The LCMON 1.0 server to client data rate is influenced by the number of clients connected to the server and the clients' movement, however the packet rate

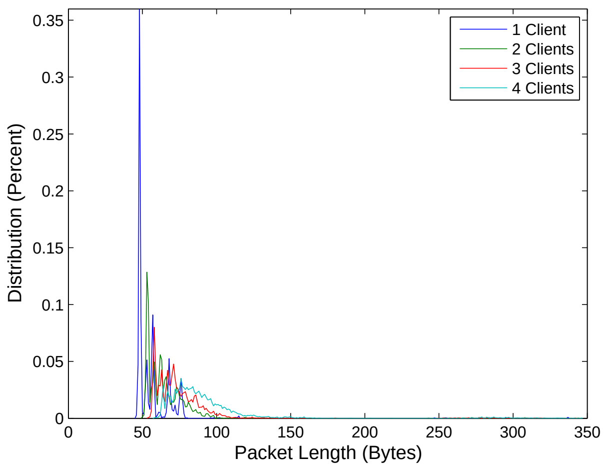

Fig. 14. Aggregate LCMON server to client packet length distribution with moving clients

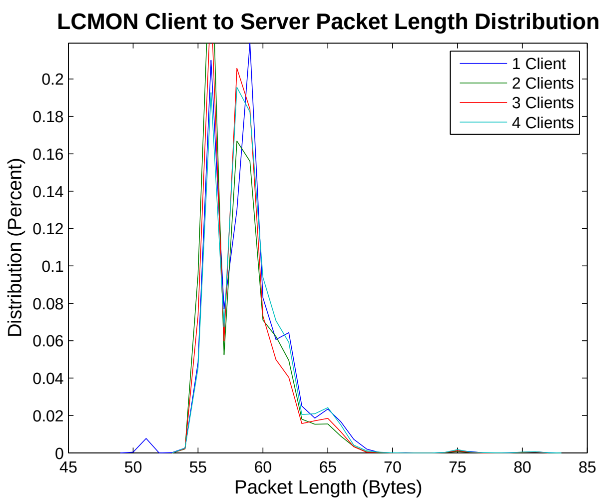

Fig. 15. Aggregate LCMON client to server packet length distribution with moving clients

TABLE II
SERVER TO CLIENT DATA RATE PROJECTION

LCMON Clients (N)	Data Rate (KB/sec)
5	10
8	23
10	35
15	74
20	129
25	199
30	283
35	383
40	498
50	772

always remains constant at approximately 20 pkts/sec per client. The client to server data rate is only affected by the movement of clients in the 3D world and the packet rate always remains constant at around 90 pkts/sec. This is due to the frame rate on the clients' machines.

IV. LCMON 1.0 Traffic With Alternating Avatars

The various metrics or statistics for each cluster node in the supercomputer are represented by avatars (or entities) changing their behaviour in the LCMON 3D world, metrics also consist of data such as hostname, IP address, hard disk usage and other statistics which are not reflected by physical changes in the avatars.

The L3DGEWorld (or LCMON) server does not interact directly with the supercomputer, instead this is performed by a C program bundled with the LCMON package named gpoll. The purpose of gpoll is to download the latest statistics from the supercomputer and then parse and transmit the statistics to a L3DGEWorld server, which can be on the same, or different machine as the gpoll process. This is because gpoll transmits metrics to the supercomputer using the UDP protocol. In the case of the experiments presented in this section gpoll was configured to transmit metrics to the localhost, ie. The gpoll and LCMON 1.0 server processes both resided on the same PC.

When transmitting UDP updates from gpoll to the L3DGEWorld server it is necessary to implement a delay between the departure of each packet so the L3DGEWorld server is not overloaded. For this reason gpoll contains some delays between the transmission of each metric. In this section we will measure the server to client data rate caused by the LCMON avatars being updated during gameplay. We will also investigate the time required to send metric updates to the L3DGEWorld 2.1 server and attempt to reduce the delays in gpoll to see how this affects the LCMON traffic and entity updates.

A. Method

In order to test the server to client bandwidth utilisation during metric updates a single LCMON 1.0 client connected to the server and remained idle while various fabricated metrics were sent to the L3DGEWorld server. Tcpdump was used on the server to monitor the LCMON traffic.

The test consisted of three phases. In the first phase all the cluster nodes were set to the 'minimum metric' such that they were idle, not spinning and the minimum size. Two minutes later the entities were set to the 'maximum metric' such that they were spinning and bouncing at the maximum rate and the largest size. Another two minutes later the entities were configured with randomly assigned metrics and the three phase cycle continued for several iterations.

We also ran the test (as described above) with some variations in gpoll. In Section IV-B we left gpoll completely unmodified. In Setion IV-C we ran two tests in which we modified gpoll to reduce the delay between the transmission of metrics to the LCMON server process. The testbed configuration for this experiment is shown in Figure 16

Note that There are 145 entities (cluster nodes) in the Swinburne supercomputer and for each entity gpoll transmits ten metrics to the L3DGEWorld server.

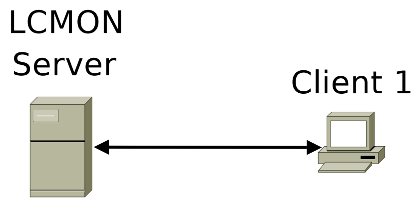

Fig. 16. LCMON client-server testbed for 'Alternating Avatars' experiments

B. Default GPOLL Configuration

We performed the tests in this section with the default version of gpoll, which contains an effective 420 ms delay between the update of each entity in the supercomputer cluster. Considering there are 145 cluster nodes we would expect the entire cluster to be updated in approximately: $0.42 * 145 = 61$ seconds.

Figure 17 displays the average data rate over time with a five second time bin, and Figure 18 shows the packet length histogram for the duration of the entire test. s

Figure 17 shows there is an increase in the server to client data rate when the metrics are being updated and sent to the client. Approximately 63 seconds after the metrics have been sent to the LCMON server the data rate drops to a constant level of approximately 0.9 KB/sec, which indicates the L3DGEWorld server only sends delta snapshots to its clients. This means the game engine only informs the client of changes it needs to know about.

Another thing to note is the first phase of entity updates consumes more bandwidth than consecutive updates. This is because when the LCMON client first initialises it has no 'static' information about any of the entities. For example the host name and IP address. After

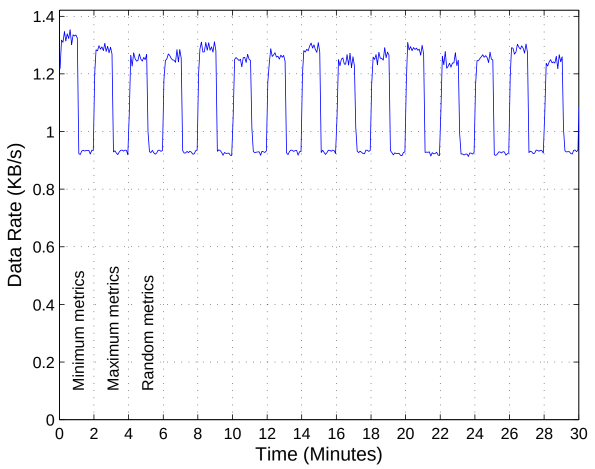

Fig. 17. LCMON server to client data rate (420 ms delay per entity)

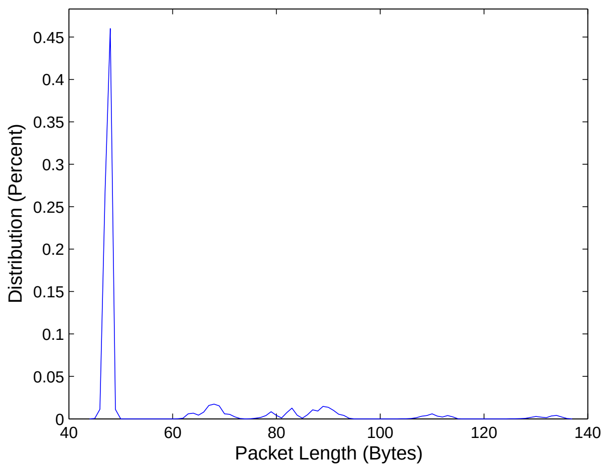

Fig. 18. LCMON server to client packet length distribution (420 ms delay per entity)

the first phase of updates these 'static' metrics are never retransmitted to the LCMON client.

We can also see from Figure 17 the various metric update phases (min, max, random) result in a slightly different server to client data rate. This occurs because transmitting larger numbers (in the maximum metrics phase) consumes more bits over ethernet than sending smaller values (in the other two phases).

C. Modified GPOLL Configuration

For the first test we modified gpoll to create an effective 190 ms delay between the update of each avatar. For the final test we reduced the delay further such that there was a 100 ms delay between entity updates. In the first test we would expect to wait approximately $0.19 * 145 = 28$ seconds before all, 145 avatars are updated with the latest metrics. For the second scenario we expect all the entities to be updated in around: $0.1 * 145 = 15$ seconds.

Figure 19 and 20 show the results of the first test scenario, while Figure 21 and 22 show the results of the second test scenario.

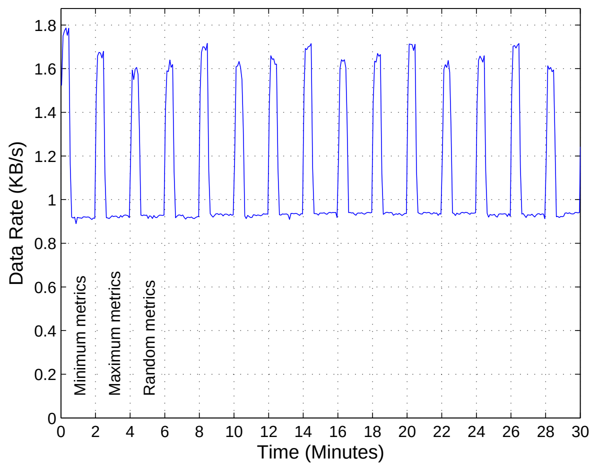

Fig. 19. LCMON server to client data rate (190 ms delay per entity)

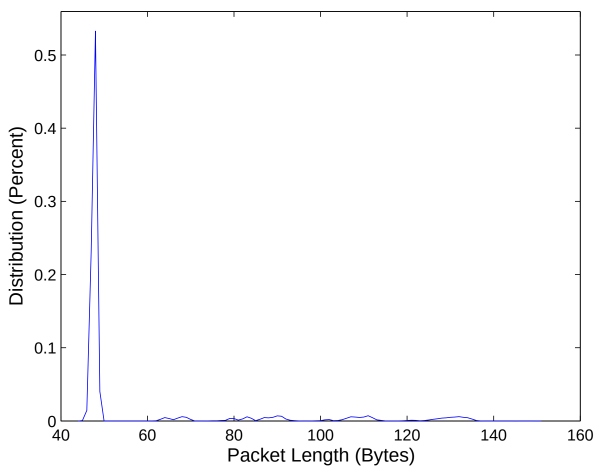

Fig. 20. LCMON server to client packet length distribution (190 ms delay per entity)

Figure 19 and 21 exhibit very similar behaviour to what was demonstrated in Figure 17. We can see however that the peak server to client data rate increases as the metrics are sent more quickly to the L3DGEWorld server.

In Figure 19 the metric update period lasts approximately 30 seconds and in Figure 21 this period is reduced

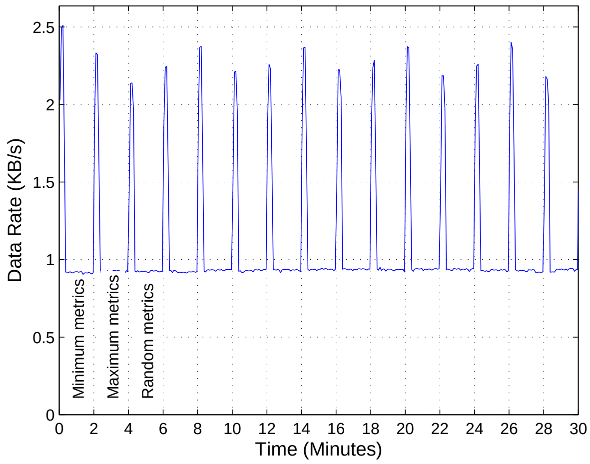

Fig. 21. LCMON server to client data rate (100 ms delay per entity)

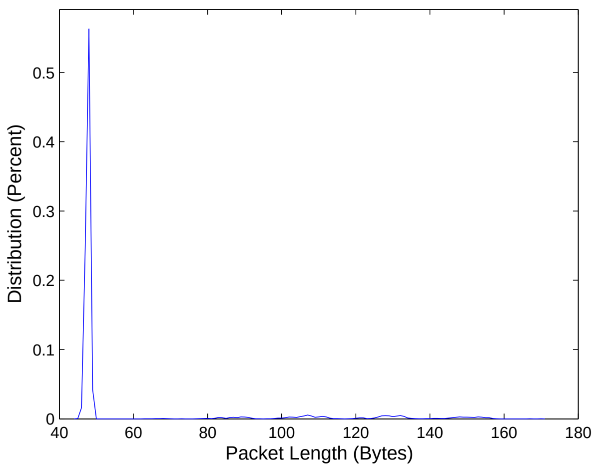

Fig. 22. LCMON server to client packet length distribution (100 ms delay per entity)

to 16 seconds. This means if gpoll is modified such that there is a 100 ms delay between entity updates, we are able to update 145 entities (containing 10 metrics each) in approximately 16 seconds.

D. Limitation

We did not conduct similar experiments with gpoll running on a separate PC.

E. Conclusion

We demonstrated that L3DGEWorld only sends delta snapshots to clients. We also found we are able to reduce the time required to update all 145 avatars from 63 seconds to 16 seconds by increasing the transmission rate of metric updates to the LCMON server.

V. Conclusion

LCMON 1.0 (L3DGEWorld Cluster-node Monitoring) is a utility designed to visually represent activity in the Swinburne supercomputer clusters using L3DGEWorld 2.1. In Section III we ran several experiments to model the relationship between the number of clients connected to a LCMON 1.0 server and the traffic load. The results showed the server to client packet rate remains quite constant at 20 pkts/sec per client, while the data rate increases non-linearly as more clients join the server. We also demonstrated the client to server data rate and packet rate is not affected by the number of clients in the server; the data rate is affected by the movement of clients and the packet rate is influenced by the frame rate of LCMON clients. In Section III-B we found that active clients cause a greater server to client traffic data rate than stationary clients.

In Section IV we considered how alternating the metrics of avatars during gameplay affects the LCMON server to client traffic. The results suggested that metric changes to avatars causes an increase in outbound LCMON server traffic, however after the avatars' metrics have been updated at the client the data rate reduces. In Section IV-C we found that we could lower the time taken to update all 145 avatars in the LCMON 1.0 world from 63 seconds to 16 seconds by sending metric updates to the L3DGEWorld 2.1 server at a faster rate.

References

[1] L. Parry, "Leveraging 3D Game Engines (L3DGE)," Viewed 23 October 2007, http://caia.swin.edu.au/urp/l3dge/tools/l3dgeworld_2.1/index.html.

[2] Open Arena, "OpenArena," Viewed 23 October 2007, http://openarena.ws.