Chapter 8

Results

8.1 Experimental Setup

The system comprises of 22 server node processes organized into the topology shown in the Figure 8.1.1. Each server node process is hosted on 1 physical Sun SPARC-5, with no SPARC-5 node hosting two server node processes. The run-time environment for the server node processes is JDK-1.2. For the purpose of gathering performance numbers we have 1 publisher in the system and 200 client node processes with 5 client nodes attached to every server node within the system. The 100 client node processes reside on a SPARC-60 machine. The publisher is responsible for issuing events, while the subscribers are responsible for registering their interest in receiving events. The publisher and the mesasuring subscriber reside on another SPARC-5 machine.

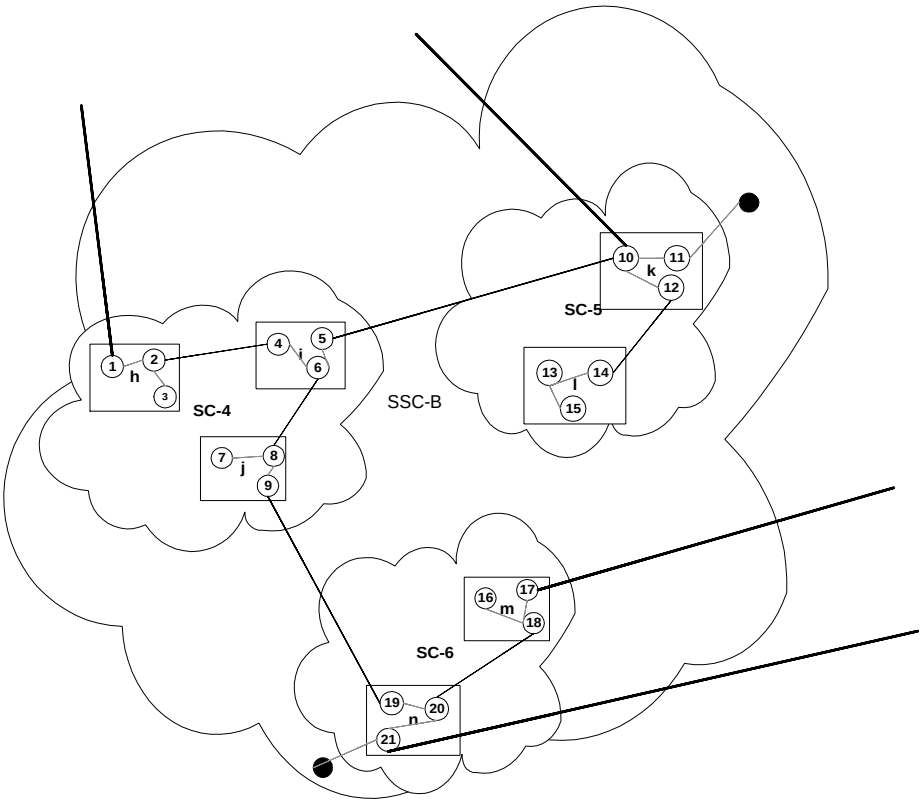

Figure 8.1.1: Testing Topology

8.2 Factors to be measured

Once the publisher starts issuing events the factor that we are most interested in is the *latency* in the reception of events. This latency corresponds to the response times experienced at each of the clients. We measure the latencies at the client under varying conditions of *publish rates*, *message sizes* and *matching rates*. Matching rate is the percentage of events that are actually supposed to be receieved at a client. In most publish subscribe systems, at any given time for a certain number of events being present in the system, clients are only interested in a subset of these events. Varying the matching rates allows us to simulate such a scenario. For a sample of messages received at a client we calculate the *mean latency* for sample of received messages, the *variance* in the sample of these messages and the *system throughput measured* in terms of the number of messages received per second. We also measure the highest and lowest message latencies within the sample of messages that have been received.

8.2.1 Measuring the factors

For events published by the publisher the number of tag-value pairs contained in every event is 6, with the matching being determined by varying the value contained in the fourth tag. The profile for all the clients in the system, thus have their first 3 <*tag=value*> pairs identical to the first 3 pairs contained in every published event. This scheme ensures that for every event for which destinations are being computed there is some amount of processing being done. Clients attached to different server nodes specify an interest in the type of events they are interested in. This matching rate is controlled by the publisher, which publishes events with different footprints. Since we are aware of the footprints for the messages published by the publisher, we can accordingly specify profiles which will allow us to control the dissemination within the system. When we vary the matching rate we are varying the percentage of events published by the publisher that are actually being received by clients within the system. Thus when we say that the matching rate is set at 50%, any given client will receive only 50% of the events published by the publisher. To vary the publish rates, we control the *sleep time* associated with the publisher thread, and also the number of messages that it publishes at a time once the publisher thread *wakes up*. This requires some preliminary tuning, once the values for the sleep time and the number of messages at a time have been fixed, we proceed to compute the real publish rates for the sample of messages that we send. This is the publish rate that we report in our results.

For each matching rate we vary the size of the messages from 30 to 500 bytes, and vary the publish rates at the publisher from 1 to 1000 Messages per second. For each of these cases we measure the latencies in the reception of events. To compute latencies we have the publishing client and one of the subscribers residing on the same machine. Event's issued by the publisher are time-stamped and when they are received at the subscribing client the difference between the present time and the time-stamp contained in the received message constitutes the latency in dissemination through the server network. In case the publisher and the subscriber were on two different machines, with acess to different underlying system clocks we would need to synchronize the clocks and also account for the drift in clock rates prior to computing the latencies in message reception. Having the publisher and one of the subscribers on the same physical machine with access to the same underlying clock, obviates this need for clock synchronization and accounting for clock drifts. It should be noted that though the publisher and one of the subscribers are on different machines, they are connected to two different server nodes in the server network.

8.3 Discussion of Results

At high publish rates and increasing message sizes, the effects of queuing delays come into the picture. This queuing delay is a result of the messages being added to the queue faster than they can be processed. The results clearly demonstrate the effects of flooding/queuing that take place at high publish rates and high message sizes and high matching rates at a client. It is clear that as the matching rate reduces the latencies involved also reduce.

The variance in the sample of messages varies from 69713$mSec^2$ to 133.76$mSec^2$ for $< 952msg/Sec, 450Bytes >$ at matching rates of 100% to $< 877msg/Sec, 450Bytes >$ at matching rates of 5%.

Figures 8.3.1 through 8.3.7 depict the pattern of decreasing latencies with decreasing matching rates. The latencies vary from 391.85 *mSecs* to 48.2 *mSecs* with the *¡publish rate, message size¿* varying from $< 952, 450 >$ for a matching rate of 100% to $< 961, 425 >$ for a matching rate of 5%. This reduction in the latencies for decreasing matching rates, is a result of the routing algorithms that we have in place. These routing algorithms ensure that events are routed only to those parts of the system where there are clients which are interested in the receipt of those events. Thus events are queued only at those server nodes which

- Have attached clients interested in those events
- Are en route to server nodes which are interested in these events. These server nodes generally fall in the shortest path to reach the node.

In the flooding approach, all events would still have been routed to all clients irrespective of the matching rates. Variance for the sample of received messages at a client, demonstrate how queueing delays can add up to increase the mean latency, and also how this mean latency has high deviations from the highest and lowest latencies contained in the sample of messages.

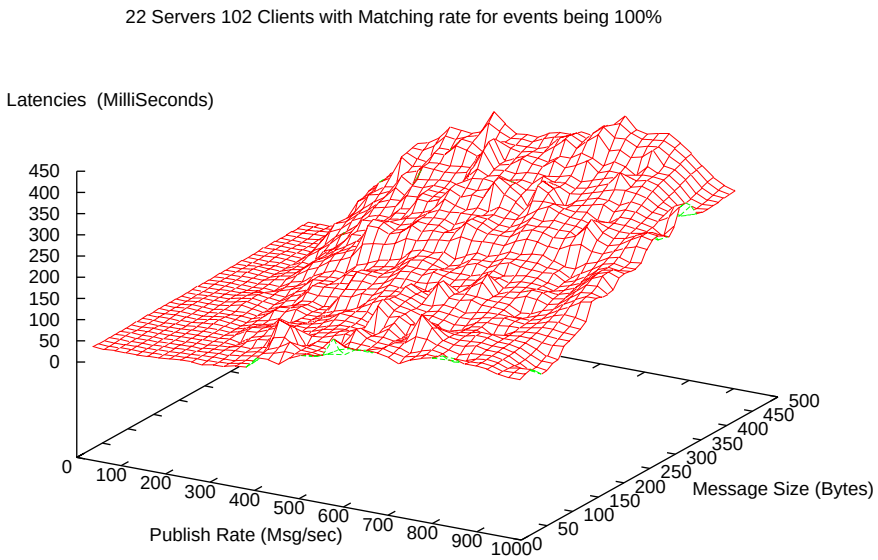

Figure 8.3.1: Match Rates of 100

Figure 8.3.1 depicts the case for matching rates of 100%. In this case the mean latency for the sample of messages varies from 15.54 *mSec* for <1 message/Sec, 50 Bytes> at a throughput of 1 message/Sec to 391.85 *mSec* for <952 messages/Sec, 450 Bytes> with a throughput of 78 messages/Sec at the client. The variance in the sample of messages varies from 2.3684 $mSec^2$ to 69,713.93 $mSec^2$ for the 2 cases respectively. The maximum throughput achieved was 480.76 messages/Sec at publish rates of 492 messages/Sec with messages of size 75 bytes.

Figure 8.3.2 depicts the case for matching rates of 50%. In this case the mean latency for the sample of messages varies from 13.02 *mSec* for <20 messages/Sec, 50 Bytes> to 178.66 *mSec* for <952 messages/Sec, 350 Bytes>. The variance in the sample of messages varies from 56.8196 $mSec^2$ to 14,634 $mSec^2$ for the 2 cases respectively.

Figure 8.3.2: Match Rates of 50

Figure 8.3.3 depicts the case for matching rates of 33%. In this case the mean latency for the sample of messages varies from 15.18 $mSec$ for <20 messages/Sec, 50 Bytes> to 95.969 $mSec$ for <952 messages/Sec, 425 Bytes>. The variance in the sample of messages varies from 26.57 $mSec^2$ to 1,263 $mSec^2$ for the 2 cases respectively.

Figure 8.3.4 depicts the case for matching rates of 25%. In this case the mean latency for the sample of messages varies from 14.40 $mSec$ for <20 messages/Sec, 50 Bytes> to 66.6 $mSec$ for <961 messages/Sec, 400 Bytes>. The variance in the sample of messages varies from 0.24 $mSec^2$ to 587.04 $mSec^2$ for the 2 cases respectively.

Figure 8.3.5 depicts the case for matching rates of 20%. In this case the mean latency for the sample of messages varies from 15.35 $mSec$ for <20 messages/Sec, 50 Bytes> to 62.35 $mSec$ for <952 messages/Sec, 400 Bytes>. The variance in the sample of messages varies from 12.027 $mSec^2$ to 312 $mSec^2$ for the 2 cases respectively.

Figure 8.3.6 depicts the case for matching rates of 10%. In this case the mean latency for the sample of messages varies from 14.40 $mSec$ for <20 messages/Sec, 50 Bytes> to 52.0 $mSec$ for <952 messages/Sec, 400 Bytes>. The variance in the sample of messages varies from 0.44 $mSec^2$ to 103 $mSec^2$ for the 2 cases respectively.

Figure 8.3.7 depicts the case for matching rates of 5%. In this case the mean latency for the sample of messages varies from 14.0 $mSec$ for <20 messages/Sec, 50 Bytes> to 47.6 $mSec$ for <961 messages/Sec, 425 Bytes>. The variance in the sample of messages varies from 0.44 $mSec^2$ to 87.44 $mSec^2$ for the 2 cases respectively.

System Throughput

We also depict the system throughputs at the client under conditions of varying message sizes and publish rates. We choose to depict the system throughputs at a Matching rate of 100% since at other matching rates the only relevant events are being routed to the clients, and thus don't reveal the true throughputs that can be achieved at a client. Figure 8.3.8 depicts the system throughputs achieved at a client under

Figure 8.3.3: Match Rates of 33

conditions of different publish rates and message sizes. varies from 2.3684 $mSec^2$ to 69,713.93 $mSec^2$ for the 2 cases respectively. The maximum throughput achieved was 480.76 messages/Sec at publish rates of 492 messages/Sec with messages of size 75 bytes.

Figures 8.3.9 and 8.3.10 depict the latencies in delivery of events at persistent clients, with singluar and double replications.

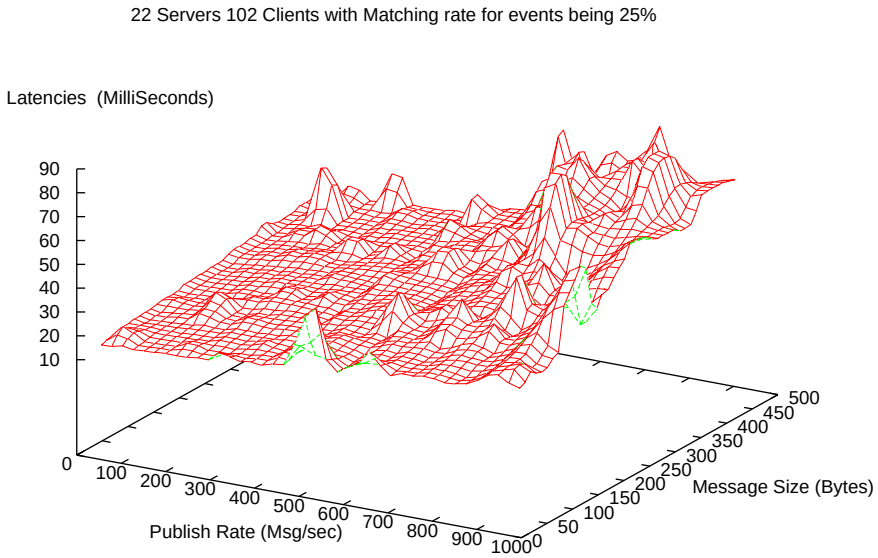

Figure 8.3.4: Match Rates of 25

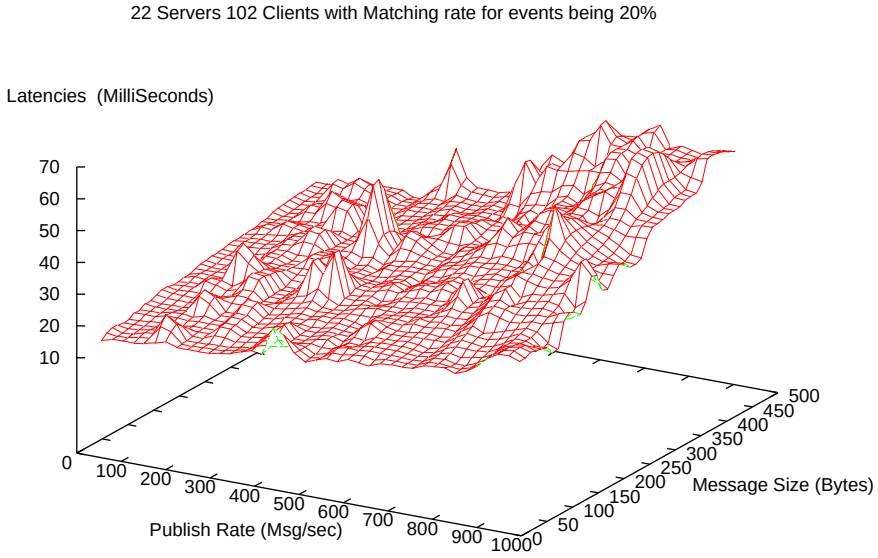

Figure 8.3.5: Match Rates of 20

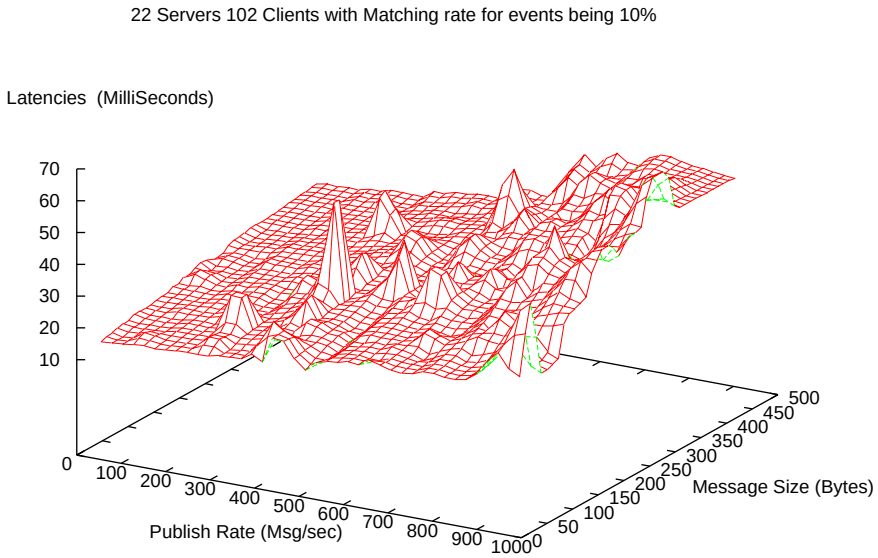

Figure 8.3.6: Match Rates of 10

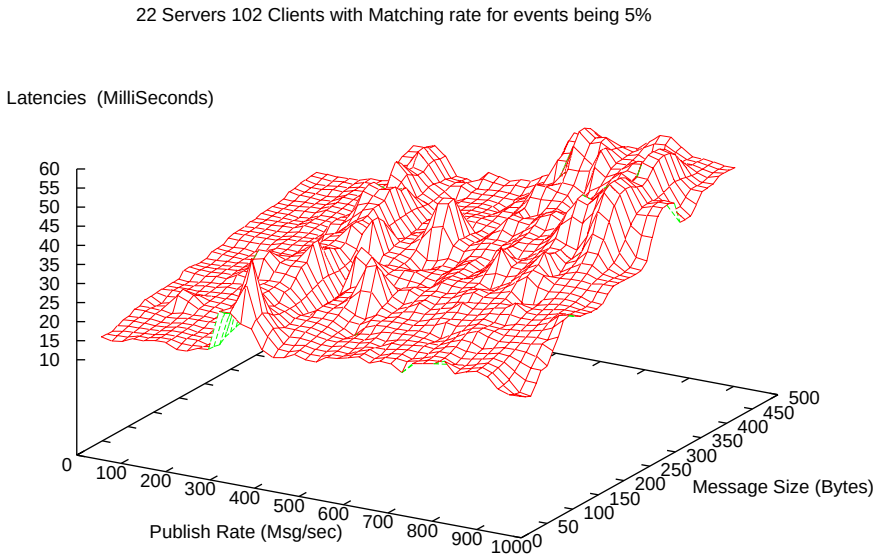

Figure 8.3.7: Match Rates of 5

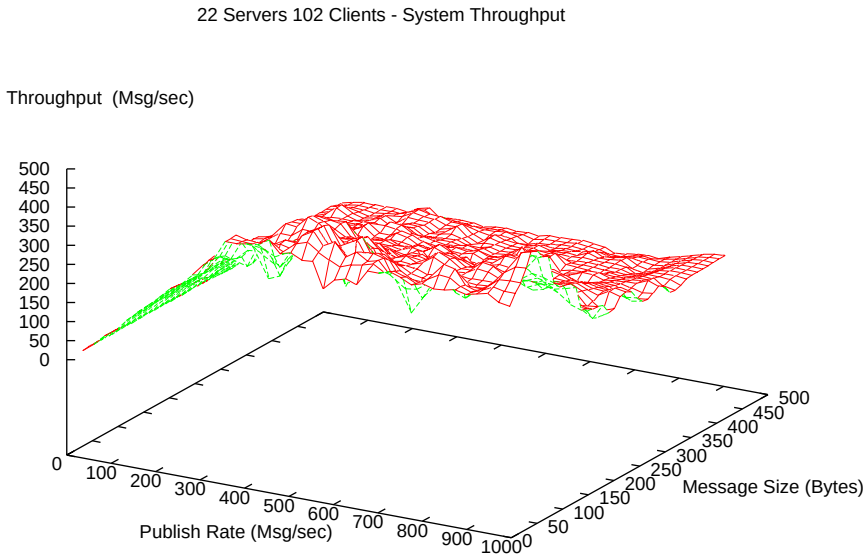

Figure 8.3.8: System Throughput

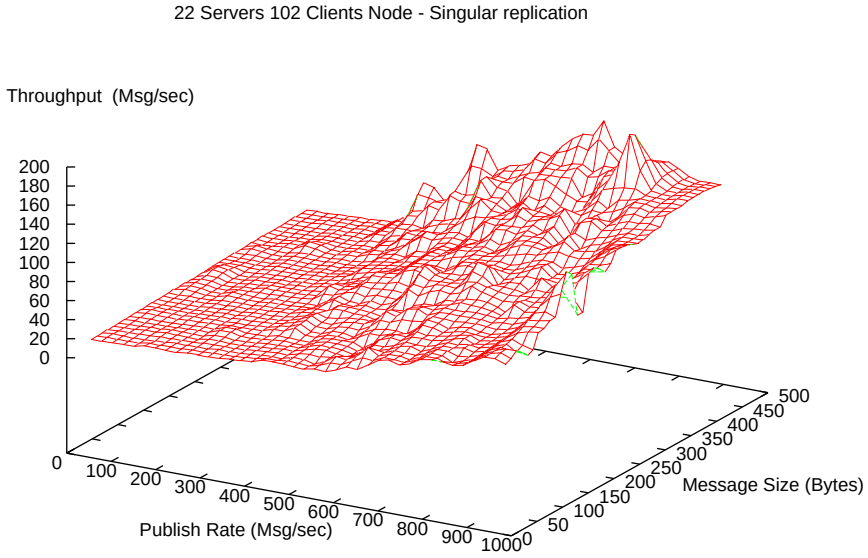

Figure 8.3.9: Match Rates of 50% - Persistent Client

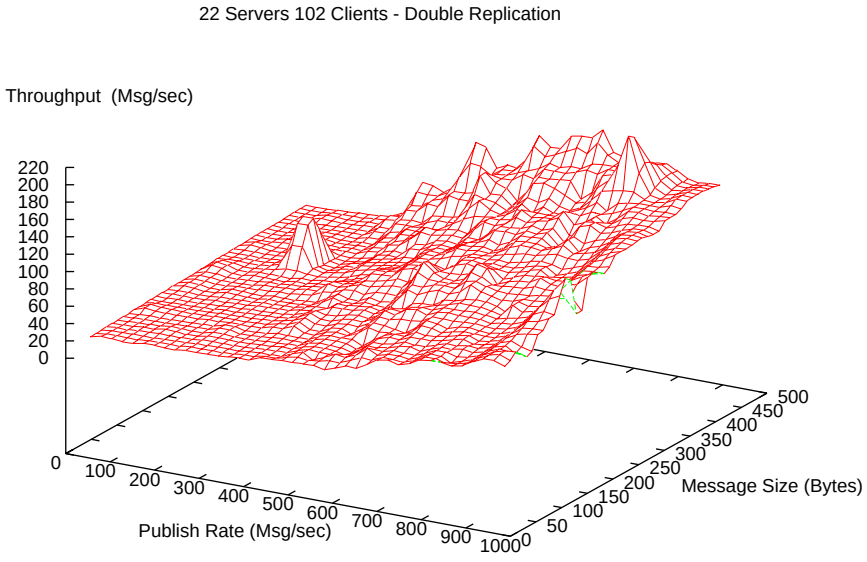

Figure 8.3.10: Match Rates of 50% - Persistent Client