



21497 Salt Creek Rd.
Grass Valley, CA. 95949
916.268.8437
ShilohLLC@aol.com

Scalability Benchmark

Oracle Messaging Server Version 4.2

August, 1999

Executive Summary

A mail messaging benchmark was conducted to determine the performance and scalability of Oracle's Messaging Server, version 4.2, running on a single-node Sun Enterprise 6500 server. The test's objective was to measure the scalability of Oracle's Internet Messaging Server while determining the correlation between a very high rate of message traffic, average transaction and user response times, and server resource utilization.

The benchmark demonstrated

- Over 820,000 mail transactions in one hour (average of 13,700 transactions per minute)
- A concurrency rate of 360,000 users. Using a lighter load model would have pushed the concurrency rate higher.
- Using the 90% percentile for response times all transactions were sub second.
- A message load ranging in message size from 1K to a 140Kb averaging 6.2 Kb.

Shiloh believes that these benchmark results clearly demonstrate that the Oracle Messaging Server can provide excellent performance for a large POP3 or IMAP4 user community or a heterogeneous community of mixed users, as demonstrated in this benchmark.

Results

During the timed, steady state portion of the test (63 minutes), ClientScaler, the test utility, simulated a load of approximately 10,000 IMAP users and 350,000 POP3 users. The simulated users generated a total of 721,818 POP3 get new mail cycles and 40,252 IMAP4 check and fetch cycles. Additionally, 61,367 SMTP messages were sent with each message carbon copied to two users. This equates to a total of 184,101 delivered messages as described in the load profile (see Methodology section).

Based on these measurements, the Oracle Messaging Server sustained 360,000 concurrent users with an average rate of over 13,700 messaging transactions per minute, and a response time of less than 1 second for a typical user sequence of sending or checking/retrieving mail.

The response time of each transaction was measured. The 90th percentile results for the transactions are presented in the Table 2. Table 1 correlates the transaction response rates to corresponding user response time for a typical sequence of sending or checking/retrieving mail. User response time naturally varies depending on the size and number of messages sent or retrieved. Therefore, Table 1 lists three response ranges depending on the message file size being sent or retrieved. These results illustrate that Oracle's messaging server provides excellent response time for a very large user population and range of message file sizes.

Protocol	File Size (Kbytes)	Response (sec)	File Size (Kbytes)	Response (sec)	File Size (Kbytes)	Response (sec)
SMTP	1	.35	Up to 35	.37	141	.4
POP3	1	.57	Up to 5	.6	141	1.1
IMAP4	1	.86	Up to 5	.88	141	1.3

Table 1 User response time (seconds) by sequence of mail operations and file size as described in the Methodology section.

Table 1 response times include the following operations per protocol as described in detail in the Methodology section:

- SMTP message send operation, which includes one recipient with 2 cc's.
- POP3 message retrieval.
- IMAP4 message check and fetch.

Protocol Transaction	90% Percentile Response in msec
IMAP4	
all flags	49
all headers	414
capability	41
connect	5
expunge	7
list	9
login	147
logout	50
namespace	3
noop	77
select	160
imap fetch 1k	90
imap fetch 2k	95
imap fetch 5k	110
imap fetch 35k	188
imap fetch 141k	512
POP3	
pop3 connect	112
pop3 delete	11
pop3 list	58
pop3 pass	174
pop3 quit	21
pop3 stat	7
pop3 uidl	59
pop3 user	7
pop3 retr 1k	129
pop3 retr 2k	143
pop3 retr 5k	159
pop3 retr 35k	319
pop3 retr 141k	673
SMTP	
smtp connect	85
smtp helo	4
smtp mail from	63
smtp quit	16
smtp rcpt to	50
smtp data 1k	138
smtp data 2k	142
smtp data 5k	141
smtp data 35k	154
smtp data 141	283

Table 2 Response time by protocol transaction type in milliseconds.

During the benchmark, server CPU utilization was recorded every 10 minutes. Throughout the benchmark, the combined total of user and system CPU consumption registered approximately 60 to 70 percent, and IDLE time approximately 30 to 40 percent. This indicates that the E6500 was not CPU constrained during the benchmark.

Methodology

The mail message load is generated using an Oracle utility called ClientScaler. This is a general purpose JAVA application that can generate a range of loads representative of various protocols, such as SMTP, POP3, IMAP4, and HTTP, for example. Prior to running the benchmark, Shiloh analyzed the transactions and transaction order that ClientScaler generates for the 3 protocols under test. Shiloh compared the transaction output of ClientScaler to the output of Netscape Communicator versions 4.02 and 4.5. Based on this analysis, Shiloh believes that the load generated by ClientScaler is representative of an actual user load that would be generated by a messaging application such as Communicator for SMTP, POP3 and IMAP4 protocols.

The benchmark consists of a combined message load of multiple sequences of mail operations. For SMTP the basic operation is a message “**send**”. For POP3 the basic operation is “**retrieve**” mail, and for IMAP4 the basic operation is “**check and fetch**” mail. Each operation is described in detail below. All the mail operations are simultaneously generated against the same Messaging Server. This load simulates a heterogeneous user community that is representative of many large mail environments having both concurrent Intranet and Internet mail access.

IMAP4 Client Behavior

The following section describes the client behavior for the simulated IMAP4 clients. The simulated behavior is consistent with a Netscape Communicator 4.5 IMAP4 client.

When the IMAP4 client is started it executes the following sequence of tasks a single time.

```
1  CAPABILITY
2  LOGIN <username> <password>
3  NAMESPACE
4  LIST "" "%"
5  LIST "" "%"
6  LIST "" "/Shared"
7  LIST "" "/Public"
9  LIST "" "INBOX"
10 SELECT INBOX
11 UID FETCH 1:* (FLAGS)
12 UID FETCH 1:* (UID RFC822.SIZE FLAGS BODY.PEEK[HEADER.FIELDS (From
    To Cc Subject Date Priority X-Priority Message-ID References Newsgroups
    Return-Receipt-To Disposition-Notification-To)])
```

After the initial cycle is completed, the client executes cycles in a continual loop until time has expired for the test. Each client executes the loop once every fifteen minutes. The following several blocks of commands describe this primary loop. The exact sequence of commands will vary slightly based on two factors. First, if the user has logged out at the end of the previous cycle, the user must reconnect to the server (This happens 2% of the time.) Second, the user

fetches three messages an average of once every four cycles, so extra commands are executed to perform this work.

If the IMAP4 client has disconnected from the server at the end of the previous cycle, the following commands are executed. During the reconnection sequence the client spawns an extra connection to the messaging server. The sequence of commands executed in this second connection is also described below.

```
13 CAPABILITY
14 LOGIN <username> <password>
15 NAMESPACE
16 LIST "" "%"
17 LIST "" "%"
18 LIST "" "%"
19 LIST "" "/Shared"
20 LIST "" "/Public"
21 LIST "" "INBOX"
22 SELECT INBOX
23 UID FETCH <highest UID>:* (FLAGS)
24 UID FETCH <highest UID>:* (UID RFC822.SIZE FLAGS
BODY.PEEK[HEADER.FIELDS (From To Cc Subject Date Priority X-Priority
Message-ID References Newsgroups Return-Receipt-To Disposition-
Notification-To)])
```

As mentioned above, when the IMAP4 client reconnects to the server, it will spawn an addition background thread. This thread will open its own connection to the protocol server and list each of the subfolders owned by the user. This behavior is consistent with the Netscape 4.5 client. Once all the folders are listed this connection will remain idle until the user logs out or the server terminates this connection. The following are the sequence of tasks executed by this background thread in this test.

```
25 LOGIN <username> <password>
26 LIST "" "Drafts"
27 LIST "" "Templates"
28 LIST "" "Wastebasket"
29 LIST "" "Public Template"
30 LIST "" "Shared"
31 LIST "" "Public"
```

Once the user has reconnected to the server (or was already connected from the previous cycle) the following simple sequence of commands is executed to check for new mail.

```
32 NOOP
33 UID FETCH <highest :UID>:* (FLAGS)
```

Every fourth cycle, three random messages are fetched from the user's inbox. The following commands are executed to perform this.

```
34 FETCH <random msg. #> (UID RFC822.SIZE BODY[])
35 FETCH <random msg. #> (UID RFC822.SIZE BODY[])
36 FETCH <random msg. #> (UID RFC822.SIZE BODY[])
```

As mentioned above, two percent of the time the user will log out at the end of the cycle. The following commands are executed when this occurs.

```
37 EXPUNGE
38 UID FETCH <highest :UID>:* (FLAGS)
39 LOGOUT
```

If the server, as described above, has not timed the background connection out, it is also logged out with a simple LOGOUT command.

POP3 Client Behavior

The following section describes the client behavior for the simulated POP3 clients. The simulated behavior is consistent with a Netscape Communicator 4.5 POP3 client.

When each client is started it executes the following sequence of commands in a loop until the test is complete. In this benchmark each client completed the loop approximately 2.1 times during the timed portion of the test (721,818 POP3 get new mail cycles / 350,000 POP3 users). The only variation between cycles is that the client fetches all messages in his inbox, which will vary depending on the random distribution of messages sent to each user.

```
PASS <password>
STAT
LIST
UIDL
```

For each message in the user's inbox:

```
RETR <uid>
DELE <uid>
```

After all the messages have been received and deleted:

```
QUIT
```

SMTP Client Behavior

The following section describes the client behavior for the simulated SMTP clients. Each cycle simulates sending one message to one user and is carbon copied to two other users. The user selection was randomly chosen among the 360,000 registered accounts. In this benchmark there were approximately 974 SMTP cycles executed per minute (61,367 SMTP message cycles / 63 minutes of measured time).

When each client is started it executes the following sequence of commands in a loop until the test is complete.

```
HELO
MAIL FROM: ClientScaler
RCPT TO: <username>
RCPT TO: <username2>
RCPT TO: <username3>
```

```
DATA:
<message text.  See below for message distribution>
.
QUIT
```

The SMTP profile of messages sent to the server consists of the following distribution and message sizes:

45%	1K messages (1062 bytes per message body)
38%	2K rich text messages (2050 bytes per message body)
10%	5K messages (5321 bytes per message body)
5%	34K rich text messages (34762 bytes per message body)
2%	140K rich text messages (139602 bytes per message body)

The total test time is 2 hours. The first hour is a “warm-up” period in which the ClientScaler applications running on the client machine gradually ramp up the user load until all users are logged in and the system achieves a steady-state level of message flow. The warm-up period includes:

- Starting and ramping the level of SMTP load generated against the server.
- Logging in the IMAP4 users on the server.
- Starting and ramping the level of POP3 client message retrieval against the server

The second hour represents the timed portion of the benchmark. Once a steady state of the target load level is reached, each individual SMTP, POP3 and IMAP4 transaction is timed and recorded for approximately one hour. At the end of the hour the test is stopped and the transaction rate and response time for each is calculated to provide the benchmark results. These results are presented as 90th percentile rates for each protocol transaction type.

Shiloh believes that while it is virtually impossible to define a “typical” messaging load and mail profile, the load and profile used herein is representative of “real-world” rates, message sizes and distributions as seen in many messaging systems today. Therefore, these results should provide users and potential customers insight into the level of performance and scalability that can be supported by Oracle’s Messaging Server.

Hardware Configuration

The test bed configuration shown in Figure 1 consists of the following hardware and software:

Server Platform

Hardware	SUN Enterprise E6500
Operating System	SUN Solaris 2.6
CPUs	22 @ 336 MHz
Memory	16 Gigabytes, 60 Ns, 16 –way interleaved factor
Hard Disk	26 Gigabytes
	Configured as 8 RAID 5 arrays of 34300 Megabytes each
Database Manager	Oracle 8 version 8.05
Messaging Application	Oracle Messaging Server version 4.2 Patchset 3

Client Platform

Hardware	SUN Enterprise E4000
Operating System	SUN Solaris 2.6
CPUs	12 @ 336 MHz
Memory	7 Gigabytes, 60 Ns, 4 –way interleaved factor
Load Generator Application	ClientScaler-0.6 (Oracle in-house developed JAVA utility)

Network

Switch	100 Mbit Ethernet switched network with no other traffic than benchmark transactions
--------	--

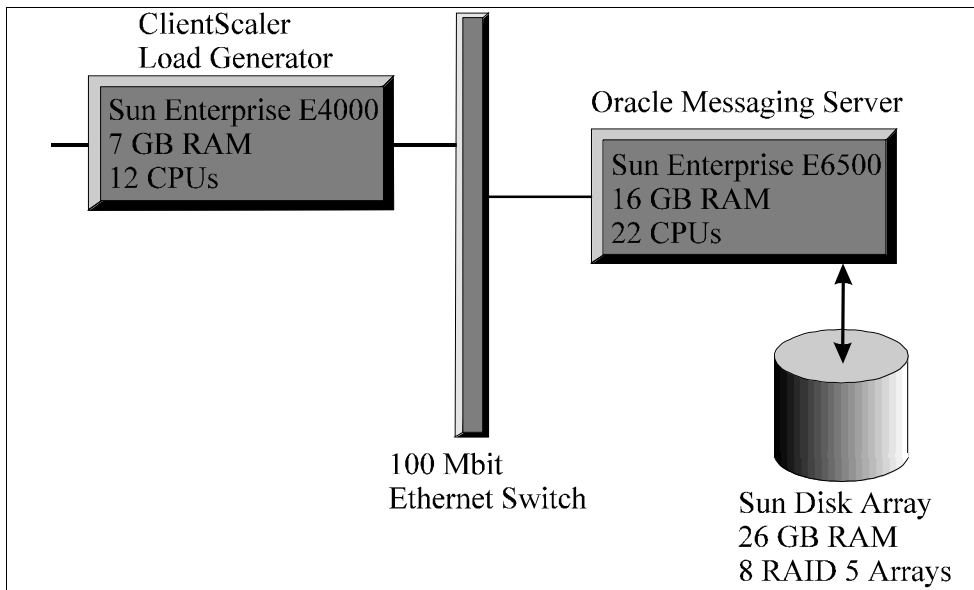


Figure 1 Benchmark test bed configuration.

Conclusions

This benchmark illustrates that Oracle's Internet Messaging Server demonstrates excellent scalability and performance characteristics under heavy transaction loads generated by 360,000 concurrent mail users. Oracle's server sustained an average rate of over 13,700 messaging transactions per minute with a response time of less than 1 second for a typical user sequence of sending or checking/retrieving mail.

The measurements also show that the server provides comparable sub-second response for both POP3 and the more complex IMAP4 transactions, which indicates that Oracle's Messaging Server is a good choice for either internet or intranet mail environments, or for a mixed environment as tested in this benchmark.

About Shiloh Consulting

Shiloh Consulting is an independent network consulting company that provides network, Internet, and Intranet planning, management, and testing services to product manufacturers and Fortune 500 companies. Shiloh is led by Robert Buchanan who has over twenty years of experience in product development, management, and testing for Lockheed, ROLM Corporation, and 3Com Corporation. From 1990 to 1994, Mr. Buchanan ran the testing and

operations of LANQuest Labs, a leading network product testing laboratory. His book, *The Art of Testing Network Systems*, published by John Wiley & Sons in April, 1996, has received strong reviews and sales and was selected by Computer Literacy as “recommended reading.” His second book titled, *Measuring the Impact of Your Web Site*, was released in February, 1997 from Wiley & Sons. It includes case studies of many pioneer web sites and has been translated into Kanji and other languages. Clients include Microsoft, Novell, 3Com, Netscape, Oracle, Compaq, NASA, Sun, CommVision, Mercury Interactive, ImageNet, and other well known companies. Mr. Buchanan can be reached at ShilohLLC@aol.com.