WHITE PAPER

September 1997

Prepared by Internet Solutions Business Unit

Compaq Computer Corporation

CONTENTS

Performance Analysis and Capacity Planning for Microsoft Site Server on Compaq ProLiant Servers

This white paper provides guidelines for planning the configuration of Compaq ProLiant 2500 and ProLiant 7000 computers that run Microsoft® Site Server Internet Locator Server and Personalization System. In order to determine capacity planning guidelines, system performance was measured under various load conditions.

COMPAQ
ECG027.0997

NOTICE

The information in this publication is subject to change without notice.

COMPAQ COMPUTER CORPORATION SHALL NOT BE LIABLE FOR TECHNICAL OR EDITORIAL ERRORS OR OMISSIONS CONTAINED HEREIN, NOR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL.

This publication does not constitute an endorsement of the product or products that were tested. The configuration or configurations tested or described may or may not be the only available solution. This test is not a determination of product quality or correctness, nor does it ensure compliance with any federal, state or local requirements. Compaq does not warrant products other than its own strictly as stated in Compaq product warranties.

Product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

Compaq, ProLiant, and NetFlex are registered with the United States Patent and Trademark Office.

Other product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

Microsoft, Windows, Windows NT, Windows NT Server, Internet Information Server, Site Server, Internet Locator Server, Personalization System, and *perfmon* are trademarks and/or registered trademarks of Microsoft Corporation.

Ethernet is a trademark or registered trademark of Xerox Corporation.

First Edition (September 1997)
Document Number ECG027.0997

INTRODUCTION TO MICROSOFT SITE SERVER

As enterprise customers become more dependent on the Intranet, Internet, and World Wide Web, there is a tendency to get bogged down in the creation and management of web-based applications. Microsoft® Site Server, with its comprehensive publishing and site management capabilities, offers an easy to use and affordable solution to this problem.

Microsoft® Site Server is designed for use in the creation and continuing management of Web-based applications. It is designed to work with Microsoft® Windows NT® Server network operating system (Windows NT) and Microsoft® Internet Information Server (IIS), and offers a sweeping set of server and client components. These include the client software Usage Analyst and Site Analyst, and the server side software Posting Acceptor, Internet Locator Server, Content Replication Server, and Personalization System. These components are used for deployment (Posting Acceptor and Content Replication System), management and analysis (Usage Analyst, Site Analyst, and Content Replication System), and site enhancement (Internet Locator Server and Personalization System). A brief description of each component follows.

Posting Acceptor allows greater freedom with Web content by working with Web Publishing Wizard in the transfer of information to a specified Web server. Posting Acceptor is used on a Web server running Windows NT Server, Windows NT Workstation, or Windows 95. It allows Microsoft Personal Web Server, IIS , and Microsoft Peer Web Services to accept content from Netscape Navigator 2.02 and Microsoft Web Publishing Wizard/API or, later, any standard HyperText Transfer Protocol (HTTP) connection. Posting Acceptor allows users to post Web content to the server, Web content providers to publish their content using HTTP post (RFC1867), posts to be received from other clients that use HTTP-Post protocol, and includes ActiveX control for Web pages which allows the user to drag information to the Web server. It also manages Web pages posted to the server and allows for the transfer of files to a chosen Web server from a client's computer.

Usage Analyst accesses and analyzes data from a Web site's log file in order to increase site efficiency (by identifying seldomly viewed pages, etc.). Usage Analyst reports on usage, visitor profiles, the number of "hits, visits, and page views," or customer interactions with a site, and can cross-reference information to identify useful trends. Usage Analyst is an invaluable tool for maximizing effectiveness in business and commerce. It is supported by Microsoft Access in the standard edition, and also supported by Microsoft® SQL Server in Site Server, Enterprise Edition. Usage Analyst includes over 25 standard analysis reports and enables the analysis of over 25 web server log file formats, including IIS, Netscape, and Apache.

Site Analyst can be used by Webmasters, content authors, and Web server administrators. It creates WebMaps that allow for a quick survey of the site, no matter how complex. It even allows for different views of the site such as the Tree view, which shows the site in a hierarchical structure, and the Hyperbolic view, which maps items in a web-like structure to show their interconnectivity. These views can be helpful in site structure analysis, allowing the user to see the site from a different perspective. Site Analyst also enables analysis of the site structure and content, finds and repairs broken links, and creates reports.

Content Replication System (CRS) is instrumental in automating the Web publishing process and managing network loads through the replication of content across multiple servers. CRS is an NT server-based Internet replication system that uses the authentication mechanism to ensure secure links between CRS servers. It replicates content from one NT server to as many others as desired, either locally, through any area of the Internet, or on any network that supports TCP/IP. CRS automatically replicates information to and from varied servers, using either API functions, Standard Windows NT file transfers, or its own frame file transfer mechanism. CRS can also distribute Web content to multiple remote servers, and is capable of providing administration

through command-line tools or through the Web browser. This allows for the automatic updating of Web farms and makes the CRS a vital tool for the on-line publisher.

Internet Locator Server allows you to find others who are currently on line. It finds information that is constantly changing, such as a user's IP address, and continually updates its information as users log on and off. ILS uses a Random Access Memory database to store the IP addresses of ILS registered users (users are registered when they connect to an ILS server) and can be used on the Intranet, perhaps for project collaboration between coworkers, as well as the Internet, for service capabilities such as Chats. ILS can be used in conjunction with software such as Intel Internet Phone or Microsoft® NetMeeting to connect computers for Internet phone calls or even to set up meetings sharing on line whiteboards. ILS is capable of handling millions of inquiries each day, per server, and is designed specifically for commercial use.

Personalization System allows a site to deliver a personalized experience for each customer, based on their recorded preferences. This personalization creates site loyalty without using programming languages such as C or PERL. Instead, it uses Active Server Pages (ASP) hard scripting (a feature of IIS) to create personalized Web pages that are tailored to individual users. Through the use of this technology, the PS furnishes a server-based profile database, and provides support components and provisions to take full advantage of this database, thus simplifying the creation of the personalized Web site.

Each of these components would individually be a valuable asset to any intranet or internet site. However, the ease of manageability of deploying content through CRS, the enhanced manageability and statistics provided by Site Analyst and Usage Analyst, and the enhanced user experience provided by the Personalization System make Microsoft® Site Server truly outstanding – and unique.

EXECUTIVE SUMMARY

This white paper provides guidelines for planning the configuration of Compaq ProLiant 2500 and ProLiant 7000 computers to run two features from Microsoft® Site Server: Internet Locator Server (ILS) and Personalization System (PS). They were selected because we expect them to be used frequently and because their use will have a significant effect on the performance of a server at times of peak use. Planning for the load these Site Server components will put on a server will allow for the construction of a more responsive, reliable server for the user community.

In order to determine the capacity planning guidelines, system performance was measured under various load conditions. ILS and PS run in conjunction with Microsoft® Internet Information Server (IIS) version 3.0, which is the Web server included in Microsoft® Windows NT® Server 4.0 network operating system (Windows NT).

ILS provides a dynamic directory service that makes it possible to find users who are online. In order to test the performance of ILS it is necessary to simulate many online users on a system. This was accomplished in two ways: first, an Active Server Page (ASP) was used to load ILS with many simulated users; then Microsoft's InetLoad tool was set up to simulate the activity of thousands of users accessing ILS to find other users. The maximum number of online users simulated for testing was 10, 000 for a period of 600 seconds. This is more thoroughly explained in the "ILS Performance Measurement Procedures" section of this document.

PS provides a user property database that simplifies the process of providing personalized Web pages for each user. PS features are made available via Active Server Pages. The performance of PS was tested by creating user property databases of various sizes representing the user communities on a Web server. The maximum simulated community size tested was of 1,000,000 users. Then Microsoft's WCAT (Web Server Analysis Tool) tool was used to simulate thousands of Web browsers accessing personalized Web pages. The tests were run for a total of 15 minutes, with a 5 minute warm-up period and a 10 minute measurement period. This is explained more thoroughly in the "PS Performance Measurement Procedures" section of this document.

For Internet Locator Server (ILS), the test results have shown that both ProLiant 2500 and ProLiant 7000 are highly scalable; a ProLiant 2500 can address approximately 500 connects/sec, 850 adds/sec, and 900 direct lookups/sec and a ProLiant 7000 can address approximately 800 connects/sec, 1300 adds/sec, and 1500 direct lookups/sec for online user communities ranging in size from 500 to 10,000.

For Personalization System (PS), the test results have shown that both ProLiant 2500 and ProLiant 7000 can easily serve personalized Web pages for each customer, for sites with up to one million online users. A Proliant 2500 can address 50 Web pages requests/sec and 50 Active Server pages requests/sec (4.3 million requests/day). A ProLiant 7000 can address 60 Web pages requests/sec (5.2 million requests/day) and 68 Active Server pages requests/sec (5.9 million requests/day).

PERFORMANCE MEASUREMENTS

The goal of capacity planning is to determine the types and configurations of computer systems, networks, and other resources that can support a specified number of users with a precise response time. In order to do this with realistic resources and in a reasonable amount of time, it is necessary to simulate many users interacting with the System Under Test (SUT), in this case the server being tested. The simulation tools used in this study were Microsoft's most recent versions of InetLoad and WCAT, as well as some specially developed Web pages using Microsoft ASP facilities.

The most important performance measurement needed to aid in capacity planning is the maximum transaction rate for a specified load that a computer system can support. A system reaches its maximum transaction rate when its CPU or CPUs are utilized 100% of the time. This metric is called the "platform transaction capacity", and it is highly dependent on the mix of transactions used for its measurement. For example, a Web server that is tested transferring files that average 500 bytes to a simulated browser might have a platform transaction capacity of 2,000 operations per second. Another test of the same computer transferring an average file size of 7,800 bytes, which is a much more realistic average size for Web pages and graphics, would show a platform transaction capacity of 900.

There are two ways to measure platform transaction capacity. The first is to simulate many people using a system with a specified sequence of transactions, including time for a person to read or think about the information returned by the system. InetLoad supports this approach. The second way to measure platform transaction capacity is to generate transactions as quickly as possible, in a random sequence, but with a specified frequency. Both InetLoad and WCAT support this approach.

When measuring platform transaction capacity, it is imperative that the system's CPUs are the limiting factor, not some other aspect of the environment. This clearly requires a controlled and well defined environment. Other limitations, or bottlenecks, can greatly affect measured platform transaction capacity. Microsoft's *perfmon* performance monitoring tool, which comes with Windows NT Server 4.0, was used to help determine the existence and location of bottlenecks. These limitations are discussed in the "Bottlenecks" section of this document.

The InetLoad Tool

InetLoad supports testing multiple protocols, including the Lightweight Directory Access Protocol (LDAP), which is needed for ILS testing, and the HyperText Transfer Protocol (HTTP), which is necessary for PS testing. It works by having worker threads execute scripts on behalf of simulated users. InetLoad has a graphical user interface (GUI) that lets you set the number of users it is simulating and the number of worker threads that cycle through the script on behalf of each simulated user. Experiment with setting the number of users simulated and the number of worker threads. This will help to avoid response time artifacts based on the client system's performance

In order to simulate large numbers of users, one or more copies of InetLoad must be run manually on multiple client systems. Each copy of InetLoad produces its own log file containing command response time information. To determine the response times per command type from multiple InetLoad logs, it is necessary to use Islog, another Microsoft tool, that merges and produces statistics from log files.

The WCAT Tool

The WCAT tool simulates many users accessing a Web server simultaneously. WCAT uses HTTP so that a server acts as though normal Web browsers are using it. A script controls WCAT's execution and allows for the setting of the number of client systems used per test and the number of WCAT threads that run on each client system. Each WCAT client represents a much heavier load than a single person using a Web browser because WCAT does not take the time to understand the information returned. Rather, each WCAT client requests a new file as soon as it has received the previously requested file. Once the WCAT client program is started on each client system, the WCAT controller connects to the number of client systems specified in the script, synchronizes their execution, collects test results from each client system, and generates a consolidated report.

Server Configurations

Table 1 shows the configurations of each ProLiant server tested. The actual amount of memory used for testing was selected at boot time through a configuration parameter in the Windows NT *boot.ini* file.

Table 1: Server Configurations

System	Component	Configuration
Compaq ProLiant 2500	Processor	1 x 200 MHz Pentium Pro (512 KB cache)
	Memory	128 MB ECC
	Network	1 x Compaq NetFlex3, 100Base-TX
	Disk	4 x 2.1GB 7200 RPM disks in a RAID 5 configuration with a SMART-2 controller
Compaq ProLiant 7000	Processor	2 x 200 MHz Pentium Pro (1 MB cache) for ILS testing; one processor for PS testing
	Memory	256 MB EDO ECC
	Network	1 x Dual Port Compaq NetFlex3, 100Base-TX
	Disk	4 x 4.3 GB 10,000 RPM disks in a RAID 5 configuration with a SMART-2 controller

Internet Locator Server (ILS) Performance Measurement Procedures

For the ILS performance measurements, both InetLoad and *perfmon* were used. InetLoad was selected because it uses the Lightweight Directory Access Protocol (LDAP) for ILS testing. LDAP was chosen as a test tool because ILS is implemented as part of Microsoft's LDAP server. In addition, the real-client applications that use ILS, such as NetMeeting, use LDAP to communicate with ILS.

The dynamic nature of ILS, which is a real-time directory service for online users, requires online user simulation on the SUT. This simulation was accomplished with an ASP-enabled Web page and a Web browser that initialized the ILS directory with the desired number of simulated online users. In order to gather performance data that would be useful for capacity planning purposes, four common operations done by applications using ILS were defined. The performance of each operation was tested separately. Table 2 shows the ILS operations measured and which ILS or LDAP commands were used.

Table 2: ILS Operations Measured

ILS Operation	Definition and Comments
Connect	Connection to the ILS directory service. This operation must be done before any application can use ILS and is always done at least once by all ILS applications. The performance of the LDAP command sequence CONNECT, BINDSIMPLE ANONYMOUS, and QUIT was measured to simulate this operation. Note, LDAP requires that CONNECT and a form of BIND are always paired and that a QUIT must be issued before another CONNECT from the same client. ****
Add	Add a new online user to the ILS directory. This operation is done after a Connect operation to make the new user accessible to ILS applications. It is done only once by an ILS application, typically just after the connection is made. We measured performance by doing one LDAP CONNECT and one BINDSIMPLE ANONYMOUS, then issuing 100 LDAP ADD commands, and then issuing an LDAP QUIT command. ****
Direct Lookup	Search the ILS directory for a specified user. It must be done after a Connect operation. This is typically the most frequent ILS operation. We measured performance by doing one LDAP CONNECT and one BINDSIMPLE ANONYMOUS, then issuing 100 ILS SEARCH RETURN ALL commands looking for a random user, and then issuing an LDAP QUIT command. ****

**** Disk performance is not a significant factor affecting ILS performance, and the amount of memory on a system is not a significant factor affecting PS performance.

ILS User Scenario

With a goal of determining how many users the SUT can support, a scenario to represent a typical ILS user was developed (see Table 3).

Table 3: ILS User Scenario

Operation #	ILS Command	Comment
1	Connect	Connect to the ILS server
2	Bind anonymous	Start ILS services
3	Retrieve directory	Get a listing of all online users
4	Direct Lookup	Find a specific user
5	Direct Lookup	Find a specific user
6	Direct Lookup	Find a specific user
7	Direct Lookup	Find a specific user
8	Direct Lookup	Keep this user online
9	Direct Lookup	Keep this user online

Other user scenarios are possible and different scenarios may produce varied platform transaction capacities. For this reason, an Excel spreadsheet computing platform transaction capacities based on different user scenarios has been provided in Appendix A of this paper.

ILS Test Procedures

The basic procedure used for performing the ILS tests was:

1. Automatically start IIS and the LDAP server at SUT boot time. This enables both IIS and LDAP logging.
2. Use the ASP-enabled Web page to create a simulated online user base of the appropriate size.
3. Run *perfmon* on the SUT and have it log performance data to a file, with data being logged to disk, as well as show a real-time chart of the CPU utilization and the appropriate transaction rates.
4. Start the tests running on the client systems. The test tool used by the client systems will log performance data to files.
5. At the end of each test, collect the *perfmon* log from the SUT, the ILS log file, and the client performance data files.
6. Reboot the SUT and retest for the next sized online user base.

All tests were performed on isolated LANs that were not online, with the exception of the test traffic.

All of the ILS tests were run for 600 seconds with the original default number of online ILS users (10,000).

ILS Test Results

Figure 1 summarizes the ILS performance measurements for the ProLiant 2500 while Figure 2 summarizes ILS performance for the ProLiant 7000. They show the number of operations per second for each ILS operation. These figures are based on the measurements presented in Tables 4 through 11.

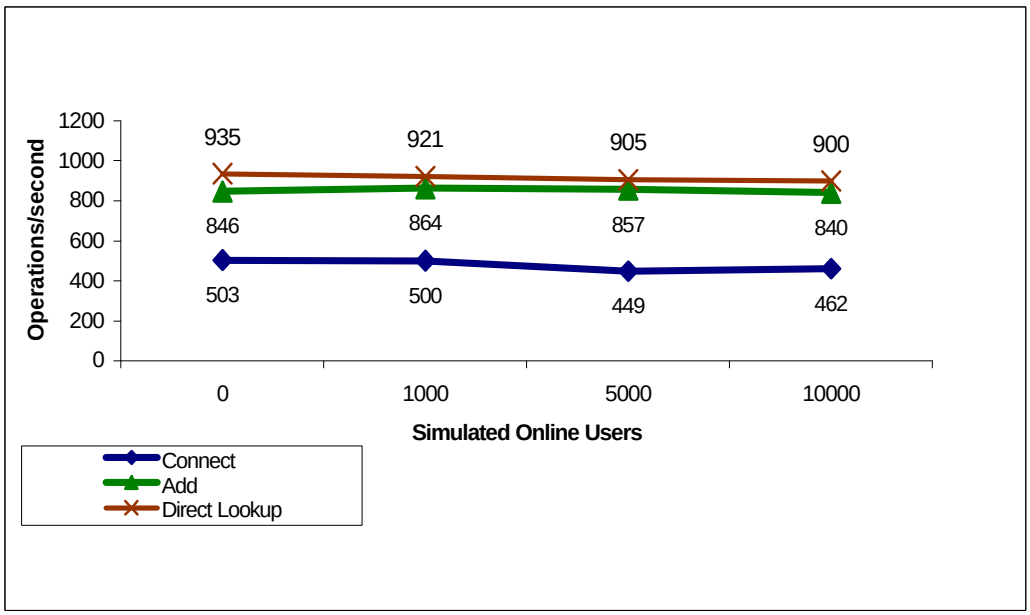


Figure 1: ProLiant 2500 ILS Connect, Add, and Direct Lookup Operations Performance Summary

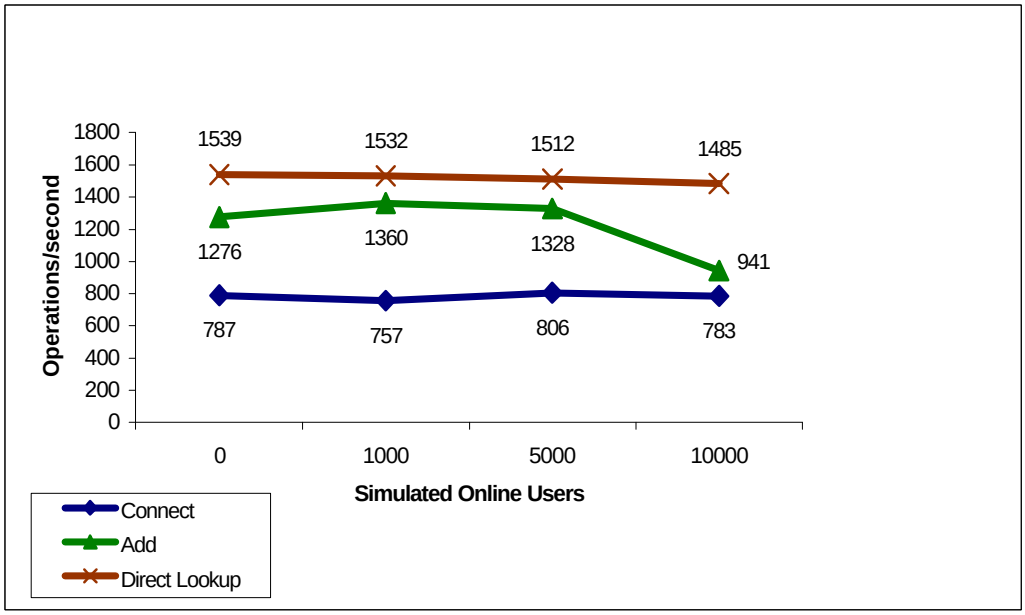


Figure 2: ProLiant 7000 ILS Connect, Add, and Direct Lookup Operations Performance Summary

In Tables 4 through 9, the "Available Memory" *perfmon* measurement is the last sample taken during the test instead of the average. This value was used because it shows the amount of memory available at the end of the test, which gives a good indication as to whether or not performance could be improved by adding memory. The "Connect Ops/sec" measurement in Tables 4 and 7 is computed by dividing the "Connections Received" *perfmon* measurement at the end of the test, less the value at the beginning of the test, by the total time the test ran. This computation was necessary since ILS does not create a *perfmon* ILS Connections/sec measurement.

The "CPU Utilization" measurements for each of the ProLiant 7000's CPUs are shown in Tables 7 through 9, with the "P" indicating the processor number.

Table 4: ProLiant 2500 Connect Operation Perfmon Measurements

Average Perfmon Measurement	500 Users	1,000 Users	5,000 Users	10,000 Users
CPU Utilization	100%	99.9%	99.1%	98.6%
Connect Ops/sec	503	500	449	462
Bytes sent/sec	10 KB	10 KB	9 KB	9 KB
Bytes received/sec	16 KB	17 KB	15 KB	15 KB
Available memory	1.0 MB	1.0 MB	1.8 MB	0.6 MB

Table 5: ProLiant 2500 Add Operation Perfmon Measurements

Average Perfmon Measurement	500 Users	1,000 Users	5,000 Users	10,000 Users
CPU Utilization	100%	100%	100%	100%
Add Ops/sec	815	718	844	741
Bytes sent/sec	33 KB	33 KB	33 KB	34 KB
Bytes received/sec	103 KB	91 KB	106 KB	93 KB
Available memory	65 MB	68 MB	65 MB	67 MB

Table 6: ProLiant 2500 Direct Lookup Operation Perfmon Measurements

Average Perfmon Measurement	500 Users	1,000 Users	5,000 Users	10,000 Users
CPU Utilization	97.7%	97.5%	97.6%	97.9%
Search Ops/sec	935	921	905	900
Bytes sent/sec	1.2 MB	1.1 MB	1.1MB	1.1 MB
Bytes received/sec	97 KB	96 KB	95 KB	94 KB
Available memory	22.8 MB	30 MB	50.1 MB	2.0 MB

Table 7: ProLiant 7000 Connect Operation Perfmon Measurements

Average Perfmon Measurement	500 Users	1,000 Users	5,000 Users	10,000 Users
CPU Utilization	99.9% P0 99.9% P1	92.3% P0 93.3% P1	99.9% P0 99.9% P1	99.9% P0 99.9% P1
Connect Ops/sec	787	757	806	783
Bytes sent/sec	16 KB	15 KB	16 KB	16 KB
Bytes received/sec	26 KB	25 KB	27 KB	26 KB
Available memory	48.2 MB	50.3 MB	37.3 MB	38.3 MB

Table 8: ProLiant 7000 Add Operation Perfmon Measurements

Average Perfmon Measurement	500 Users	1,000 Users	5,000 Users	10,000 Users
CPU Utilization	97.8% P0 99.7% P1	99.0% P0 98.7% P1	98.0% P0 98.3% P1	76.1% P0 75.8% P1
Add Ops/sec	1276	1360	1328	941
Bytes sent/sec	73 KB	72 KB	73 KB	55 KB
Bytes received/sec	161 KB	171 KB	168 KB	119 KB
Available memory	192 MB	194 MB	186 MB	187 MB

Table 9: ProLiant 7000 Direct Lookup Operation Perfmon Measurements

Average Perfmon Measurement	500 Users	1,000 Users	5,000 Users	10,000 Users
CPU Utilization	97.6% P0 97.4% P1	96.7% P0 96.7% P1	97.2% P0 97.2% P1	96.6% P0 96.6% P1
Search Ops/sec	1539	1532	1512	1485
Bytes sent/sec	1.9 MB	1.9 MB	1.9 MB	1.9 MB
Bytes received/sec	160 KB	161 KB	160 KB	157 KB
Available memory	115 MB	114 MB	115 MB	112 MB

ILS Bottlenecks

Server performance depends on many factors. The most important are:

- the type of CPU in the system, its speed, the number of processors, and how much of the CPU is utilized
- the type of network, its speed, the number of Network Interface Cards (NICs), and the device drivers for the NICs
- the amount and type of memory in the system
- the operating system
- the server software
- the disk subsystem

ProLiant 2500 ILS Bottlenecks

Connect, Add, and Direct Lookup: The performance measurements for the ProLiant 2500 show that it is possible to create loads that can fully utilize the CPU. Analysis of Connect, Add, and Direct Lookup Operations follows:

Table 10: Analysis of ProLiant 2500 Operations

Resource	Bottleneck	Comments
CPU	Primary	Fully utilized
Network	No	The total network traffic is less than 3% of the available bandwidth.
Memory	No	The page fault rate indicated by the "Pages/sec" measurement is low.
Operating System	Not enough information	Unless ILS implementation details are known, you cannot determine where the performance bottlenecks are.
ILS Software	Not enough Information	Unless ILS implementation details are known, you cannot determine where the performance bottlenecks are.
Disk Subsystem	No	There is almost no paging and other disk accesses are low.

ProLiant 7000 ILS Bottlenecks

Connect, Add, and Direct Lookup: The performance measurements for the ProLiant 7000 show that it is possible to create loads that can fully utilize the CPU. Analysis of Connect, Add, and Direct Lookup Operations follows:

Table 11: Analysis of ProLiant 7000 Operations

Resource	Bottleneck	Comments
CPU	Primary	Fully utilized
Network	No	The total network traffic is less than 3% of the available bandwidth.
Memory	No	The page fault rate indicated by the "Pages/sec" measurement is low.

Operating System	Not enough information	Unless ILS implementation details are known, you cannot determine whether performance bottlenecks are in the operating system or ILS.
ILS Software	Not enough Information	Unless ILS implementation details are known, you cannot determine whether performance bottlenecks are in the operating system or ILS.
Disk Subsystem	No	There is almost no paging and other disk accesses are low.

Personalization System (PS) Performance Measurement Procedures

Testing PS requires a tool that uses the standard HyperText Transfer Protocol (HTTP) of the Web because PS is accessed via ASP-enabled Web pages served by IIS. With this in mind, both WCAT and *perfmon* were used for the PS performance measurements. WCAT was selected because it can create a reliable, reproducible SUT load using few clients, and because it summarizes the performance statistics gathered from all of the client systems. While InetLoad seems better suited to answer the question "How many users can the SUT support," the current release, Release 2.0, is sufficiently weak in reporting and unstable enough to merit the use of WCAT at this time.

The performance of an SUT will depend on the size of the user community that it serves because PS keeps a database of user information for personalizing Web pages. A User Property Database (UPD) was created on the SUT to represent the size of the user community of interest. This was done using an ASP-enabled Web page and a Web browser to initialize a UPD for the number of users desired. UPDs of various sizes were also created. Which one was used for testing was managed by renaming them to the name specified via the User Property Database Administration Tool, which is provided as part of the PS. The PS performance tests were run on an SUT for user communities of various sizes.

PS User Scenario

With the goal of determining how many users the SUT can support, a scenario representing a typical user at a 100% personalized Web site was developed (see Table 12). That is, each Web page was delivered to the user via an ASP that accessed the UPD and customized the Web page accordingly. Such a Web site represents a worst-case scenario for measuring PS performance, since a Web site with less personalization will be able to support higher transaction rates because it does not access as many ASP-enabled Web pages.

Table 12: PS User Scenario

Operation #	Operation	Operation Frequency	Comment
1	Create a new UPD entry	5%	5% of the users will be new, so create a new UPD entry for each of them.
2	Get a personalized Web page	91%	Each Web page will consist of an HTTP GET of the .asp file and 4 HTTP GETs of .gif files. Each .asp file accessed will be 8,290 bytes long and each .gif file will be 5,480 bytes long.
3	Get a personalized Web page and update personalization information	4%	This does the same HTTP GETs as does Operation #2 and also updates 2 properties in the UPD for the simulated user.

As mentioned, other user scenarios are possible and may lead to varied platform transaction capacities. An Excel spreadsheet has been provided in Appendix A to address this issue.

PS Test Procedures

Before starting any tests, UPDs were created to simulate user communities of 1,000, 10,000, 100,000, and 1,000,000 people. Then, the following basic PS test procedures were used:

- At SUT boot time, automatically start IIS. IIS logging is enabled.
- Rename the UPD with the appropriate-sized user community to the active one used by PS.
- Run *perfmon* on the SUT and have it log the processor, Active Server Pages, HTTP Service, memory, and User Property Database performance data to a file, as well as show a real-time chart of the CPU utilization and the appropriate transaction rates. The Network Interface and Physical Disk may also need to be monitored, depending on configuration performance.
- Start the tests running on the client systems. The WCAT collects performance statistics and logs them to a file.
- At the end of each test, collect the *perfmon* log from the SUT, the PS log file, and the client performance data file.
- Reboot the SUT and test for the next-sized user community.

All tests were done on isolated LANs that were inactive except for the test traffic. Tests were run with a five minute warm-up period and a 10 minute measurement period using WCAT.

PS Test Results

Figure 3 summarizes the ProLiant 2500 PS performance measurements shown in Table 13. Similarly, Figure 4 summarizes PS performance for the ProLiant 7000 based on the data in Table 14.

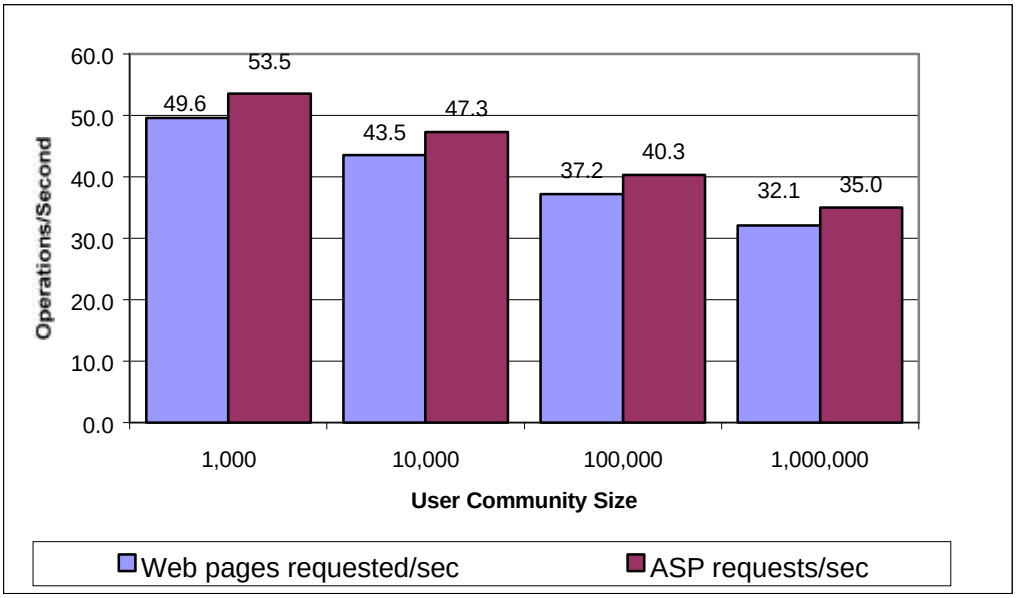


Figure 3: ProLiant 2500 PS Performance Summary

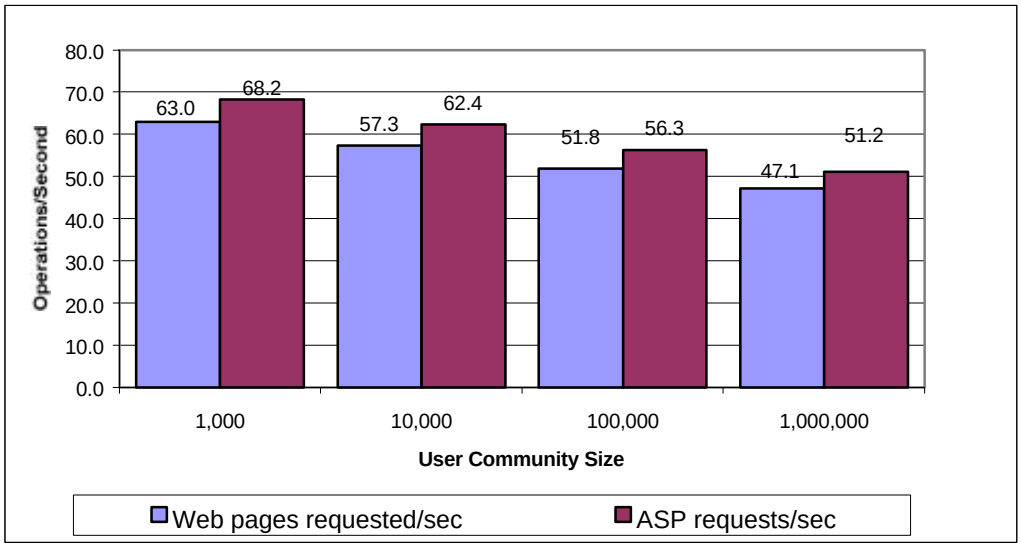


Figure 4: ProLiant 7000 PS Performance Summary

All measurements in Tables 13 and 14 come from *perfmon*, except for the "Web pages requested/sec" and the "Average Connect Time (sec)" measurements, which come from WCAT. The "Available Memory" measurement is the last sample taken during the test. This value was used because it shows how much memory is available at the end of the test. This measurement, gives a good indication as to whether or not performance could be improved by adding more memory.

Table 13: ProLiant 2500 PS Performance Measurements

Measurement	1,000 Users	10,000 Users	100,000 Users	1,000,000 Users
CPU Utilization	96.7%	91.5%	85.0%	77%
Web pages requested/sec	49.6	43.5	37.2	32.1
Average Connect Time (sec)	2	1	1	1
ASP requests/sec	53.5	47.3	40.3	35.0
UPD reads/sec	53.5	47.3	40.3	35.0
Bytes sent/sec	1.5 MB	1.3 MB	1.1 MB	1.0 MB
Bytes received/sec	8 KB	7 KB	6 KB	5 KB
Available memory	96.2 MB	92.2 MB	88.3 MB	83 MB

Table14: ProLiant 7000 PS Performance Measurements

Measurement	1,000 Users	10,000 Users	100,000 Users	1,000,000 Users
CPU Utilization	93.9%	96.4%	89.0%	83.3%
Web pages requested/sec	63.0	57.3	51.8	47.1
Average Connect Time (sec)	2	1	1	1
ASP requests/sec	68.6	62.4	56.3	51.2
UPD reads/sec	68.6	62.4	56.3	51.2
Bytes sent/sec	1.9 MB	1.8 MB	1.6 MB	1.4 MB
Bytes received/sec	10 KB	10 KB	9 KB	8 KB
Available memory	222 MB	213 MB	215 MB	213 MB

PS Bottlenecks

As with ILS, the performance of a server running PS depends on many factors. The most important are:

- the type of CPU in the system and its speed, the number of processors, and how much the CPU is utilized
- the type of network and its speed, the number of network interface cards (NICs), and the device drivers for the NICs
- the amount and type of memory in the system
- the operating system
- the web server software (IIS)
- Personalization Software
- the disk subsystem

ProLiant 2500 PS Bottlenecks

Analysis of PS bottlenecks on the ProLiant 2500 shows:

Table 15: ProLiant 2500 Analysis of PS Bottlenecks

Resource	Bottleneck	Comments
CPU	No	The CPU utilization decreases as the number of users in the UPD increases.

Network	No	The total network traffic is less than 15% of the available bandwidth.
Memory	No	There is between 87 MB and 96 MB of memory available. The "Pages/sec" measurement shows a high page fault rate with larger UPDs. The cause is most likely that PS is using memory-mapped files. The Physical Disk *perfmon* measurements support this conclusion because the disks are busy four times as much with reads than they are with writes.
Operating System	No	
Web Server Software (IIS)	Possible	The web server may be a bottleneck as the number of ASP-enabled web pages requested increases.
PS Software	Possible	PS may be a bottleneck as the number of concurrent users increases.
Disk Subsystem	Yes	See the analysis for the PS Software above

ProLiant 7000 PS Bottlenecks

Analysis of PS bottlenecks on the ProLiant 7000 shows:

Table 16: ProLiant 7000 Analysis of PS Bottlenecks

Resource	Bottleneck	Comments
CPU	No	The CPU utilization decreases as the number of users in the UPD increases.
Network	No	The total network traffic is less than 15% of the available bandwidth.
Memory	No	There is over 200 MB of memory available. The "Pages/sec" measurement shows a high page fault rate with larger UPDs. The cause is most likely that PS is using memory-mapped files. The Physical Disk *perfmon* measurements support this conclusion because the disks are busy four times as much with reads than they are with writes.
Operating System	No	
Web Server Software (IIS)	Possible	The web server may be a bottleneck as the number of ASP-enabled web pages requested increases.
PS Software	Possible	PS may be a bottleneck as the number of concurrent users increases.
Disk Subsystem	Yes	See the analysis for the PS Software above

Test Lab

The test lab configuration used for the ProLiant 2500 testing is shown in Figure 5. The number of client systems needed was determined by adding one at a time until the ProLiant 2500's CPU was driven to 100% utilization. Table 17 shows the actual number of client systems used and the configurations for each test. There was always one network for the ProLiant 2500 tests.

Table17: ProLiant 2500 Test Lab Configuration

Test	# of Client Systems	# of Simulated Users and Worker Threads per Client System
ILS Connect	5	50
ILS Add	2	20
ILS Direct Lookup	4	50
All PS Tests	3	11

Figure 6 shows the test lab configuration used for the ProLiant 7000 testing. The number of client systems and networks was also determined by experimentation until the CPUs were 100% utilized. Table 18 shows the actual number of client systems used and the configurations for each test. There were always two networks used in the ProLiant 7000 tests.

Table18: ProLiant 7000 Test Lab Configuration

Test	# of Client Systems	# of Simulated Users and Worker Threads per Client System
ILS Connect	8	50
ILS Add	4	20
ILS Direct Lookup	8	50
All PS Tests	3	11

Each of the PC client systems ran either InetLoad or WCAT, depending on the tests being done. They communicated with the SUT over one or two 100-BaseTX Ethernets. When running WCAT, the WCAT client programs communicated with the WCAT controller over the same 100-BaseTX Ethernet(s) used to communicate with the SUT. Using the same network for both controlling WCAT clients and testing the SUT does not interfere with the test measurements because the control communications occur only before a test starts and after it is over. There are no WCAT control communications during a test.

The PC client systems were 200MHz Pentium Pro systems. All of the systems used the same operating system: Windows NT Server 4.0 with Service Pack 3. The SUT also had the TCP/IP driver hot fix from June 1997 and Microsoft's Internet Information Server (IIS) version 3.0 installed.

Figure 5: ProLiant 2500 Test Lab Configuration

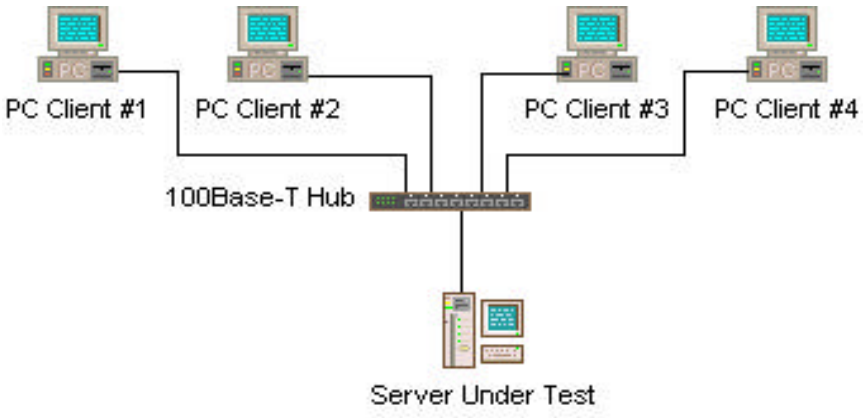
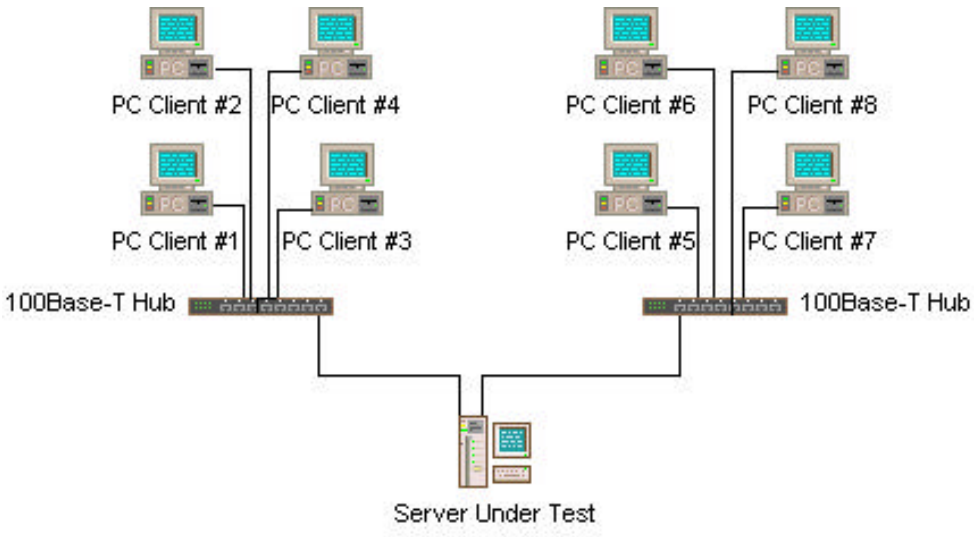


Figure 6: ProLiant 7000 Test Lab Configuration



CAPACITY PLANNING GUIDELINES

The successful installation of any networked application requires careful planning. To get the most out of these components, the installation must offer sufficient capacity to satisfy the user community.

An Excel workbook has been provided in Appendix A of this white paper to help determine how many users each site can support. This workbook helps to determine how many simultaneous users a server can support and what size user community that represents. By doing a "what-if" analysis using this workbook, which Compaq computers, and how many are needed to support the specified number of users with a given response time, can be determined.

Key guidelines for implementing ILS that will help to avoid performance bottlenecks are:

- You must have enough CPU processing power to handle the expected user transaction rate. Remember that the mix of ILS operations plays a very significant role in determining how many transactions per second a system can process.
- You must have enough memory in the system to reduce or eliminate paging. This will let the system handle the number of simultaneously active users expected. Remember that ILS is a memory-resident directory service and does not store its information on disk.
- Be sure that the network bandwidth is sufficient to handle the projected transaction rate. If the server will use a 100 Mbit/second network, then bandwidth will not be a bottleneck. However, at least two 10 Mbit/second networks will be needed to handle the maximum transaction rate supported by either a ProLiant 2500 or ProLiant 7000.

Disk performance is not a significant factor affecting ILS performance.

Key guidelines for implementing PS that will help to avoid performance bottlenecks are:

- Use a RAID, either RAID 1 or 5, if the active user community surpasses a few thousand people (PS can be disk intensive). Raid 5 was used in this implementation of ILS and PS.
- The higher performance of CPUs will increase the number of users the installation can support, as well as lower the response time each sees.

- Be sure that the network bandwidth is sufficient to handle the projected transaction rate. If the server will use a 100 Mbit/second network, bandwidth will not be a bottleneck. However, at least two 10 Mbit/second networks will be needed to handle the maximum transaction rate supported by either a ProLiant 2500 or ProLiant 7000. If a 10 Mbit/second network is used, a bottleneck is possible, depending on the load one plans to generate in the environment.

The amount of memory on a system is not a significant factor affecting PS performance, as long as it is at least 128MB.

Please note that these guidelines are based on the functionality supported by ILS and PS. Other functions supported by a potential Web server must also be taken into account to provide for adequate capacity planning.

CONCLUSION

This Whitepaper has provided guidelines for the configuration of Compaq ProLiant 2500 and ProLiant 7000 computers running the Site Server Internet Locator Server and Personalization System. ILS and PS performance measurements for both the ProLiant 2500 and 7000 have shown that they were capable of generating loads that fully utilize their CPUs. Microsoft Site Server, on Compaq ProLiant 2500 and 7000, delivers the return on investment businesses are looking for from their Intranet and Internet sites.

Above all, remember that the successful installation of any networked application requires careful planning. Utilizing these products to their full potential requires sufficient capacity to satisfy the user community.

APPENDIX A: EXCEL WORKBOOK

An Excel workbook is included in the online version of this White Paper for your use in determining how many of your users can use Internet Locator Server in Compaq ProLiant 2500 and ProLiant 7000 servers configured like those tested in this White Paper.

Click here to launch the online workbook:

ftp://ftp.compaq.com/pub/supportinformation/papers/ecg0270997.xls

A printed example follows for use as a reference.

Capacity Planning Workbook for Microsoft Site Server

Capacity Planner for Internet Locator Server

This workbook will help you determine how many of your users can use Internet Locator Server on Compaq ProLiant 2500 and ProLiant 7000 configured like those tested in the accompanying white paper.

You can use this workbook to plan the capacity of your own system by entering the appropriate data in the "Your System" worksheet.

1. Enter the values for the assumptions below in the space provided:

ILS User Scenario

A. Percentage of your entire user community active at any one time (Enter as a 12.3 for 12.3%. Typical values are 4% or less. Peak values may be 10% or more.)	4.00%
B. Time in minutes of a typical user session	20
C. Number of ILS Connect operations per user session (This is usually 1.)	1
D. Number of ILS Add operations per user session (This is usually 1.)	1
F. Number of ILS Search operations per user session	6

(This must include the number of refresh operations that keep the user active. A refresh will typically happen every 10 minutes; however, this may vary by application and configuration.)

2. Below are the number of users that can be supported by a ProLiant 2500 and a ProLiant 7000

ProLiant 2500 ILS User Capacity

A. Number of simultaneous active users that can be supported	**3,946**
B. Size of the user community that can be supported	**98,639**

ProLiant 7000 ILS User Capacity

A. Number of simultaneous active users that can be supported	**7,842**
B. Size of the user community that can be supported	**196,052**

Your System's ILS User Capacity

A. Number of simultaneous active users that can be supported	**7,842**
B. Size of the user community that can be supported	**196,052**

Capacity Planner for Personalization System

1. Enter the values for the assumptions below in the space provided:

PS User Scenario

A. Percentage of your entire user community active at any one time (Enter as a 12.3 for 12.3%. Typical values are 4% or less. Peak values may be 10% or more.)	4.00%
B. Time in seconds a typical user will take to look at an average Web page	25

C. Number of PS-enabled Web pages your server can deliver per second 43.5

(This number is usually determined by benchmarking. To get an estimate for a ProLiant 2500 enter 43.5 and for a ProLiant 7000 enter 57.3. The accompanying white paper will give you other values for these systems. You can enter any number that is valid for

PS Server User Capacity

A. Number of simultaneous active users that can be supported **1,088**

B. Size of the user community that can be supported **27,188**

ProLiant 2500 ILS Data

Operations/sec data - You may change the table below, but replace its values in the same positions.

	Active users	500	1,000	5,000	10,000
	Connect	503	500	449	462
	Add	846	864	857	840
	Search	935	921	905	900

ProLiant 7000 ILS Data

Operations/sec data - You may change the table below, but replace its values in the same positions.

	Active users	500	1,000	5,000	10,000
	Connect	787	757	806	783
	Add	1276	1360	1328	941
	Search	1539	1532	1512	1485

Try Your System Here

Enter your operations/sec data - You may change the table below, but replace its values in the same positions.

	#of Users in Directory	500	1,000	5,000	10,000
	Connect	787	757	806	783
	Add	1276	1360	1328	941
	Search	1539	1532	1512	1485

APPENDIX B: GLOSSARY

The following terms are used in this white paper:

ASP: Active Server Page, a feature of Microsoft® Internet Information Server

Clients: The number of processes or threads simultaneously requesting Web services from the server.

Client Systems: The computers used to generate a load on an SUT.

CRS: Content Replication System, a feature of Microsoft® Site Server

HTTP: Hyper Text Transfer Protocol

IIS: Microsoft® Internet Information Server

ILS: Internet Locator Server, a feature of Microsoft® Site Server

LDAP: Lightweight Directory Access Protocol

NICs: Network Interface Cards

Platform Transaction Capacity: The maximum transaction rate under a specified load that a computer system can support.

PS: Personalization System, a feature of Microsoft® Site Server

SUT: System Under Test.

UPD: User Property Database

WCAT: Web Server Analysis Tool