Contents

License Agreement

Goal of This Manual

What's New in ServerBench 3.0

Chapter 1 ~ Running ServerBench's Standard Test Suites

Chapter 2 ~ Looking at Your Test Results

Chapter 3 ~ Publishing Your ServerBench Results

Chapter 4 ~ Creating Test Suites

Chapter 5 ~ Using ServerBench's Windows and Menus

Chapter 6 ~ Contacting ZDBOp

ServerBench's Problem Report Form

ServerBench's Benchmark Request Form

Benchmark Request Form

LICENSE AGREEMENT FOR ZIFF-DAVIS' SERVERBENCH® VERSION 3.0

READ THIS AGREEMENT CAREFULLY BEFORE USING THE SOFTWARE EMBODIED IN THE SERVERBENCH® 3.0 CD-ROM, DISKETTE(S), OR, IF PRELOADED ON YOUR HARD DRIVE, DOWNLOADED OR IF PROVIDED AS PART OF A COLLECTION, THE PRELOADED, DOWNLOADED OR COLLECTED FILE(S) (the "Media"). Embodied in the ServerBench 3.0 Media is the ServerBench version 3.0 computer programs and related documentation (the "Software"). Ziff-Davis Publishing Company, having a place of business at One Park Avenue, New York, New York 10016 ("Ziff-Davis") is the licensor under this Agreement and you are the licensee. By using the Software, in whole or in part, you agree to be bound by the terms of this Agreement. If you do not agree to the terms of this Agreement, promptly return the Software to the Ziff-Davis Benchmark Operation at 1001 Aviation Parkway, Suite 400, Morrisville, North Carolina 27560 (or, if downloaded, delete the Software, or if preloaded on your hard drive or if provided as part of a collection, cease use of the Software). Title to the Software and all copyrights, trade secrets and other proprietary rights therein are owned by Ziff-Davis. All rights therein, except those expressly granted to you in this Agreement, are reserved by Ziff-Davis.

1. Limited License

This Agreement grants you only limited rights to use the Software. Ziff-Davis grants you a non-exclusive, non-transferable license to use the Software on a file server networked with multiple PC computers for the sole purpose of conducting benchmark tests to measure the performance of computer hardware and operating system configurations. You have the right to make a single copy of the Software for archival purposes and the right to transfer a copy of the Software across an internal local area network only to the PC computers attached to such network; provided, however, that all such copies are considered Software hereunder, that all uses of such copies are governed by the terms and conditions of this Agreement and that you shall be responsible for all uses of such copies in violation of the terms and conditions of this Agreement. Ziff-Davis hereby grants you the right to publish, except in any country where a third party claims during the term of this license that such distribution infringes that party's proprietary rights, benchmark test results obtained by you from your use of the Software provided that with the publication of each such result you:

A. Identify Ziff-Davis, the name and version number of the benchmark Software used and the name of the test suite used in the test (e.g., Ziff-Davis' ServerBench® version 3.0 with the standard system test suite SYS_60.TST);

B. Identify for the applications server the exact name, number of processors, processor speed (including clock speed) and type(s), amount of RAM, amount of secondary RAM cache (if any), size of hardware hard disk cache, (if any), number and type of hard disk controller(s), type of I/O bus, number and type of hard disks, hard disk size, driver version of the disk controller(s), type of disk organization (e.g., mirrored), number and type of network controllers, driver version of network controller, network operating system name and version, and any relevant modifications to the default network operating system parameters (e.g., WXY Corp. Model 466 with 1 66-MHz Intel® 486DX2-66 CPU, 64MB of RAM, 256KB CPU cache, WXY Integrated Drive Array disk controller, EISA I/O bus, 2 ABC 520MB disk drives, a xxxxx386.dsk 12,621 4/29/92 disk driver, hardware striping, WXY 32-bit ABC controller, a xxxxxxx.lan 40,905 9/11/92 net driver, NetWare® 4.1, and the following NOS parameters: set maximum physical receive packet size = 4202);

C. Identify for the test bed the network type, the number of clients, the client operating system version (e.g., Windows® 95), the number and type of hubs/concentrators, the number of clients per segment, the client CPU type and speed in percentages, client network software name and version (drivers and protocols), the size of the client network cache, if any (e.g., 10Base-T with 32 clients, 2 XYZ Ethernet Hubs, 16 clients per segment, 75% of the clients are 486/25 and 25% of the clients are 386/20, 75% of the clients have ABC NE2000 network cards and 25% of the clients have GHI network cards, Microsoft® Windows 95, Microsoft TCP/IP using Windows Sockets Version 1.1, enhanced mode 32-bit NDIS driver);

D. Identify the controller operating system version (e.g., Microsoft Windows 95) and network software and version;

E. State that all products used in the test were shipping versions available to the general public;

F. State that the test was performed without independent verification by Ziff-Davis and that Ziff-Davis makes no representations or warranties as to the results of the test; and

G. Follow proper trademark usage and acknowledge Ziff-Davis' trademark rights (e.g., "[ ] achieved a ServerBench® overall score of X transactions per second. ServerBench is a registered trademark or trademark of Ziff-Davis Publishing Company in the U.S. and other countries.").

This Agreement and your rights hereunder shall automatically terminate if you fail to comply with any provision of this Agreement. Upon such termination, you agree to cease all use of the Software, cease the transfer of any copies of the Software and cease the publication of benchmark test results obtained by you from use of the Software. Further, you agree to delete the Software and to destroy all tangible copies of the Software and other materials related to the Software in your possession or under your control, or, if downloaded or preloaded on your hard drive or if provided as part of a collection, to cease use of and destroy any and all copies of the Software in your possession or under your control.

2. Additional Restrictions

A. You shall not (and shall not permit other persons or entities to) rent, lease, sell, sublicense, assign, or otherwise transfer the Software or this Agreement. Any attempt to do so shall be void and of no effect.

B. You shall not (and shall not permit other persons or entities to) reverse engineer, decompile, disassemble, merge, modify, include in other software or translate the Software, or use the Software for any commercial purposes, except for the publication of test results, as provided above.

C. You shall not (and shall not permit other persons or entities to) remove or obscure Ziff-Davis' copyright, trademark or other proprietary notices or legends from any of the materials contained in this package or downloaded.

D. You acknowledge that the Software contains Ziff-Davis' trade secret information and you agree not to disclose or disseminate such information other than as provided herein.

3. Limited Warranty and Limited Liability

THE SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE SOFTWARE IS ASSUMED BY YOU, AND ZIFF-DAVIS AND ITS AUTHORIZED DISTRIBUTORS ASSUME NO RESPONSIBILITY FOR THE ACCURACY OR APPLICATION OF OR ERRORS OR OMISSIONS IN THE SOFTWARE. IN NO EVENT SHALL ZIFF-DAVIS OR ITS AUTHORIZED DISTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE, EVEN IF ZIFF-DAVIS OR ITS AUTHORIZED DISTRIBUTORS HAVE BEEN ADVISED OF THE LIKELIHOOD OF SUCH DAMAGES OCCURRING. ZIFF-DAVIS AND ITS AUTHORIZED DISTRIBUTORS SHALL NOT BE LIABLE FOR ANY LOSS, DAMAGES OR COSTS, ARISING OUT OF, BUT NOT LIMITED TO, LOST PROFITS OR REVENUE, LOSS OF USE OF THE SOFTWARE, LOSS OF DATA OR EQUIPMENT, THE COSTS OF RECOVERING SOFTWARE, DATA OR EQUIPMENT, THE COST OF SUBSTITUTE SOFTWARE OR DATA, CLAIMS BY THIRD PARTIES, OR OTHER SIMILAR COSTS.

THE ONLY WARRANTY MADE BY ZIFF-DAVIS AND ITS AUTHORIZED DISTRIBUTORS IS THAT ANY ORIGINAL PHYSICAL MEDIA IN WHICH THE SOFTWARE IS EMBODIED AND WHICH IS DISTRIBUTED BY ZIFF-DAVIS OR ITS AUTHORIZED DISTRIBUTORS SHALL BE FREE OF DEFECTS IN MATERIALS AND WORKMANSHIP. ZIFF-DAVIS' AND ITS AUTHORIZED DISTRIBUTORS' ENTIRE LIABILITY AND THE USER'S EXCLUSIVE REMEDY SHALL BE LIMITED TO THE REPLACEMENT OF THE ORIGINAL PHYSICAL MEDIA IF DEFECTIVE. THE WARRANTIES AND REMEDIES SET FORTH HEREIN ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. NO ZIFF-DAVIS DISTRIBUTOR, AGENT OR EMPLOYEE, OR THIRD PARTY, IS AUTHORIZED TO MAKE ANY MODIFICATION OR ADDITION TO THIS WARRANTY.

SOME STATES DO NOT ALLOW EXCLUSION OR LIMITATION OF IMPLIED WARRANTIES OR LIMITATION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES; SO THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.

4. U.S. Government Restricted Rights

The Software is licensed subject to RESTRICTED RIGHTS. Use, duplication or disclosure by the Government or any person or entity acting on its behalf is subject to restrictions as set forth in subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software Clause at DFARS (48 CFR 252.227-7013) for DoD contracts, in paragraphs (c)(1) and (2) of the Commercial Computer Software-Restricted Rights clause in the FAR (48 CFR 52.227-19) for civilian agencies, or in the case of NASA, in Clause 18-52.227-86(d) of the NASA Supplement to the FAR, or in other comparable agency clauses. The contractor/manufacturer is the Ziff-Davis Benchmark Operation, 1001 Aviation Parkway, Suite 400, Morrisville, North Carolina 27560.

5. General Provisions

Nothing in this Agreement constitutes a waiver of Ziff-Davis rights under U.S. copyright laws or any other Federal, state, local or foreign law. You are responsible for installation, management, and operation of the Software. This Agreement shall be construed, interpreted and governed under New York law. If any provision of this Agreement shall be held by a court of competent jurisdiction to be illegal, invalid or unenforceable, the remaining provisions shall remain in full force and effect.

Trademarks

Overview

The goal behind this manual is to give you a relatively short document that helps you run ServerBench's standard test suites, look at your results, and even create a quick test suite of your own without drowning you in hundreds of pages of information.

All the information in this manual is an abbreviated subset of the information in ServerBench's main manual, *Understanding and Using ServerBench® 3.0*. If you need more information about any of the steps in this manual or just want to know more about how ServerBench works, look in the main ServerBench manual.

Quick reminder:

ServerBench is a Ziff-Davis benchmark program that measures the performance of application servers in a client/server environment. It provides you with an overall score for your server and individual scores for the clients, which are PCs running Windows® 95 or Windows for Workgroups. You start and monitor the test runs from the controller, which is a PC running Windows 95 or Windows for Workgroups.

Related Topics:

Experienced ServerBench users, read this

New ServerBench users, read this

Experienced ServerBench users, read this

Because you're already familiar with ServerBench, you only need to read the "What's New in ServerBench 3.0" chapter. This chapter summarizes ServerBench's new features and tells you how to use them.

The rest of this manual is just a quick refresher course on the ServerBench basics that you already know. You can use the manual if you want to quickly double-check a task without having to pull out the more-detailed the *Understanding and Using ServerBench® 3.0* manual.

New ServerBench users, read this

This manual is designed to help you get started with ServerBench. It quickly steps you through some of the basic tasks, such as running the standard test suites. It even gives you an overview of how to create a test suite. However, it doesn't explain these steps in detail. For more information, you'll need to look in the *Understanding and Using ServerBench® 3.0* manual.

To know what's in this manual, check out the Table of Contents. Because this manual emphasizes the tasks you'll perform in the most likely order you'll perform them, the Table of Contents is the best reading path to the manual we can provide.

Overview

We've added a number of new features to ServerBench 3.0. Here's a quick look of some of the major new features that you may find useful. For a complete list of features as well as detailed explanations of them, see the *Understanding and Using ServerBench® 3.0* manual.

New RISC and OS/2 Warp Server platforms

ServerBench 3.0 has added three new Windows NT Server 3.51 RISC platforms and an OS/2 Warp Server platform. The new RISC ports support Digital Alpha, MIPS, and PowerPC microprocessors. In addition, ServerBench continues to run on Windows NT servers using x86-compatible microprocessors.

NOTE: ServerBench 3.0 does not support NetWare 3.1x or SCO Unix 4.0.

For the OS/2 Warp Server platform, ServerBench has switched network protocols. This port uses TCP/IP as the network protocol. ServerBench 2.0 on OS/2 required you to have NetBIOS on the server, controller, and clients.

Tip:

*For a complete list of operating systems that ServerBench supports, see the main ServerBench **README.TXT** file. This file is located in the **\SB30** directory on the Ziff-Davis Server Benchmarks CD-ROM.*

The controller and clients can run on Windows 95

With ServerBench 3.0, both the clients and controller will work with Windows 95 as well as Windows for Workgroups 3.11. When we added support for Windows 95 clients and controller, we kept the ServerBench client and controller programs as 16-bit Windows applications. Because we didn't upgrade them to 32-bit applications, they'll continue to work on PCs running Windows for Workgroups 3.11 and Windows 3.1.

Support for Winsock-compliant TCP/IP stacks

Another key change is that ServerBench 3.0 accepts any third-party TCP/IP stack that is Winsock 1.1 compliant as the network protocol for the controller and clients on each port except NetWare®. As a result of this change, more users will be able to use ServerBench. Previously, ServerBench had tied its controller and clients to a specific version of a TCP/IP stack (or, in the case of OS/2, to NetBIOS).

For more information on the network protocol your server platform of ServerBench requires, see the installation manual for your ServerBench port.

Unattended mode adds new error-handling features

ServerBench's new error handling features let you specify in advance what you want ServerBench to do if it encounters an error. These features work only when you run ServerBench in unattended mode (the default mode for running ServerBench). As a result of these new features, you can tell ServerBench to keep running test suites when an error occurs. Earlier versions of ServerBench stopped when an error occurred.

NOTE: If youve checked the option **Log Errors** option in the Unattended Mode dialog box, then, should an error while a mix is running, ServerBench displays the error message to the controller window, writes the message to the error log file, and displays the name of the error log file in the pop-up **Error Log(s) Detected dialog box**. This dialog box contains the name of each error file ServerBench creates during this test run. Each error file contains all the errors for one test suite. If several mixes in the test suite reported errors, then this file will contain multiple error messages.

You can view these error log files directly from the dialog box. If there's more than one file in the dialog box, select the file you want to look at first and then choose View. ServerBench displays the file using the text editor listed at the bottom of this dialog box. To make sure you dont miss this error information, ServerBench will continue to display this dialog box until you close it.

While the dialog box will stay open until you close it, once you do close it, this box won't appear again until another error occurs. At that point, the box will only list the new error file; it won't display the files you've already had a chance to view. This dialog box will also dynamically add/remove files that have been deleted (for example, if you delete the log files at the View Results dialog box or you specify a test suite with an identical name set).

When all the test suites complete and the Do you want to select a test suite? dialog pops up, the Error Log dialog box will begin flashing its title bar. This is a precaution to make sure you don't accidentally overlook this dialog box. You'll want to deal with the errors before starting another test run.

Tip:

When unattended mode is disabled, ServerBench stops if it encounters an error. To disable unattended mode, choose the Unattended Mode option from the Advanced menu in the controller window. At the Unattended Mode dialog box, choose Disabled. You can re-enable unattended mode by choosing Enabled.

You set the error-handing options you want from the Unattended Mode dialog box, which you access from the drop-down Advanced menu in the controller window. Among unattended mode's new error-handling features are:

- **Skipping** any mix that reports an error. Or you can tell ServerBench to skip the rest of the current test suite or simply to stop the test run by telling it to skip all the remaining test suites. (ServerBench's default is to skip the mix.)

Reminder:

ServerBench saves its results each time it finishes a mix. It doesn't save the results for any mix that has an error. So, even if ServerBench has an error in one mix of a test suite, the results for all the other mixes in the test suite will be fine. For example, suppose ServerBench encounters an error at mix 8 in a test suite that has 16 mixes. It skips that mix but it finishes all the other mixes. You'll have valid results for mixes 1 through 7, and 9 through 16.

- **Restarting** a mix that reports an error. You can tell ServerBench how many times you want it to try to restart the mix before moving on.
- **Writing** any errors to an error log file that you can view with any text editor. To make sure you don't miss the fact that an error occurred, ServerBench displays an Error Log dialog box that lists any error files it created for the different test suites during the test run. This dialog box stays open on the controller window until you close it by clicking on OK.

NOTE: ServerBench only writes error messages to a file when it runs in unattended mode.

- **Having** ServerBench execute a program, such as dialing up a pager, from the controller when it encounters an error. (See the next section "External notification" for more information.)

Related Topics:

External Notification

External Notification

ServerBench's external notification feature means that you can configure the controller to run a program, such as a pager program, if an error occurs. Basically, you enter a command line in the External Notification dialog box. Then, if an error occurs, the controller executes that command line. This command line can start any program, such as an executable, a **.bat** file, a **.pif** file, or an **.exe** file program. If you specify a pager program that dials a specified pager number and that program accepts alphanumeric pagers, you can actually send the error message to the pager.

The How-To Summary

Setting up the controller to notify you of errors:

Here're the basic steps for configuring the controller so it runs the program you specify when an error occurs:

1. Choose the Advanced menu in the controller window.

2. Choose the External Notification option.

3. At the External Notification dialog box, choose Enabled.

4. Enter the command line you want ServerBench to execute.

   If you include **%s** argument in this command line, the controller will replace that argument with the error message. For example, if you're executing a pager program that supports alphanumeric pagers, you can get it to display the error message when it pages you by entering a command line similar to:

```
c:\pager.exe tester %s
```

   This command line tells the controller to execute the pager program and display the error message.

4. Choose the run style you want the controller to use when it executes the program you specified. You can select any of the standard Windows run styles: Normal, Minimized, Maximized, or Hidden.

5 To see what will happen when the controller executes the command line, choose the Test option. This option lets you try the command line before starting the test suite.

6 Specify the maximum number times you want the controller to execute that program. If you enter 0 (i.e., unlimited), the controller will execute the command line you entered each time an error occurs. However, if you enter a number greater than 0, then, once the error count exceeds the notification count, the controller will no longer execute the command line.

7 Tell the controller whether to reset the error notification count each time ServerBench starts a new test suite. If you choose this option,

ServerBench resets the notification count to its original value each time it begins a test suite.

You can replace a client that terminates

If you accidentally reboot a client during a "Connect Clients" stage and cause the ServerBench client program to terminate, you can continue without that client, replace that client with another client that has the same ID number, or reconnect that client. The controller will accurately reflect the number of connected clients when the test continues.

NOTE: The controller only updates the client information when a test is running. This means that, if you've just connected five clients, but you accidentally rebooted four of them, the controller will say you have five clients connected. However, once you press OK to indicate you've finished connecting clients and then selected a test suite to run, the controller will correct its display to show that you only have one client connected.

In addition, ServerBench gives you opportunities to reconnect clients that terminate. For example, if a client terminates when you're running ServerBench with unattended mode disabled and the Pause option set to "Run without Pausing," the controller will automatically give you one chance to re-connect the client when the mix ends. If you don't reconnect the client, the test will proceed normally until it finishes.

Enhanced results reporting features

With ServerBench 3.0 comes enhanced results reporting. Two of the biggest changes are that ServerBench now uses a **.TLG** extension to identify its results files and that ServerBench now places each set of results tables in an Excel workbook.

Tip:

For a quick look at the steps you need to perform to view your results, go to Chapter 3.

Here's a summary of some of the results enhancements in ServerBench 3.0:

You can display results for 60 or more clients at once. ServerBench 2.0, because of limitations with the Excel macros, limited its results tables to a maximum of 30 clients at a time. With ServerBench 3.0, the only limits on the maximum number of clients you can display results for at one time depends on the number of clients, the number of mixes, and how much memory there is on the controller. The bottom line is that you can generally display results for a very large number of clients.

It's easier to name results files. Once you've selected a test suite, ServerBench gives you the option of entering a name for that test suite's results file as well as an identifying comment to go with the results file. This way you can easily create several sets of results from the same test suite without having to worry about overwriting the previous set of results. By default, ServerBench uses the test suite name as the base name of the results file, gives it a **.TLG** extension, and places the file (and the other log files it uses) in the controller's **RESULTS** subdirectory. When you enter the name of the results file, you can also enter a path name to a different directory. If you enter the same results path name that an existing set of results has, ServerBench warns you and asks whether you want to enter a new name or overwrite the existing results.

You only need one session of Excel running. Previously, each time you selected a set of results to view, ServerBench would launch a new session of Excel, even if you current had a session of Excel running.

You can create a TPS graph and a variance graph. When you're setting up your results, you can easily install ServerBench's new Add-In module for Excel, **SVRBENCH.XLA**. This module lets you create a TPS summary graph and a variance summary graph.

Tip:

*While you can permanently install the **SVRBENCH.XLA** module through Excel itself, we recommend that you install this module from the View Results window. This way, it won't affect your daily Excel work. If you install the module permanently in Excel, you'll notice that your Excel start-up time is slower even when you're not generating ServerBench results.*

The TPS summary graph displays the overall ServerBench TPS scores for multiple test runs. For example, if you have results from the **SYS_60.TST** for ServerBench on Windows NT using a PowerPC processor, NetWare, SCO OpenServer, and OS/2, you can combine these four individual graphs into a single graph that shows the relative performance of these network operating systems. Or you could run **SYS_60.TST** on NetWare 4.1 SMP with one, two, three, and then four processors and create a TPS summary graph to show how having additional processors affects the same test suite.

The variance summary graph displays the ratio of the mean to the standard deviation of individual client scores for different test runs.

Tip:

We recommend that, if you create a TPS graph or a variance graph, you use results from different runs of the same test suite.

The **SVRBENCH.XLA** module also enables you to easily print the graphs or tables from within the ServerBench workbooks without printing other forms of data.

The How-To Summary

Creating TPS and variance graphs :

If you choose "Load svrbench.xla Add-In" in the Select Results window, you can generate a TPS summary graph and a variance summary graph. To use this feature:

1. Once the results workbooks you want are open in Excel, choose the Data menu.

2. Choose the option ServerBench.

3. Choose either the option Create TPS Summary Graph or Create Variance Summary graph.

4. When the Select a Source Workbook dialog box appears, use the pull-down list to choose one of the current workbooks. (If the active workbook is valid, ServerBench by default marks it as selected; however, you can easily choose a different workbook.)

5. Enter a label in the comment box to describe this set of results and click on OK.

6. When the Select a Destination Workbook dialog box appears, choose New and click on OK. (You could place the graph in your current workbook, but we recommend creating a new workbook.)

7. To add other results to the graph, repeat steps 1 through 5 and choose Existing Workbook when the Select a Destination Workbook dialog box appears and highlight the name you've given this workbook. Then click on OK.

It's easier to edit the database snapshot files. The database snapshot files contain information about how your server and your clients are set up. ServerBench uses these files when it creates the server and client disclosure tables. Now you can directly edit a database snapshot file and change its contents. You'll find this option in the View Results dialog box. In addition, ServerBench displays the name of the current snapshot file in the Server and Client Disclosure dialog boxes to help you track which snapshot file you're using.

We've improved the results tables. We've streamlined spreadsheet layouts through Excel Workbooks to make ServerBench's results tables easy to understand and quick to manipulate. In addition, we've reorganized the information to make it easier for you to understand what the results mean and how ServerBench calculated them.

The standard test suites are now even more "stressful"

We've enhanced ServerBench's standard test suites to make them stress the server more.

System test suite. As a result of our research, we've tuned the system test suite to increase the amount of time the network operating system spends in kernel mode. The system test still includes processor, disk, and network tests structured into nine distinct transactions. This change does not affect the amount of time it takes to run ServerBench's system test suites with 60 clients; they still take about four to four and a half hours.

Processor test suite. We've improved the Processor Test so that it has more control over the CPU load it places on the server. When you create or modify a Processor test in the Mix Definition window, you enter a value in the range of 1 to 2800 iterations for the Total Size parameter. (The value of the Total Size parameter for all the other tests is 1 to 16777215 bytes.)

Disk test suite. We've changed the Disk test suite to modify the sizes of the data files so that none of these files will dominate the buffer cache during the execution of a mix. In addition, the server now aligns its read and write operations on boundaries that are same size as the Request Size parameter. For example, if the Request Size parameter is 2048, then the server will align the read and write operations on 2048-byte boundaries.

And, because you can display results tables containing 60 or more clients, we now supply test suites that go from 1 to 60 clients instead of the paired test suites we supplied with earlier versions of ServerBench. For example, instead of having **SYS_28.TST**, which used a maximum of 28 clients, and **SYS_60.TST** which started at 32 clients and ended with 60 clients, we now have **SYS_60.TST**, which starts at 1 client and ends with 60 clients. This means a single test suite does the same work of two test suites in previous versions of ServerBench. So the standard test suites that ship with ServerBench 3.0 are **SYS_60.TST**, **D_60.TST**, **N_60.TST**, and **P_60.TST**.

New features for creating mixes

To make it easier for you to create or edit a mix, we've added a number of improvements. One of the key improvements is that you can work with multiple mixes and test suites at one time. ServerBench lets you grab multiple test suites when you're selecting the test suites you want to use. Previously, you could only select one test suite at a time.

In addition, you can change the values in the Mix Definition window for one mix and have the changes apply to all the mixes in the test suite.

NOTE: Because you can now scroll through and edit all the mixes in a test suite by using the Mix Definition window's **Previous Mix** and **Next Mix** button, we've added flexibility to the **Cancel** function. Here's how it works now: If you've edited a mix and then press Cancel, ServerBench will display a dialog box with only one option. This option asks you if you wish to cancel the changes you've just made to that mix.

However, if you've edited a mix, then scrolled to another mix and edited it, and now choose Cancel, ServerBench will display a dialog box with two options. Option one asks whether you want to cancel the changes to the mix you just edited. Option two asks if you want to cancel all the changes to the mixes to the test suite.

Another major change from ServerBench 2.0 is that you can insert mixes anywhere within a test suite. You can also reorder the mixes in a test suite.

The following is a summary of some of ServerBench 3.0's new features for working with mixes.

- **An improved Mixes in Suite dialog box.** This dialog box lists the test suites you've selected, lets you add more suites or remove suites, and displays information about the mixes each suite contains. You can also perform several functions from this window.

NOTE: To get to the Mixes in Suite dialog box, choose Create or Edit Test suites at the ServerBench main window and then select a test suite in the Create or Select Test Suites dialog box.

- **The ability to insert mixes anywhere in a test suite.** From the Mixes in Suite dialog box, you can insert a new mix anywhere in the existing test suite (except as the first mix).

The How-To Summary

Inserting a new mix into a test suite:

To insert a new mix in a specific location in a test suite:

1. Go to the Mixes in Suite dialog box and highlight the name of the test suite in the Suite List. ServerBench displays all the mixes currently in this test suite in the Mix list. (Remember, choose the Create or Edit Test Suites button at the main ServerBench window and then select a test suite at the Create or Select Test Suite dialog box in order to get to the Mixes in Suite dialog box.)

2 Highlight the name of the mix that you want the new mix to follow.

3. Click on New. ServerBench displays the Mix Definition window. It places the mix you're creating in the test suite after the highlighted mix.

- **The ability to reorder mixes within a test suite.** You can reorder the existing mixes in a test suite as long as all the mixes in the suite have a unique name. The only caveat is that all the mixes must have a unique name.

The How-To Summary

Reordering mixes:

To reorder the existing mixes in a test suite:

1. Go to the Mixes in Suite dialog box and highlight the name of the test suite in the Suite List. ServerBench displays all the mixes currently in this test suite in the Mixes in Test Suite list. (Remember, choose the Create or Edit Test Suites button at the main ServerBench window and then select a test suite at the Create or Select Test Suite dialog box in order to get to the Mixes in Suite dialog box.)

2. Click on Re-Order. ServerBench displays the Reorder Suite Mixes dialog box. This box lists the name of the suite in the title bar and displays the mixes in their current order in the Previous Mixes list.

3. In the Previous Mixes list, click on the mix that you want to be first and then choose Add. If you select multiple mixes, ServerBench will move those mixes to the New Order list based on their current order. For example, suppose the Previous Mixes list contains Mix1, Mix2, Mix3, Mix4, Mix5, and Mix6. You highlight Mix3, Mix5, and Mix6 and click on Add. ServerBench will place those mixes in that order in the New Order list.

   NOTE: As with the New button, ServerBench inserts the mixes you're adding to the New Order list after the currently highlighted mix instead of simply appending the mixes to the list.

4. Continue moving the mixes to the New Order list until you've moved all the mixes you want and they're in the order you want. Then choose Done.

When you're at the Mix Definition window, you'll notice that this window has a new look and several new features. One of the biggest changes to this window is that it is now a viewer that lets you move back and forth between all the mixes in that test suite. Use the Previous Mix and Next Mix buttons to scroll through the mixes in this test suite. Because of this feature, we've also modified the Cancel function. If you've edited a mix, then scrolled to another mix and edited it and choose Cancel, ServerBench will display a dialog box that will asks whether you want to cancel the changes to the mix just edited or cancel all the changes to this test suite.

Tip:

You can get to the Mix Definition window from the Mixes in Suite window. To bring up the Mix Definition window, you can choose the Edit button, double-click on a *test suite name in the Suite List, or double-click on a mix name in the Mixes in Test Suite list.*

Some of the new features you'll see at the Mix Definition window include:

- **Increased precision for the Disk Test parameters**. You can enter values for the Disk test file initial size and Disk test file I/O Range parameters that go to three decimal points 0.001 MB up to 1024.000 MB.
- **Changes to the Total Size parameter**. The value the Total Size parameter changes based on whether you're setting up a Processor test or one of the other tests. For the Processor test, the value of the Total Size parameter is in iterations (1 to 2800); for all the other tests this value is in bytes (1 to 16777).
- **Entering client path names**. You can set up sequentially numbered path names for clients simply by entering a path name that ends in an asterisk in the Data File pathnames box. For example, if you've specified 60 clients and you enter the path name **f:\data***, the path names for those 60 clients become **f:\data1**, **f:\data2**, and so on up to **f:\data60**.
- **Inserting and deleting mixes**. Just as you can insert a mix into a specific place in the test suite from the Test Suite window, you can also insert a new mix into a specific place in the test suite from the Mix Definition window. When you have a mix displayed in the Mix Definition window, choose Insert new mix option in the drop-down Mix menu. ServerBench will let you create a mix that will follow the mix you were just working on. In addition, you can delete the mix you're currently editing. Choose the Delete current mix option from the Mix menu.
- **Copying fields to all the mixes in the test suite** . You can copy the fields in the mix you are working on to all other mixes in the test suite.

The How-To Summary

Copying mix parameters to other mixes:

To copy a mix parameter to other mixes in the test suite:

1. Go to the Mix Definition window and enter the value or values that you want to copy to the other mixes.

2. From the Advanced menu, choose the option Copy mix fields across this suite.

3. When the Duplicate Mix Fields Across Suite dialog box appears, select the parameters you want to copy from the current mix to all the mixes in the test suite.

   A word of caution: when you choose the option All Information under Client Information, ServerBench will change the client information in all the mixes of that test suite to make it match the client information in the current mix. This means all the mixes will have the same number of clients with the same path names and the same group numbers.

   For example, if you're editing Mix 2, which has 32 clients, and you choose All Information, then Mix1, which had 16 clients, will now have 32 clients and Mix3, which had 60 clients, will now also have only 32 clients.

However, if you choose only the individual options of either Pathnames or Groups, you won't affect the number of clients in the mix. Mix1 and Mix3 will each still have 16 and 60 clients respectively, but the path names for these clients will match the path names in Mix2, up to the maximum number of clients for Mix2. Because Mix2 has 32 clients, all of the 16 clients in Mix1 will get new path names, but only clients 1 through 32 in Mix3 will change their path names. Clients 33 through 60 will keep their current path names.

NOTE: If you change one field without changing other dependent fields (such as Length, Ramp up and Ramp down), you can create an error condition in one of the other mixes in the test suite. If that happens, ServerBench will display an error message. For example, suppose that Mix1 uses Ramp Up and Ramp Down values of 1 second each and Length value of 20 seconds. Mix2 uses Ramp Up and Ramp Down values of 50 seconds each and a Length value of 200 seconds. If you tell ServerBench to change the Length value for the mixes to match the value for Mix1, you'll create an error condition for Mix2, because the Length of the test will no longer exceed the combined values of Ramp Up and Ramp Down.

4. Click on OK. ServerBench will change the values for the specified parameters in all the mixes in the test suite.

Other new ServerBench 3.0 features

This is a quick list of some ServerBench 3.0's other enhancements.

- **Controller window icon shows test run status**. ServerBench displays the status of the current test run even when you minimize the controller window. To disable this feature, click on the ServerBench icon in the title bar of the controller window. When the System menu appears, click on the Animate Icon option to de-select it.
- **The controller sounds different tones when it displays certain dialog boxes**. The controller uses audible indicators when it displays various questions or special dialogs. This way you'll know without having to constantly watch the controller when certain events occur. For example, the controller sounds a tone when ServerBench finishes executing the selected test suites.
- **ServerBench displays client's real-time status.** If you click on a client square, the data in the Client Information pop-up box reflects the client's current status. The Status field continuously updates itself. For example, if that client is changing from the Initializing to Running test stage, you'll see the Status field change.
- **You can choose a smaller size for the client grid.** You can specify a client grid size of 77 squares. This grid size is more appropriate when you're running the standard test suites, which use a maximum of 60 clients.
- **Warning box appears if you choose Quit while the controller's in performing a crucial task**. If you try to quit the controller when it's processing critical data, ServerBench displays a warning message. For example, if you've told ServerBench to skip a mix, the controller will wait for the

server and clients to clean up. If you press Quit now, the controller won't be able to tell the clients and server to abort. So ServerBench displays a pop-up warning box asking if you really want to quit the controller now.

- **New Test Suite History window lets you track test runs.** You can use this window to review which test suites ServerBench has run and is currently running, the name of the results file for each test suite, the path names for the results files, any special comments you entered, and the status of the test suite, such as whether ServerBench skipped any mixes in a particular suite or if any errors occurred. If you request the status of a running test suite, ServerBench displays the status in real time; i.e., the current status of that test suite at that moment.

  You can display the Test Suite History window by choosing the View suite history option in the drop-down Sessions menu in the controller window. (If you haven't selected any tests during this ServerBench session, this option will be grayed out.)

- **ServerBench provides a Sticky Suites feature and a Sticky Results feature.** These two features mean that ServerBench 3.0 takes you to the directory where you last selected test suites to run or results to view. If you keep all your test suites in one directory and all your results files in another, you'll won't have to spend time searching to get to these files.

End of "What's New"

Overview

This chapter tells you what you need to do to run ServerBench, how to set up your testbed for ServerBench, and how to install the ServerBench program.

This chapter tells you how to select and run ServerBench's standard system test suite, **SYS_60.TST**. It also provides some strategies for getting good scores when you run these test suites.

Tip:

Remember that, to get the best test results, you must run ServerBench on an isolated test network.

What's different about using ServerBench 3.0

Here're the key differences in the way you run and use ServerBench 3.0 as opposed to how you ran earlier versions of ServerBench:

- You can grab multiple files when you're selecting test suites to run and when you're selecting results files to view.
- You can configure ServerBench to run in unattended mode (the default mode) so that it handles any errors that crop up, including notifying you via a program such as a pager program if an error occurs.
- You can set the client grid in the controller window to 77. Also, if you choose a client square, ServerBench displays that client's status in real time.
- Controller error messages now have numbers linked to them to make it easier to locate information on them.

Starting ServerBench:

The following steps tell you how to run ServerBench's standard system test suite **SYS_60.TST**. To successfully run this test suite, you need 60 clients.

Reminder:

You must read and agree to the License Agreement before you can use ServerBench. This agreement appears at the beginning of this manual and on the controller screen the first time you start ServerBench.

1. Start the ServerBench on the controller.

   a. Start the ServerBench controller program.

      On a controller running Windows 95, choose:

      Start -> Programs -> Ziff-Davis Benchmarks -> ServerBench 3.0 Controller

      On a controller running Windows for Workgroups, double-click on the controller icon in the Ziff-Davis Benchmarks program group.

   b. When the main ServerBench window appears, choose the Start Test button.

   c. When the controller window appears, choose the Start button.

Tip:

If you get either of the following error messages when you press Start Test at the main ServerBench window or Start in the controller window:

```
File Error: Cannot Find winsock.dll

Error 4: Cannot initiate network connection
```

you probably have an uninstalled TCP/IP stack. You'll need to install your TCP/IP stack before you can run ServerBench.

2. Start ServerBench on the server.

   a. Go to your ServerBench directory on the server.

   b. Enter the correct command line for ServerBench on your server operating system. (In most cases you can start the server executable by entering **SVR**. However, there may be other command parameters you need to enter or you may have a different starting command. To determine the correct server command line, see the ServerBench installation manual for your server operating system.)

3. When the Connect Clients message appears on the controller window, start ServerBench on the clients.

   a. Start the ServerBench client program.

On a Windows 95 client, choose:

Start -> Programs -> Ziff-Davis Benchmarks -> ServerBench 3.0 <*TCPIP or NW41*>

On a Windows for Workgroups client, choose the client icon from the Ziff-Davis Benchmarks program group.

Tip:

If you get the one of the following error messages (either in a pop-up box or on the client screen) when you start ServerBench on a client:

```
File Error: Cannot Find winsock.dll

Cannot initialize TCP/IP stack
```

you probably have an uninstalled TCP/IP stack. You'll need to install your TCP/IP stack before you can run ServerBench.

b. After ServerBench is running on each client, return to the controller. You'll notice that ServerBench has highlighted a square on the client grid for each client you connected. Click on OK in the Connect Clients dialog box.

4. Answer Yes to the dialog box in the controller window that asks if you want to select a test suite.
5. When the Choose Test Suites dialog box appears, go to the directory containing **SYS_60.TST** and highlight that test suite.
6. Choose OK.

Tip:

*While ServerBench has other standard test suites, we recommend you use **SYS_60.TST.** If you want to get an overall indication of how well your server performs. This system test suite performs a variety of tests involving your server's processor, disk, and network subsystems. If you want to determine how well a specific server subsystem is performing, you can run one of ServerBench's other standard test suites. The **D_60.TST** test suite performs only disk tests, the **P_60.TST** test suite performs only processor tests, and the **N_60.TST** test suite performs only network tests.*

7. Enter a name for the results file and enter any comments in the comments box. For example, you might name the results file **SYS_60_1** and enter the comment: *Add one processor for this test run*. (You can also enter a path name for the results file in this dialog box. If you don't supply a path name, ServerBench will place your results file in the controller's **RESULTS** subdirectory. To change the path name, use the Browse button.)
8. Choose OK. ServerBench will start running the test suite. You can monitor the test run from the controller window.

Tip:

*Running **SYS_60.TST** takes about four to five hours.*

Exiting ServerBench:

To exit ServerBench, quit any open windows, such as the controller window, and return to the main ServerBench window. Now choose the Quit button.   If you choose Quit from the main ServerBench window without closing the controller window, ServerBench will continue to run. To shut down the controller window:

1. Choose No at the Select a test suite pop-up box.
2. Choose Yes at the pop-up box that asks if you want to end the test.
3. Now choose the Quit button.

If you're running any tests when you exit, ServerBench will halt those tests. It will save the results for any mixes it has completed, but not for a mix it was in the process of running.

Here's the testing procedure we recommend

In a best-case scenario, here's how we recommend you run ServerBench:

- Perform a clean installation of ServerBench. In other words, install:

    1. Your server operating system, any drivers it needs, and only the necessary software. If your server has tunable parameters, change them as recommended in the ServerBench installation manual for your server operating system.
    2. Your controller PC's operating system (Windows 95 or MS-DOS® and Windows for Workgroups), any drivers it needs, and only the necessary software, such as Excel.
    3. On each client, the client PC's operating system (Windows 95 or MS-DOS and Windows for Workgroups), any drivers it needs, and only the necessary software. In addition, streamline each client's **AUTOEXEC.BAT** and **CONFIG.SYS** files as much as possible.

    ### Tip:

    The key here is that you'll probably see better results if you don't have unnecessary applications, NLMs, TSRs, and certainly not anything that would interfere with network traffic running on the clients and the server.

    4. Now install ServerBench on the controller, the server, and each client.
    5. Reboot your system.

- Reboot at least your server, and, if possible, your controller and clients, between each ServerBench test session. Doing this flushes the server disk cache, which enables you to start ServerBench with an empty file cache and a minimum of other software. If you reboot your server, controller, and clients before running ServerBench, you are always starting the test run from a known state.
- Set up a test network for running ServerBench. You'll get a more accurate measure of your server's performance if you run ServerBench on an isolated test network instead of a production network. On a production network, clients' requests get caught behind users' requests. As a result, it takes the server longer to respond to the clients' requests. Because ServerBench includes the time each client request spends waiting in a queue for service (as well as the time the request spends traveling back and forth across a network) when it calculates its results, you'll see lower scores for your server. Also, your users may get annoyed because they'll be getting poorer service from the server as it runs ServerBench's stress tests.

If your server fails when running the ServerBench tests, you should exit ServerBench, reboot your server, and restart ServerBench.

End of chapter

Overview

Once you've finished running a test suite, you can view its results. You can even view the results of one test suite while another test suite is running. ServerBench uses special Excel macros to display these results in spreadsheet format.

Reminder:

Don't try to view the results of a currently running test mix. Doing anything that affects a currently running mix could cause ServerBench to abort that mix.

What's different about ServerBench 3.0's results

Here's the main difference between results from ServerBench 3.0 and results from earlier versions of ServerBench:

- When you choose View Results, ServerBench automatically creates an Excel workbook containing all the results you specify. In addition to having ServerBench create an Excel workbook for each set of results, we've also reorganized and streamlined the results tables to make it easier for you to find the information you need and understand how ServerBench calculated the scores
- You can also tell ServerBench to generate a single graph plotting the TPS scores of the different sets of results as well as a graph plotting the variance in the test runs.
- If you currently have a set of results open and you choose View Results to view the new set, ServerBench no longer starts another session of Excel. It creates the new results using the same Excel session.
- You can select and open multiple results files at one time.
- The results files use an extension of **.TLG**. You no longer choose the test suite file (the **.TST** file) to view results.
- By default, ServerBench places the results file in the **RESULTS** directory, which is in the ServerBench installation directory on the controller..

Steps to view your ServerBench results:

The following list summarizes the steps you need to perform to view your test mix results.

1. Go to the main ServerBench window.
2. Choose View Results.
3. At the Select Results dialog box, choose **SYS60_1.TLG** (or whatever name you gave the results file for the test suite you ran) and click on OK.
4. At the View Results window, choose the database snapshot file, choose the options you want, and choose the Worksheets (i.e., tables) you want to see. You'll need to click on the More button to see all of your options. If this is the first time you've run a test suite, you may want to set up the database snapshot file by choosing Edit Disclosure. This option lets you enter information about how your server and clients are configured and then save the information to a snapshot file that you can use with other sets of results run on the same testbed. ServerBench uses this information to create the server and client disclosure tables.

   NOTE: If you want to automatically save the results spreadsheet, choose the option "Save Workbook with results name" (or the Save As option when you're in Excel). If you select the "Delete files" option, ServerBench will delete the results log files it uses to generate the results tables after it creates the results spreadsheets. Then, if you don't save the spreadsheet using Excel, you won't be able to view those results again. In addition, if you didn't tell ServerBench to create all the results tables, you won't be able to generate them.

5. Choose View.

For information on how to set up a disclosure database that ServerBench will use when it displays Tables 4 and 5, see the section "Using your disclosure database" later in this chapter.

The units ServerBench uses to report its scores

ServerBench reports its results as TPS, or transactions per second. Each client measures how long each transaction takes and how many transactions take place. The client then calculates its TPS score by dividing the total number of transactions by the time they took to complete. ServerBench then combines the individual client TPS scores and, using a harmonic mean, calculates the overall server score.

If you run the standard system test suite, ServerBench provides you with an overall measure of your server's performance. If you run one of ServerBench's subsystem test suites, the results you get will tell you how well that server subsystem is performing.

The higher the score, the better your application server performed.

Related Topics:

Here's what makes up a transaction

Here's what makes up a transaction

A transaction consists of the request a client sends to the server, the response it gets back, and the time it takes from the moment the client sends the request until it receives a reply from the server. To calculate the length of a transaction, ServerBench includes the time the transaction spends:

- Traveling along the network to and from the server.
- Waiting in a queue on the server to receive a service.
- Receiving the service; for example, if the transaction requires disk service, this is the amount of time the disk took to provide the service.

Additional Information:

Once the client receives a response from the server and stops its transaction timer, the client validates the response. This validation does not cause any overhead on the server and is not included the test time.

How ServerBench measures performance

ServerBench uses a weighted harmonic mean to calculate the overall score for your server. By using a harmonic mean, ServerBench can combine the scores for the different transactions to create a single representative score. ServerBench weights the different transactions based on how often the clients request that transaction in one iteration of a mix.

To determine the overall server TPS scores it produces, ServerBench:

- Tracks the amount of time each transaction takes to complete.
- Tallies the number of completed transactions. ServerBench does not count incomplete transactions or transactions that began during Ramp up or Ramp down.
- For each transaction, creates a total TPS score by adding together the TPS score for each client.
- Uses a weighted harmonic mean to calculate the total TPS scores into an overall score.

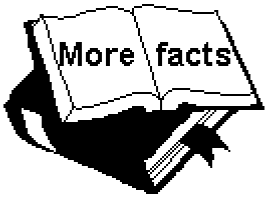
Points to check in the results tables

ServerBench has seven results tables that provide the following information:

- Table 1− ServerBench Summary. You'll find the overall ServerBench TPS score for your server in this table.
- Table 2− Overall ServerBench Data. This table contains the information ServerBench used in calculating the overall score.
- Table 3− Client Data. Information on how each client did in the test appears in this table.
- Table 4− Server Disclosure. This table contains the information about how your server is configured. You'll need this information if you want to publish your results.
- Table 5− Client Disclosure. This table contains the information about how your clients are configured. You'll need this information if you want to publish your results.
- Table 6− Test Suite Information. This table provides details on the mixes that comprise your test suite.
- Table 7− Transaction Information. This table focuses on the individual transactions. It tells you how the tests and the parameters each transaction used.

Here's a summary of some key points about the tables.

- In Table 1, look at the column containing the Harmonic Mean of Total TPS Scores to learn the score for your server. ServerBench calculates this score by applying a weighted harmonic mean to the numbers in the Total Transaction Scores (TPS) column in Table 2. (The number in parentheses in that column is the weight for that transaction. This is the same number that appears in the Transaction Iterations column in Table 7.)
- In Table 3, you can see each client's overall score calculated using a harmonic mean by looking at the Client Scores (TPS) column. You can also see each client's raw transaction score by looking in the Client Transactions Scores (TPS). The "Valid Iterations" column lists the number of iterations the mix completed during the valid portion of the test when ServerBench was recording results. The Ramp up and Ramp down columns tell you how many iterations ServerBench ignored. You'll see some variations in the number of iterations between different clients because some clients can perform more iterations in a set time than others.
- Another point to check in Table 3 is the client's deviation from the mean. This information helps you to determine whether your server is providing even service to all the clients.



Additional Information:

Check to see if your server is providing even service when it is under a heavy load. Generally, the smaller the standard deviation is, the more even the service is. As the load on your server increases, the standard deviation usually gets larger.

If you discover that clients are receiving exceptionally uneven service:

- *Make sure each network segment has the same relative load.*

- *Make sure you've set the values for Ramp up and Ramp down so that the server reaches a steady state before ServerBench begins recording test results. You can verify this by looking at the valid iterations column in Table 3 of the results. If all the valid iterations count for the clients in the mix are consistent, then the Ramp up period is long enough. (See the* Understanding and Using ServerBench® 3.0 *manual for more information about "steady state.")*

- Sometimes, especially if you're running with a small number of clients, you may get some abnormal scores. For example, one client may have a very high TPS score while the others have very low scores. This is often a sign that your server has a very large disk cache, so one client's Disk test data file may be staying in the disk cache while the others clients files are swapping out. If you notice a result like this, try increasing the size of the disk test data file and the I/O range parameter. You can also add more clients
- For Tables 4 and 5, you'll need to supply some of the information about your server and clients for your results database. ServerBench can only capture part of this information.

Concepts that play a role in ServerBench's results

Here's a summary of some of the ServerBench concepts you may want to keep in mind as you look at your results:

- ServerBench's standard test suites use stress tests.
- Because one ServerBench client generally stresses the server as much as several actual users do, you can run the test suites with a relatively small number of clients and still get an accurate measure of your server's performance.
- To get a valid measure of your server's performance, make sure you reach a knee in the ServerBench results curve. The knee indicates that the transactions per second are no longer increasing. (Table 1 of the Results spreadsheet that ServerBench produces using Excel includes a graph that plots the server's TPS scores against the number of mixes. This way you can easily spot the knee in the curve. You can use ServerBench's new TPS graph feature to create a summary graph that displays the scores from multiple results files in a single graph. This feature is handy if you want to summarize the overall results of different test runs and see where the knee occurred in each test run. To have a valid comparison, you need execute the same test each time.)

Additional Information:

Adding clients increases the total TPS ... up to a point. When the overhead of managing the additional clients outweighs the advantage of having more clients, TPS starts decreasing, causing a knee to appear in the results curve.   With ServerBench you may see one of three types of results curves:

- *A curve with a single knee that shows a very steep drop in TPS as you add clients.*
- *A curve with a single knee that shows a gradual, sloping drop in TPS as you add clients.*
- *A curve with a double knee. In this type of curve, the TPS reaches its knee and then levels out for a while, just as a plateau does, as you continue to add clients. After you've added a certain number of clients, the TPS drops off sharply. This type of curve often indicates physical bottlenecks that affect the server's performance.*

- The server with the best peak TPS number may not be the best server for your needs. You also need to consider how the server handles degradation in TPS.

What you can do with ServerBench's results

You can use these results to evaluate both the overall performance of a server as well as to check the service individual clients receive. The ServerBench results:

- Allow you to compare different servers that are tested under the same conditions and using the same testbed.

  NOTE: To get a meaningful comparison, you need to run ServerBench the same way on both systems. This means that your test parameters need to be the same and you need to run the same test suites in the same order. It also means that, if you're comparing servers that use the same operating system, you need use the same system setup and tunable parameters on the servers. In addition, your testbeds must be the same.

- Keep the server hardware constant and compare network operating systems.

- Allow you to analyze the effect of changing a single variable in a test or on your server.

Looking at ServerBench's results tables

The results spreadsheet includes five reports with a total of seven tables. Each report contains different types of data.

The next sections explain what information is in the tables for each type of report.

Related Topics:

Table 1

Table 2

Table 3

The Disclosure reports (tables 4 and 5)

The Suite Definition reports (tables 6 and 7)

Table 1

The Summary report contains the key ServerBench metric: the overall score for your server. It also includes information about the test as well as graph showing how the server performed.

The main information in Table 1: ServerBench Summary is the overall score. This is the score for your server. ServerBench calculates this score by applying a weighted harmonic mean to the numbers in the Total Transaction Score column in Table 2. The number in parentheses in the Total Transaction Score column indicates the weight that transaction had. (This is the same number that you'll see in the Transaction Iteration column in Table 7.)

ServerBench also displays information some test information. This includes comments about the test run as well as the time when the test began and ended.

Figure 2-1: Table 1

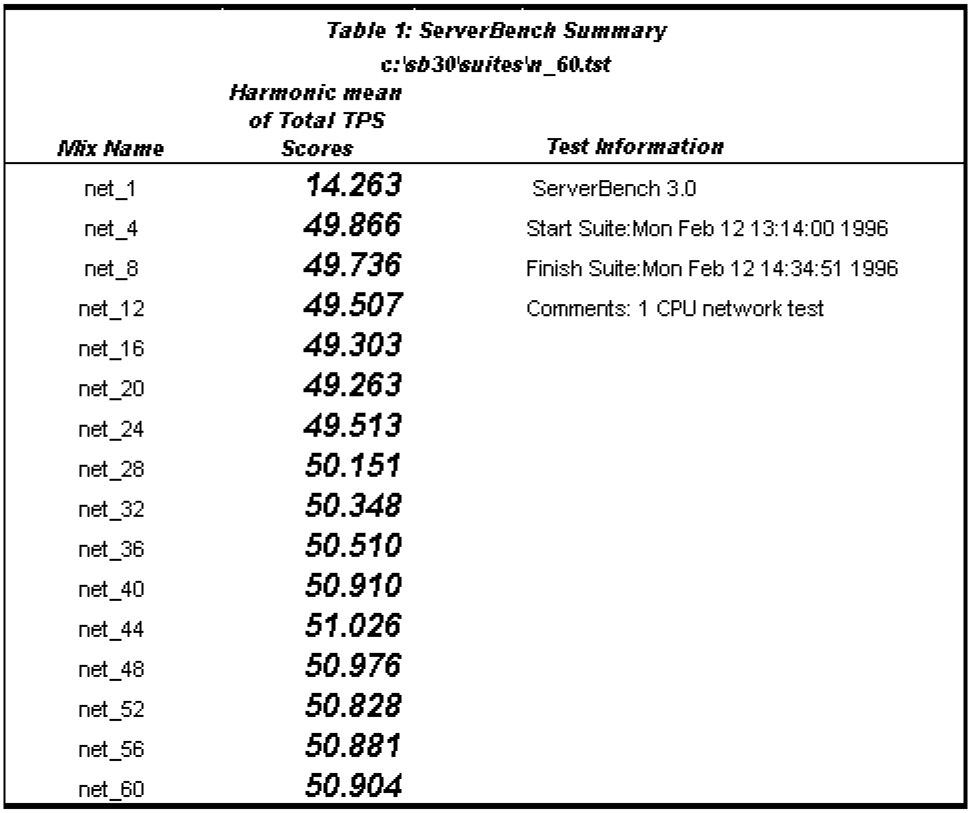


Table 1: ServerBench Summary

c:\sb30\suites\n_60.tst

Mix Name	Harmonic mean of Total TPS Scores	Test Information
net_1	14.263	ServerBench 3.0
net_4	49.866	Start Suite:Mon Feb 12 13:14:00 1996
net_8	49.736	Finish Suite:Mon Feb 12 14:34:51 1996
net_12	49.507	Comments: 1 CPU network test
net_16	49.303	
net_20	49.263	
net_24	49.513	
net_28	50.151	
net_32	50.348	
net_36	50.510	
net_40	50.910	
net_44	51.026	
net_48	50.976	
net_52	50.828	
net_56	50.881	
net_60	50.904	

ServerBench also displays a graph that plots the TPS scores against the name of each mix.

Figure 2-2: The graph associated with Table 1

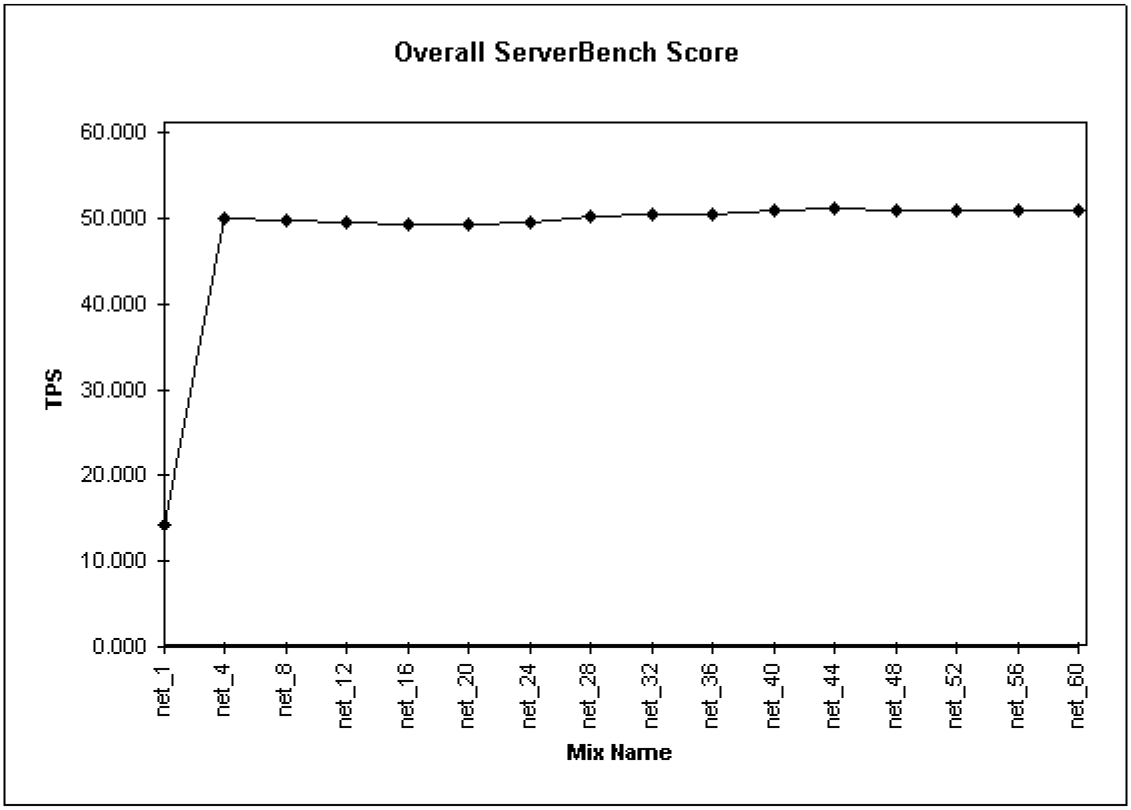

Overall ServerBench Score
TPS
60.000
50.000
40.000
30.000
20.000
10.000
0.000
net_1
net_4
net_8
net_12
net_16
net_20
net_24
net_28
net_32
net_36
net_40
net_44
net_48
net_52
net_56
net_60
Mix Name

Table 2

In the Overall ServerBench Data report, ServerBench presents Table 2. This table gives you an overview of each mix in the test suite. Because Table 2 gives you a general perspective on how ServerBench calculated your server's overall score, we include that score in this table as well.

Figure 2-3: Table 2: Overall ServerBench Data

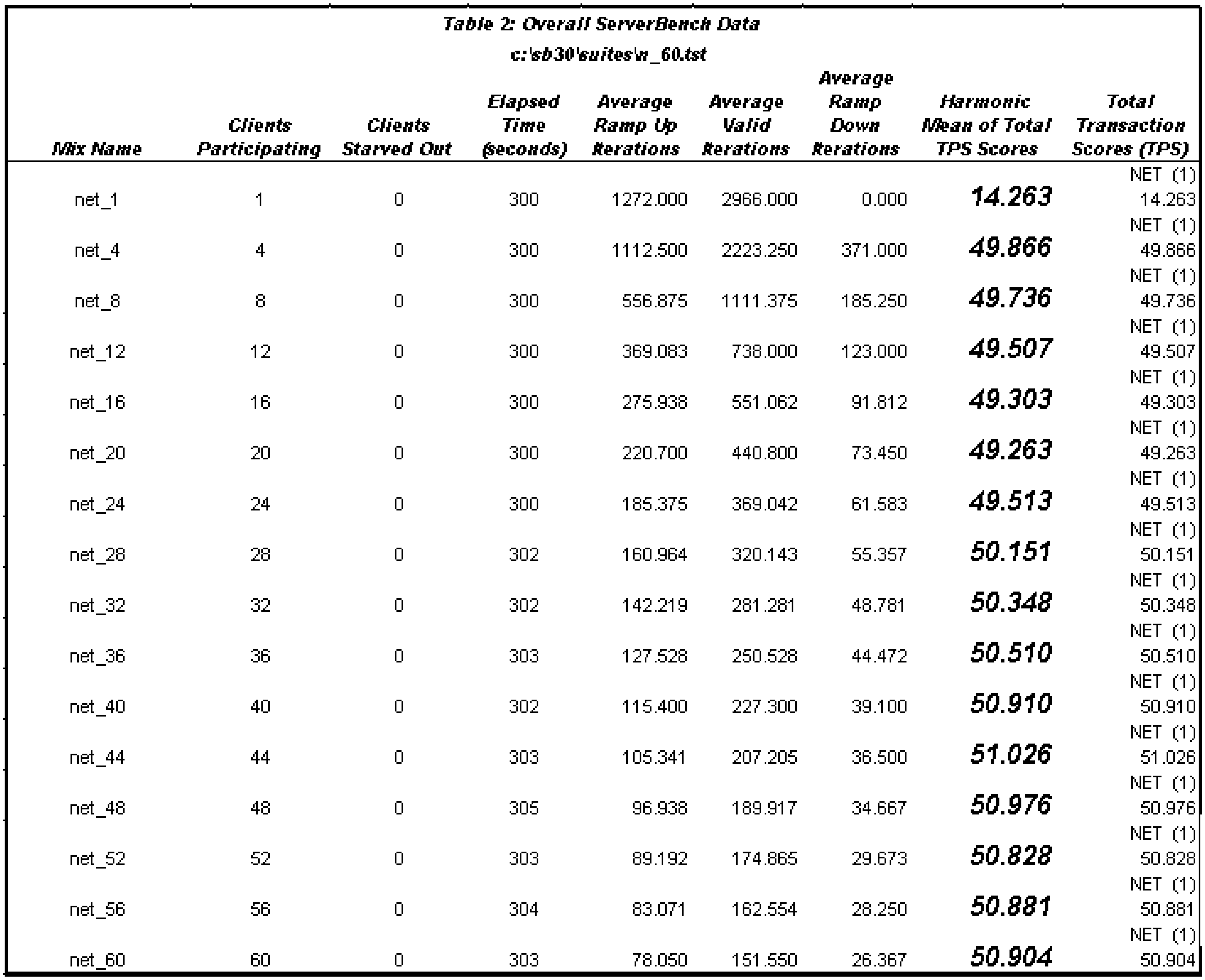


Table 2: Overall ServerBench Data

c:\sb30\suites\n_60.tst

Mix Name	Clients Participating	Clients Starved Out	Elapsed Time (seconds)	Average Ramp Up Iterations	Average Valid Iterations	Average Ramp Down Iterations	Harmonic Mean of Total TPS Scores	Total Transaction Scores (TPS)
net_1	1	0	300	1272.000	2966.000	0.000	**14.263**	NET (1) 14.263
net_4	4	0	300	1112.500	2223.250	371.000	**49.866**	NET (1) 49.866
net_8	8	0	300	556.875	1111.375	185.250	**49.736**	NET (1) 49.736
net_12	12	0	300	369.083	738.000	123.000	**49.507**	NET (1) 49.507
net_16	16	0	300	275.938	551.062	91.812	**49.303**	NET (1) 49.303
net_20	20	0	300	220.700	440.800	73.450	**49.263**	NET (1) 49.263
net_24	24	0	300	185.375	369.042	61.583	**49.513**	NET (1) 49.513
net_28	28	0	302	160.964	320.143	55.357	**50.151**	NET (1) 50.151
net_32	32	0	302	142.219	281.281	48.781	**50.348**	NET (1) 50.348
net_36	36	0	303	127.528	250.528	44.472	**50.510**	NET (1) 50.510
net_40	40	0	302	115.400	227.300	39.100	**50.910**	NET (1) 50.910
net_44	44	0	303	105.341	207.205	36.500	**51.026**	NET (1) 51.026
net_48	48	0	305	96.938	189.917	34.667	**50.976**	NET (1) 50.976
net_52	52	0	303	89.192	174.865	29.673	**50.828**	NET (1) 50.828
net_56	56	0	304	83.071	162.554	28.250	**50.881**	NET (1) 50.881
net_60	60	0	303	78.050	151.550	26.367	**50.904**	NET (1) 50.904

In this table, you'll find the following information:

- **Mix Name**. This is the name assigned to the mix when it was created.
- **Clients Participating**. This is the number of clients who completed at least one iteration of the mix during the time that ServerBench was recording test results. (ServerBench does not record results for any mix iterations that begin during the Ramp up or Ramp down periods.) Normally, this number should be the same as the number of clients included in the test mix.
- **Clients Starved Out**. This is the number of clients that were included in the mix who did not have one completed transaction.

- **Elapsed Time (seconds)**.   This is the amount of time in seconds that it took the mix to run. The controller generates this number. This number includes the time spent during the Run phase of the mix and does not include any time for initializing or reporting.
- **Average Ramp Up Iterations**. This the average number of iterations that   the clients in this mix completed during the ramp up period. To arrive at this number, ServerBench totals each client's Ramp up iterations and then divides by the number of clients participating in the mix.
- **Average Valid Iterations**. This is the average number of valid iterations for all the clients in the mix. Each client counts its valid iterations in order to calculate its TPS scores. A valid iteration is a completed mix iteration that began after the Ramp up period ended and before the Ramp down period started. To arrive at this number, ServerBench totals each client's valid iterations and then divides by the number of clients participating in the mix.
- **Average Ramp Down Iterations**. This the average number of iterations that   the clients in this mix completed during the ramp up period. To arrive at this number, ServerBench totals each client's ramp down iterations and then divides by the number of clients participating in the mix.
- **Harmonic Mean of Total TPS Score.** This is the same score you see in Table 1. It's the overall score for your server, calculated by applying a weighted harmonic mean to the Total Transaction Score in the next column. The number in parentheses in the Total Transaction Score column indicates the weight that transaction had. (This is the same number that you'll see in the Transaction Iteration column in Table 7.)
- **Total Transaction Score (TPS)**. This is the total number of transactions completed for that mix. ServerBench doesn't weight this score. The number in parentheses is the weight that transaction had. (This is the same as the number in the Transactions iteration column in Table 7.)

Table 3

Table 3 gives information about how well each client did in the mix. This table contains data about the number of iterations each client performed as well as the scores the clients achieved for each transaction. This level of granularity lets you check to see if your clients are receiving good service.

Figure 2-4: Table 3: Client Data

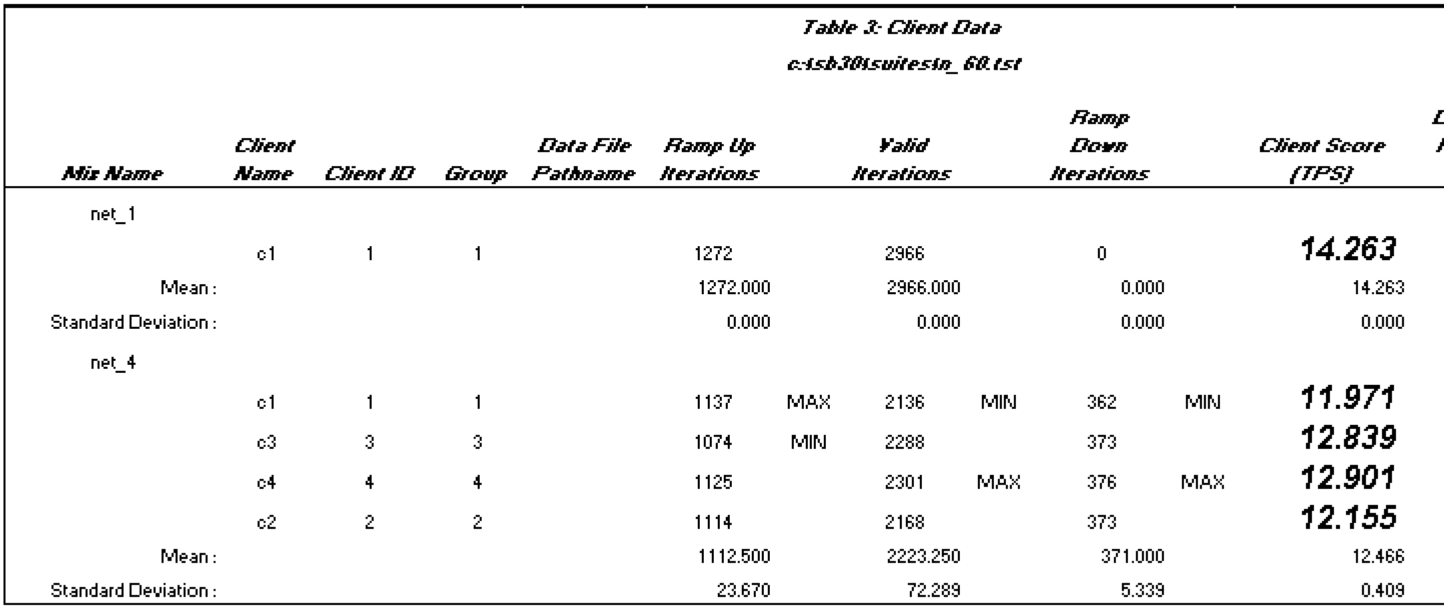


Table 3: Client Data

c:\sb30\suites\n_60.tst

Mix Name	*Client Name*	*Client ID*	*Group*	*Data File Pathname*	*Ramp Up Iterations*		*Valid Iterations*		*Ramp Down Iterations*		*Client Score (TPS)*
net_1											
	c1	1	1		1272		2966		0		**14.263**
Mean :					1272.000		2966.000		0.000		14.263
Standard Deviation :					0.000		0.000		0.000		0.000
net_4											
	c1	1	1		1137	MAX	2136	MIN	362	MIN	**11.971**
	c3	3	3		1074	MIN	2288		373		**12.839**
	c4	4	4		1125		2301	MAX	376	MAX	**12.901**
	c2	2	2		1114		2168		373		**12.155**
Mean :					1112.500		2223.250		371.000		12.466
Standard Deviation :					23.670		72.289		5.339		0.409

In Table 3, you'll find:

- **Mix Name**. This is the name you gave the mix.
- **Client name**. This is the unique name for that client. You assign the client name when you install the ServerBench client program on the client. This is the name that appears in the CLIENT.CFG file.
- **Client ID**. This is the unique ID number for that client. This is the ID number that appears in the CLIENT.CFG file. ServerBench displays the clients in the order in which it included them in the mix.
- **Group**. This is the group number for that client. You assign group numbers in the CLIENT.CFG file that you create as part of setting up ServerBench.
- **Data file pathname**. This is the path name to the data file ServerBench created for that client. By default, ServerBench creates these files in the ServerBench directory on your server. In the standard test suites, we always specify a simple path name, such as data1, data2, and so forth. If you use different names, make sure the directory you specify exists before you run a disk test.
- **Ramp Up Iterations**. This is the number of completed mix iterations that began during the Ramp up period. ServerBench also places MAX next to the client reporting the maximum number of Ramp up iterations for that mix and MIN next to the client reporting the minimum number of iterations.
- **Valid Iterations**. This is the number of completed mix iterations that each client counts toward its TPS scores. A valid iteration is one that begins after the Ramp up period ends and before the Ramp down period begins. This column is the first place you should check if you're concerned that your clients are getting uneven service. If there is a wide variation in the number of iterations completed between the different clients, then they probably aren't getting even service. One way you can smooth out service is to increase the Ramp up time. Keep in mind, though, that other factors, including the speed of the client, can influence the number of iterations a client completes.

ServerBench also places the client reporting MAX next to the maximum number of valid iterations for that mix and MIN next to the client reporting the minimum number of iterations.

- **Ramp down Iterations**. This is the number of completed mix iterations that began during the Ramp down period. ServerBench also places MAX next to the client reporting the maximum number of Ramp down iterations for that mix and MIN next to the client reporting the minimum number of iterations.
- **Client Score (TPS)**. This is the individual ServerBench score for that client. ServerBench takes the raw TPS scores for the client (shown in the column Client Transaction Scores) and uses a weighted harmonic mean to calculate the individual client ServerBench score. (ServerBench lists the weights for each transaction in Table 2. The weights are the numbers in parentheses in the Total Transaction Score column.)
- **Deviation from the Mean**. This is the distance that the client's overall score is from the mean score for that mix. You measure deviation from the mean by calculating the number of standard deviations that separate that client's score from the mean for the mix. A standard deviation of 0 to 1 is very good; 1 to 2 is OK, 2 to 3 means things are not as good as they should be, and 3 or more indicates a problem. (Mean is an arithmetic mean.) ServerBench places MAX next to the client with the largest standard deviation from the mean for that mix and MIN next to the client with the smallest deviation from the mean. (Because this is a measure of distance from the mean, it doesn't matter whether this number is positive or negative.)
- **Mean**. This is the average of the individual client ServerBench scores for the mix.
- **Standard Deviation**. This is a unit of measurement used to determine how closely the scores are clustered around the mean (i.e., to tell you how the clients scores are dispersed relative to the mean for the mix). Basically, standard deviation is a measure of what's normal. For example, if you look at a bell curve that shows a normal distribution, you'll see that most of the scores fall within one standard deviation from the mean. To calculate a standard deviation, divide the maximum distance from the mean by the mean.
- **Client Transaction Scores (TPS)**. This column contains the TPS score for that client for each transaction in the mix. These are the raw TPS scores; i.e., they're simply the number of completed transactions divided by the total amount of time spent completing the transactions. In the column Client Transaction Scores (TPS), ServerBench combines these scores using a harmonic mean to produce a individual client ServerBench score for each client.

The Disclosure reports (tables 4 and 5)

Tables 4 and 5 give you information about how your server and client test bed are set up. (This is information you'll need if you want to publish your ServerBench results.) ServerBench collects some of this information in the **.DLG** file it creates when it creates its results files. However, it can't collect all the necessary information about your server and clients. If you want the Server and Client Disclosure tables to be complete, you can either edit the **.DLG** file to include all the information or set up a disclosure database snapshot file that contains all the information. While you can only use the **.DLG** file with its associated results file, you can tell ServerBench to use the snapshot file with each set of your results. You specify which file to use for generating these tables when you're at the View Results dialog box. (See the section "Setting up your disclosure database" later in this chapter for information on how to set up a database snapshot file.)

Table 4 provides you with information about your server. We set up this table to help you gather the information the ServerBench License Agreement requires you to include if you want to publish your results. You'll also need to include the information in Table 5, which has the client disclosure information. (See the License Agreement for the details of what you must include any time you publish your ServerBench results.)

Figure 2-5: Table 4

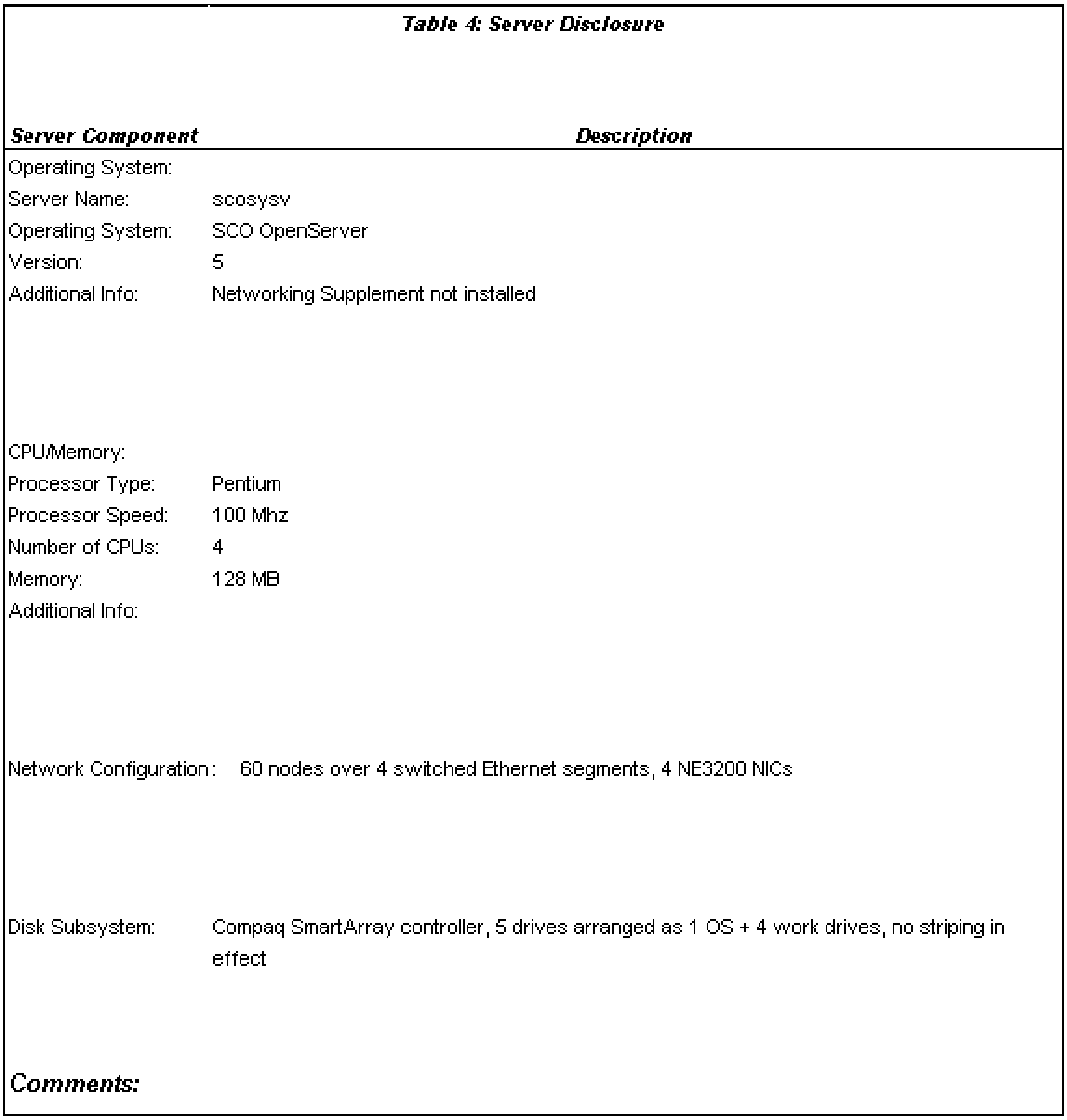


Table 4: Server Disclosure

Server Component	***Description***
Operating System:	
Server Name:	scosysv
Operating System:	SCO OpenServer
Version:	5
Additional Info:	Networking Supplement not installed
CPU/Memory:	
Processor Type:	Pentium
Processor Speed:	100 Mhz
Number of CPUs:	4
Memory:	128 MB
Additional Info:	
Network Configuration:	60 nodes over 4 switched Ethernet segments, 4 NE3200 NICs
Disk Subsystem:	Compaq SmartArray controller, 5 drives arranged as 1 OS + 4 work drives, no striping in effect

Comments:

NOTE: For illustration purposes, we've created an example of Table 4, which we show in Figure 12-5. This is not an actual disclosure sheet.

Table 4 contains the Server Disclosure information. We've broken it down into four sections to make it easier for you to follow. The first section deals with your server operating system and contains the following information:

- **The server name**. This is the name you supply for the server.
- **Operating system**. This is the operating system you use on your server; for example, NetWare, OS/2 Warp Server, SCO OpenServer, SCO UnixWare, or Windows NT
- **Version**. This is the version number of your server operating system.
- **Additional Info**. You can add any other information here that you find useful.

The second section focuses on your server's processor subsystem (i.e., CPU/Memory) and has information on:

- **Processor Type**. This is type processor your server is using, such as Pentium™.
- **Processor Speed**. This is how fast your processor is; for example, 90 MHz.
- **Number of CPUs**. This is how many processors you're using.
- **Memory**. This is how much memory your system has configured.
- **Additional Info**. You can add any other information here that you find useful, such as adding a note that you're using a multiprocessor configuration.

The third section is your Network configuration section. Use this section to identify the network cards and drivers your server is using.

The final section is your Disk Subsystem configuration section. You might enter something like you're using disk mirroring, what your server's disk drive type is, what the disk controller type is, how many drives your server's using, and what the disk drive name and version is.

At the end of the table there's a place where you can enter any general comments you want to.

Table 5 has the Client Disclosure information. This table contains information on each client in the test. You'll also need to include the information in Table 4, which has the server disclosure information if you want to publish your results. (See the License Agreement for the details of what you must include any time you publish your ServerBench results.)

Figure 2-6: Table 5

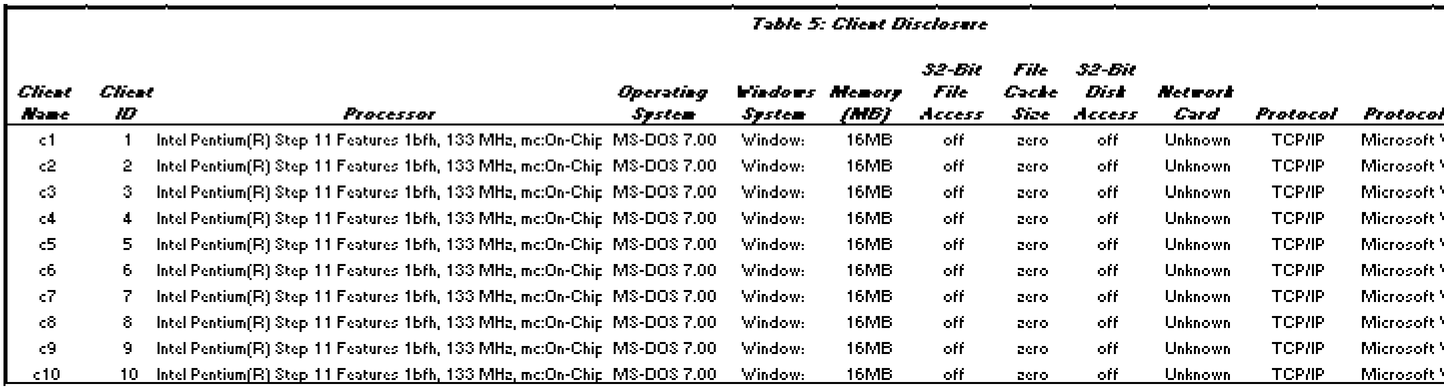


Table 5: Client Disclosure

Client Name	Client ID	Processor	Operating System	Windows System	Memory (MB)	32-Bit File Access	File Cache Size	32-Bit Disk Access	Network Card	Protocol	Protocol
c1	1	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c2	2	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c3	3	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c4	4	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c5	5	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c6	6	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c7	7	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c8	8	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c9	9	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft
c10	10	Intel Pentium(R) Step 11 Features 1bfh, 133 MHz, mc:On-Chip	MS-DOS 7.00	Window:	16MB	off	zero	off	Unknown	TCP/IP	Microsoft

Table 5 tells you:

- **Client Name**. This is the unique name you gave that client in the client configuration file (**CLIENT.CFG**).
- **Client ID**. This is the unique identification number you gave that client in the client configuration file (**CLIENT.CFG**).
- **Processor**. Here you'll find information about each client's processor. Because your clients don't all have to be same, this information can vary from client to client.
- **Operating system**. This column contains the name of the client's operating system, such as MS-DOS, and its version number.
- **Windows System**. This is the name and version number of the client's Windows system; for example, Windows 95.
- **Memory (MB)**. This is the amount of memory the client has.
- **32-Bit File Access**. You can tell whether a client is using 32-bit file access. If the client is using it, the word "on" appears in this column. If the client isn't using it, you'll see "off" here.

- **File Cache Size**. This column tells you what size the client's file cache is, if anything.
- **32-Bit Disk** Access. This column tells you whether 32-bit disk access is on or off.
- **Network Card**. This tells you what type of network card the client is using. You'll need to supply this information; ServerBench can't capture it.
- **Protocol**. This is the type of network protocol the client is using.
- **Protocol version**. This is the version number for the network protocol the client is using.

The Suite Definition reports (tables 6 and 7)

The two tables in this report provide you with details on the test suite you ran and the transactions it included. By looking at these two tables, you'll get a perspective on the work your server did in this test suite.

Table 6 gives you information about each mix that you ran. This table gives you information about each mix. These are the mix parameters you set when you're at the Mix Definition window.

This table uses a separate line to present information on each mix.

Figure 2-7: Table 6

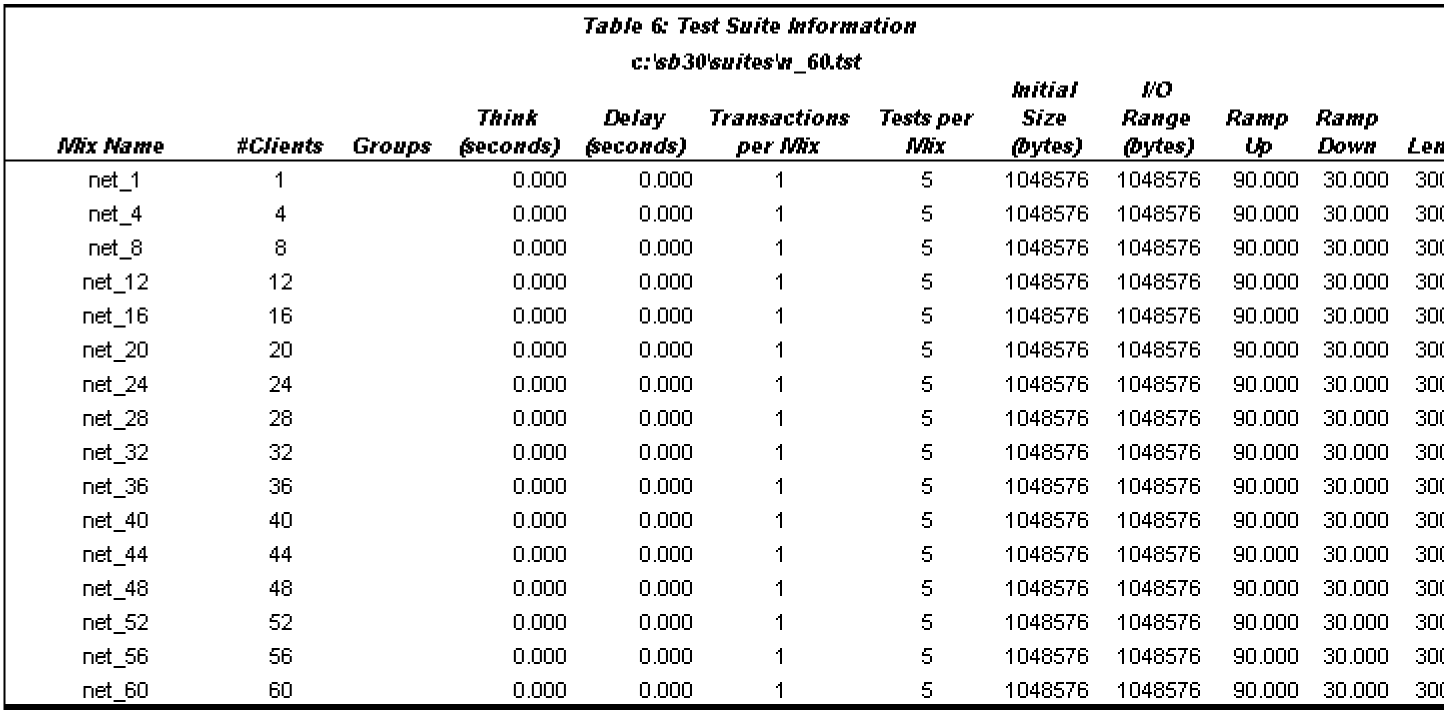


Table 6: Test Suite Information

c:\sb30\suites\n_60.tst

Mix Name	#Clients	Groups	Think (seconds)	Delay (seconds)	Transactions per Mix	Tests per Mix	Initial Size (bytes)	I/O Range (bytes)	Ramp Up	Ramp Down	Len
net_1	1		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_4	4		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_8	8		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_12	12		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_16	16		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_20	20		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_24	24		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_28	28		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_32	32		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_36	36		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_40	40		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_44	44		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_48	48		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_52	52		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_56	56		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300
net_60	60		0.000	0.000	1	5	1048576	1048576	90.000	30.000	300

In this table, you'll find the following information:

- **Mix Name**. This is the name assigned to the mix when it was created.
- #Clients. This is the maximum number of clients you want to run the mix. ServerBench will run the mix with fewer clients if necessary (for example, if you don't have that many clients connected to the server.)
- **Groups**. This is the list of group numbers selected for this mix. The ServerBench standard test suites do not select any groups. This way all the connected clients can run in the mix.
- **Think time**. This is the amount of time in seconds (or fractions of seconds) that a client waits once the server has replied to its request before it sends another request to the server. In general, the smaller the value for Think time, the more the test stresses the server. The ServerBench standard test suites use a Think time of 0 seconds. For more information on Think time, see Chapter 10 "Creating Your Own Test Suites."
- **Delay time**. This is the amount of time in seconds (or fractions of seconds) that the client waits to start the mix once the controller says all the clients can begin. By setting Delay time you can stagger the clients' initial requests to the server instead of overloading the server with a burst of requests at

the beginning of a mix. The ServerBench standard test suites use a Delay time of 0 seconds. For more information on Delay time, see Chapter 10 "Creating Your Own Test Suites."

- **Transactions per mix**. This is the total number of transactions a client requests during one iteration of the mix (i.e., it's the sum of the number of iterations for each transaction in the mix). Suppose you set the mix up to perform 6 iterations of TRANS1, 1 iteration of TRANS2, and 8 iterations of TRANS3. The number you'd see in this column would be 15 (6 + 1 + 8).
- **Tests** per **mix**. This is the total number of tests the server executes during one iteration of the mix.
- **Initial** Size **(bytes)**. This is the size in bytes of the data file used by the disk tests when the test began. If you are creating a mix, this is the test value you supply for the Disk Test File Initial Size parameter. When you run the Append test, your final size will differ from your initial size.
- **I/O** Range **(bytes)**. This is the number of contiguous bytes in the test data files that ServerBench used for input/output operations during a disk test.
- **Ramp up**. This is a fixed number of seconds at the beginning of the mix. ServerBench does not record any results from mix iterations that begin during the Ramp up period. This way your test results aren't skewed because the server load for the first few clients was very light, thus resulting in an artificially high score. (For more information on Ramp up, see Chapter 10 "Creating Your Own Test Suites.")
- **Ramp down**. This is a fixed number of seconds at the end of the mix. ServerBench does not record any results from mix iterations that begin during the Ramp down period. This way your test results aren't skewed because the server load for the last few clients was very light, thus resulting in an artificially high score. (For more information on Ramp down, see Chapter 10 "Creating Your Own Test Suites.")
- **Length**. This is the number of seconds ServerBench spends executing the mix. Because the mix will continue to run until each client finishes its current iteration, the value you see here is actually the minimum amount of time that the mix ran.(For more information on Length, see Chapter 10 "Creating Your Own Test Suites.")

Table 7 provides you with the details on the transactions in each mix. This table uses a separate line in the table for each test within each transaction.

Figure 2-8: Table 7

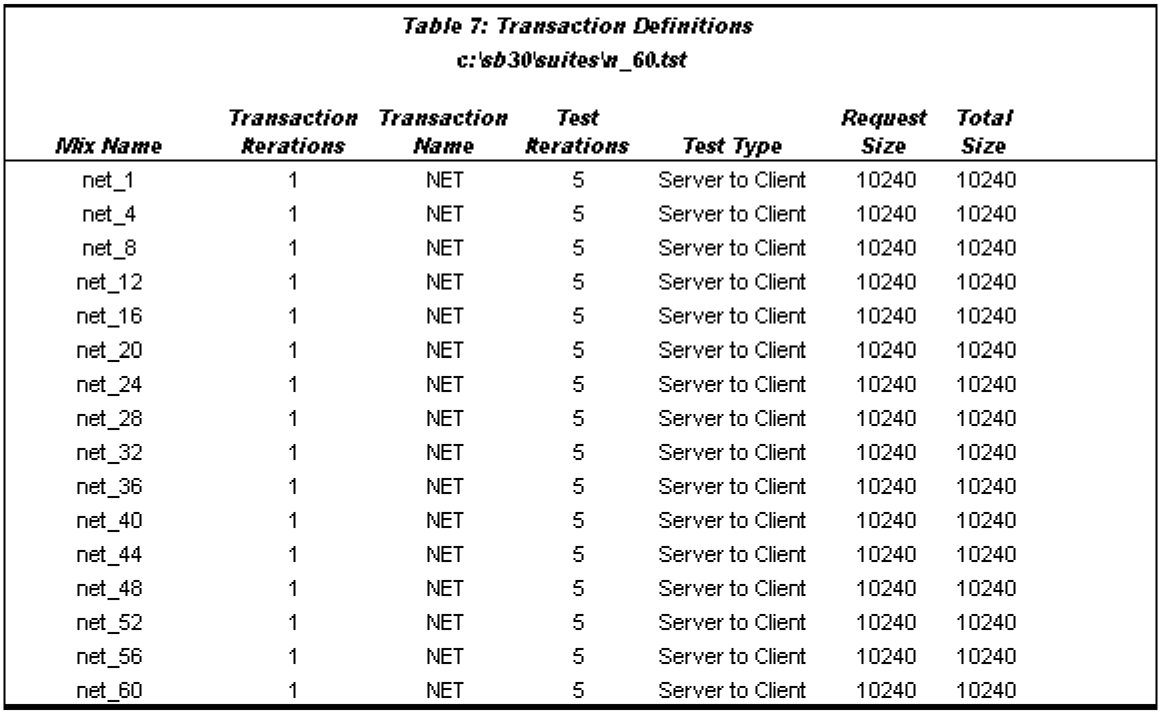


Table 7: Transaction Definitions

c:\sb30\suites\n_60.tst

Mix Name	Transaction Iterations	Transaction Name	Test Iterations	Test Type	Request Size	Total Size
net_1	1	NET	5	Server to Client	10240	10240
net_4	1	NET	5	Server to Client	10240	10240
net_8	1	NET	5	Server to Client	10240	10240
net_12	1	NET	5	Server to Client	10240	10240
net_16	1	NET	5	Server to Client	10240	10240
net_20	1	NET	5	Server to Client	10240	10240
net_24	1	NET	5	Server to Client	10240	10240
net_28	1	NET	5	Server to Client	10240	10240
net_32	1	NET	5	Server to Client	10240	10240
net_36	1	NET	5	Server to Client	10240	10240
net_40	1	NET	5	Server to Client	10240	10240
net_44	1	NET	5	Server to Client	10240	10240
net_48	1	NET	5	Server to Client	10240	10240
net_52	1	NET	5	Server to Client	10240	10240
net_56	1	NET	5	Server to Client	10240	10240
net_60	1	NET	5	Server to Client	10240	10240

In this table you'll find:

- **Mix Name**. This is the name assigned to the mix when it was created.
- **Transaction Iterations**. During one iteration of the mix, a client may request a transaction more than once. This number tells you how many times a client requested this transaction during one iteration. The more times a client requests a transaction during a single iteration, the higher the weight of that transaction. Thus, this value determines the "weight" of the transaction in the weighted harmonic mean that ServerBench uses to compute the overall score.
- **Transaction Name**. This is the name you assigned to this transaction in the Mix Definition window when you created it. We recommend that you use descriptive names when you set up transactions because a single transaction can contain more than one kind of test.
- **Test Iterations**. This is the number of times a test is repeated each time a client requests that transaction. Suppose Transaction1 contains one Sequential Read test with a test iteration of 6 and 1 Sequential Write test with a test iteration of 2. During one iteration of Transaction1, the server would perform a total of eight tests; it would execute the Sequential Read test six times and the Sequential Write test two times.
- **Test Type**. This column tells you which individual tests are in that transaction. You can create transactions using ServerBench's Processor, Sequential Read, Sequential Write, Random Read, Random Write, Append, Server to Client, and Client to Server tests.
- **Request size**. This is the amount of data ServerBench passes to your server's operating system at one time (i.e., this is the value the ServerBench application on the server uses for the size parameter when it requests file I/O or network I/O). For example, if you select a disk random read test with an Request Size of 512 bytes, ServerBench will read data from your server's disk in increments of 512 bytes. You'll see a Request Size for every test except the Processor Test. (See the section "How the Request Size parameter affects your results" in Chapter 10 "Creating Your Own Test Suites" for more information.)
- **Total size**. For a Processor test, this is the total number of iterations the test performed. For the other tests, this is the total amount of data that you want ServerBench to move for that test. For example, if your Total Size for the random read disk test is 51200 bytes and your Request Size is 512 bytes, ServerBench would read data from your server's disk in increments of 512 bytes until it had read a

total of 51200 bytes of data. (Remember, Total Size is not the same as the Initial Size of the disk test data file. See the section "How the Total Size parameter affects your results" in Chapter 10 "Creating Your Own Test Suites" for more information.)

Suggestions for improving your server's results

Both hardware and software factors can influence ServerBench results. However, you may see an improvement in your server's results if you:

- Increase the software disk cache size on the file server. The size of the cache is very important. Larger sizes generally result in better TPS scores. However, you don't want to make the cache so large that swapping occurs.
- Add memory, disks, disk controllers, or NICs.
- Reduce the number of clients on each network segment by dividing your network into multiple segments
- Add intelligent disk and network controllers. These off-load the CPU and increase performance. (If you are running ServerBench with an Windows NT Advanced Server, you should use an intelligent network controller..)
- Add processors where possible. Keep in mind, though, that because of poor processor scaling and other bottlenecks, too many processors may hurt your server's performance.

Setting up your disclosure database

Unless you specify differently, ServerBench creates Tables 4 and 5 based on information it captures and stores in its database log file (**.DLG**). However, ServerBench isn't able to capture all the information about the server and clients. This is why we give you the option of setting up a disclosure database snapshot file (**.SNP**). By using a snapshot file, you can capture all the information about your test setup once. Then, each time you want to view test results, you can tell ServerBench to use that snapshot file to set up Table 4: Server Disclosure and Table 5: Client Disclosure.

NOTE: You need the information in Tables 4 and 5 if you want to publish your results.

The first time you run a test suite after installing ServerBench, you'll see a grayed-out path name in the View Results dialog box. This is the path name ServerBench proposes for the snapshot file. If you agree to the path name, ServerBench will create this file when you create it by choosing Edit Snapshot. By default, ServerBench calls the file **SVRBENCH.SNP** and places it in the installation directory on the controller.

NOTE: You can also edit this file when you're at the View Results window.

Once ServerBench creates the snapshot file, you *cannot* add more clients to it. So the first test suite you run should use the most clients you plan to use for any test suite.

NOTE: Actually, this is not a big problem, because you can easily create a new database file whenever you need to.

Related Topics:

Creating your first database snapshot file

Creating your first database snapshot file

Here's how we recommend you create your first database snapshot file:

1. Start ServerBench on the controller, server, and the maximum number of clients that you plan to use as part of your ServerBench testbed.

2. Select and run a short test suite that uses all the clients you have connected.

Tip:

We recommend you run **SAMPLE.TST** with the maximum number of clients you plan to use with ServerBench. For example, if you plan to use ServerBench's standard test suites, you'll need to have 60 clients connected to the server. **SAMPLE.TST** takes about two minutes to run and sets up a database for 60 clients.

3. Go to the main ServerBench window and choose View Reports.

4. At the Select Results dialog box, choose the results for the test suite you just ran and click on OK. ServerBench displays the View Results dialog box. Because the snapshot file doesn't exist, ServerBench has grayed out the Use snapshot option and the path name.

   NOTE: If you don't see a grayed out path name, change your system color scheme to the Windows default scheme. Sometimes text may appear to be missing due to the colors you're using for your window text or application background.

5. Choose Edit Snapshot. ServerBench displays the Server Disclosure form.

6. Enter the correct data for your server. Because you're editing a snapshot file, ServerBench saves the information you're entering to a file.

7. Choose the Clients... button.

8. At the Client Disclosure Form, enter the correct data for your clients (ServerBench will have already filled in some of this data for you.) Again, ServerBench automatically saves these changes to your snapshot file.

9. Click on OK in the Client Disclosure window. ServerBench displays a dialog box warning you that you're about to change your disclosure snapshot file and asking if you want to overwrite that information.

10. Choose Yes. ServerBench returns you to the Server Disclosure window.

11. Click on OK in the Server Disclosure window. ServerBench displays another dialog box warning you that you're about change your disclosure snapshot file and asking if you want to overwrite that information.

14. Choose Yes. ServerBench returns you to the View Reports dialog box. You can either view your results by choosing View in the View Reports dialog box or return to the main ServerBench

window by choosing Cancel.

NOTE: The next time you choose want to look at your results, ServerBench will display the name of the snapshot file you just edited in the Use snapshot... box in the View Results dialog box.

For more information on setting up your disclosure database and working with snapshot files, see ServerBench's reference manual, *Understanding and Using ServerBench® 3.0*.

End of chapter

If you want to publish results, read this

The ServerBench License Agreement requires that you include certain information any time you publish or distribute ServerBench results.

NOTE: The License Agreement at the front of this manual and in the **README.TXT** file states exactly what information you must include when you publish your results.

If you want to publish your ServerBench results, you will need to state what your results were and information on exactly how you had your test server and PCs set up. This means you will need to provide the information about your server, testbed, and ServerBench components.

You can get this information from the disclosure database that you set up when you view the results. (For more information on the disclosure database, see Chapter 12.)

Basically, you'll need to provide the following information if you want to publish your ServerBench results:

- Server Disclosure

  Machine name Number and type of CPUs, including clock speed Size of hardware CPU cache Amount of memory Type of I/O bus Number and type of hard disk controllers Number and type of hard disks Disk organization (striped, mirrored, RAID, etc.) Disk controller driver name and version Number and type of network controllers Network controller driver name and version Network operating system name and version Any relevant modifications to default network operating system parameters
- Client Testbed Disclosure

  Network type (10Base-T, Token Ring, etc.) Number of clients Number and type of hubs/concentrators (full duplex, switching, etc.) Number of clients/segment Client OS (Windows 95, Windows for Workgroups) Client CPU type and speed in percentages Client network controller broken down by percentages Client network software name and version (drivers, protocols) Size of any client network cache
- Controller Disclosure

  Controller network software name and version(drivers, protocol)
- ServerBench Disclosure

  ServerBench version Description of the test parameters for each mix in the test suite

Related Topics:

Sample ServerBench disclosure sheet

ServerBench terms and measurements

Sample ServerBench disclosure sheet

The following example shows the type of disclosure information you'll need to provide if you publish or distribute your ServerBench results. This information is taken from one of our test runs as we developed ServerBench 3.0.

Sample Server Disclosure

Name	Compaq ProSignia
CPU	1 Intel 486DX2/66
CPU cache	256 KB
Memory	64 MB
I/O bus	EISA
Disk controller	Compaq Integrated Drive Array
Disk drives	2 Conner CP3541 512 MB drives
Disk driver	cpqda386.dsk 12,621 5/29/92
Disk organization	hardware striping
Network controller	Compaq 32-bit NetFlex controller
Net driver	cpqethnw.lan 40,9059 /11/92
NOS	NetWare 4.1
NOS parameters	set maximum physical receive packet size = 4202

Sample Client Testbed Disclosure

Network	10Base-T
Clients	32
Hubs	8 Accton EtherHub-8mini
Clients/seg	16
CPU	75% 486/25, 25% 386/20
Network cards	75% Eagle NE2000, 25% Cabletron E20
Software	Windows 95, Microsoft TCP/IP using Windows Sockets Version 1.1
Cache	32-bit-file-access enabled, 1MB cache

Sample Controller Disclosure

Controller software	Microsoft Windows 95

Sample ServerBench Disclosure

Version	3.0
Mixes	standard test suite **SYS_60.TST**

ServerBench terms and measurements

If you publish your test results, you will want to use the names associated with ServerBench correctly. The following list contains some of the key ServerBench terms and how you should use them:

- **ServerBench 3.0**. This is the name of this benchmark. The term "ServerBench" is a registered trademark of Ziff-Davis Publishing Company.

  ServerBench is a Ziff-Davis benchmark that lets you measure the performance of an applications server in a client/server environment. It provides you with an overall score for your server. ServerBench is a portable benchmark and currently runs on several server platforms (see the ServerBench 3.0 **README.TXT** file for a complete list of which platforms ServerBench runs on).
- **Test suites**. A group of test mixes that ServerBench executes in sequence.
- **Test mixes**. A group of transactions and the parameters for those transactions.
- **TPS**. ServerBench reports all of its results in terms of TPS (transactions per second). A completed transaction is one where the client submitted a request to the server and then received a response from the server saying the transaction was complete. The client starts a timer when it issues the request and stops it when it gets a response from the server.
- **Transaction**. ServerBench defines a transaction as one or more individual tests bundled together. During a mix, each client submits each transaction as a single request across the network to the server.

End of chapter

Overview

You can create your own ServerBench test suites or modify the standard test suites that come with ServerBench (**SYS_60.TST**, **D_60.TST**, **N_60.TST**, **P_60.TST**, and **SAMPLE.TST**). In this chapter, we provide you with a list of steps for creating or modifying a test suite. We don't explain all the options you have when you work with test suites. For more details, see ServerBench's reference manual, *Understanding and Using ServerBench® 3.0*.

Converting test suites to ServerBench 3.0:

Many earlier versions of ServerBench didn't include the I/O Range parameter. So, when you run a test suite that was created using an earlier version of ServerBench, ServerBench 3.0 displays a warning message and asks if you want to convert the file to 3.0 format. Choose OK. ServerBench saves your original test suite with a **.BAK** extension. ServerBench sets the I/O range value for the converted test suite to the same size as the file size. In addition, when ServerBench converts a file, it doesn't necessarily use round MB values for the Disk Test Initial File Size or the I/O range parameter. You can change the I/O range parameter and Disk Test Initial Size parameter to the values you want by following the steps in this chapter on modifying a test suite from the Mix Definition window.

Points to remember:

When you run ServerBench's tests, you're actually executing a test suite. A test suite is a container for at least one test mix. The test mix consists of at least one transaction. A transaction consists of at least one test. You can have multiple tests in a transaction, multiple transactions in a mix, and multiple mixes in a single test suite. ServerBench executes these tests once you specify the name of the test suite you want it to run.

You always create a mix or modify the values for a mix's parameters from the Mix Definition window.

Steps for developing your own test suite

Here's how you develop a test suite:

1. Go to the main ServerBench window and choose the Create or Edit Test Suites button. The Create or Select Test Suite File dialog box appears.
2. To create a test suite, enter a name for the test suite you want to create. If the name you enter does not exist in that directory, ServerBench asks if you want it to create the file. Answer Yes.

Tip:

All test suites must have a ***.TST*** *extension.*

To modify a test suite, highlight that suite's name in the list of suites in that directory. You can choose multiple test suites.

3. When the Mixes in Suite... window appears, chose the "New" button to define a new mix. If you're modifying an existing mix, highlight the mix you want to modify and choose the "Edit" button. (For more information on these features, see the section "New Features for Creating Mixes" in What's New About ServerBench 3.0 at the beginning of this manual or the manual *Understanding and Using ServerBench 3.0*.)

   The Mix Definition window appears.

Tip:

You get to the Mix Definition window from the Mixes in Suite window. To bring up the Mix Definition window, you can choose the Edit button, double-click on a test suite name in the Suite List, or double-click on a mix name in the Mix List.

4. Enter all the information to set up this mix in the boxes in the Mix Definition window.

Mix Name

a. In the Mix Name box, enter in a name for the mix. The name can be up to 15 alphanumeric characters.

Duration Information

b. In the Ramp up box, enter the amount of time in seconds that you want ServerBench to spend on Ramp up. ServerBench's standard test suites vary these values.

Additional information:

The purpose behind the Ramp up and Ramp down periods is to ensure that ServerBench only records results while the server is in a steady state, not when it has a light load. Light loads generally occur at the beginning of the test when not all the clients have started the test (the Ramp up period) and at the end of the test when some clients have already ended the test (the Ramp down period). ServerBench does not count the results of any tests that begin during the Ramp up or Ramp down periods.

c. In the Ramp Down box, enter the amount of time in seconds that you want ServerBench to spend on Ramp down. ServerBench's standard test suites vary these values.

d. In the Length box, enter the minimum amount of time in seconds that you want the test mix to run as the value for the Length parameter. This time needs to include the time for both Ramp up and Ramp down as well as time for ServerBench to run the mixes and calculate the results. ServerBench's standard test suites vary these values.

Timing Information

e. In the Delay box, enter the maximum amount of time in seconds (or fractions of seconds) that you want a client to wait before starting a test once the controller tells the clients to start. When you set the Delay time parameter, each client takes that value and generates a pseudo-random number. This way the Delay time can vary for each client. As a result, when you set Delay time, you stagger the clients' initial requests to the server instead of overloading the server with a burst of requests at the beginning of each mix. The default value for Delay is 0, which is the value ServerBench's standard test suites use.

f. In the Think box, enter the number of seconds or fractions of a second a client will wait after receiving a transaction response before it sends another request. The default value for Think time is 0, which is the value ServerBench's standard test suites use.

Transaction Definitions

g. Choose the New button. ServerBench moves your cursor to the Transactions Iterations box.

h. Under Transaction iterations, enter the number of times you want each client to request the transaction during one iteration of a mix.

i. In the Transaction Name box, enter a name for the transaction. You can enter any name you like or you can use an existing mnemonic for a ServerBench test. If you choose one of the mnemonics, such as SR, ServerBench will assume you want the transaction to be a singleton; i.e., a single test with the parameters you specify. To see a list of the test mnemonics, choose the down arrow next to this box.

j. In the Test Iterations box, enter the number of times you want the server to execute the test within that transaction. If you are creating a singleton, leave this box blank. ServerBench will only execute the test once.

k. In the Test Type box, then enter the mnemonic for the test you want ServerBench to run in this transaction. To see a list of the test mnemonics, choose the down arrow next to this box. (If you entered a test mnemonic as your transaction type, then this box will be filled in.)

l. Enter a value in bytes for the Request Size. The disk tests and the network tests require this parameter (if you're creating a processor test, ServerBench automatically puts N/A in this box). This is the amount of data ServerBench exchanges with your server's operating system at one time (i.e., this is the value the ServerBench application on the server uses for the size parameter when it requests file I/O or network I/O). For example, if you select a disk random read test with an Request Size of 512 bytes, ServerBench will read data from your server's disk in increments of 512 bytes.

m. Enter a value for the Total Size. If you're setting up a Processor test, this value will be 1 to 2800 iterations. If you're setting up any other kind of test, this value will be 1 to 16777 bytes. For the disk tests and the network tests, this is the total amount of data that you want ServerBench to move for that test. For example, if your Total Size for the random read disk test is 51200 bytes and your Request Size is 512 bytes, ServerBench would read data from your server's disk in increments of 512 bytes until it had read a total of 51200 bytes of data.

n. Enter a value in megabytes for the Disk Test File Initial Size. This parameter tells ServerBench how big to make the data files it creates for the disk tests. During the Append test, these files can grow beyond this initial size.

o. Enter a value in megabytes for the I/O Range Size. This parameter defines the contiguous range within the data file that ServerBench will read from or write to.

p. To add another test to this transaction, press New. Leave the Transaction Iterations and Transaction Name blank and fill in information for Test Iterations, Test Type, and the necessary parameters. ServerBench will add this test to your current transaction. Continue this step until you've added as many tests as you want. When you're ready to set up another transaction, choose the New button again and enter values for Transaction Iterations and the rest of the parameters.

Client Information

q. In the Total number of clients box, enter the number of clients you want to execute this test mix.

r. If your mix includes any disk tests, you'll need to supply a value in the Data file pathnames box. Enter the path name to that client's disk test file on the server. If you enter a path name that ends in an asterisk (*), ServerBench will set up sequentially numbered path names of all the clients. For example, if you enter **DATA***, ServerBench will set up the files **DATA1**, **DATA2**, and so on until it has a directory path name to a file for each client. Make sure the directories that the path names point to exist before you run the tests, otherwise the tests won't run.

Group Information

s. Choose the Included Groups option from the Advanced menu to specify any client groups that you want ServerBench to use for this mix. To specify a group, click on the button next to the group number. If you want ServerBench to draw from all the groups, you can choose the Select all button (this has the same effect as selecting no groups). If you don't like the groups you've selected, you can either choose Clear all, which removes all the group selections, or just click again on the button next to the group you don't want.

5. To modify another mix in this test suite, use the Previous and Next buttons to go to that mix. repeat steps 3 and 5. If you're happy with your current test suite, choose OK. ServerBench saves the changes and returns you to Mixes in Suite window, where you can work with a different test suite.

Tip:

If you want all the mixes in this test suite to have the same value for a *parameter, choose the Copy mix fields across this suite option from the Advanced menu.*

Setting test parameters to get the best results

Adjusting the test parameters for ServerBench's mixes can also influence your results. So, if you create your own mixes, you might want to keep these points in mind.

Number of transaction iterations. The higher the value you enter here, the greater the weight that transaction will have in the test results. In addition, the more times a client requests a transaction during an iteration of a mix, the longer one iteration of the mix takes.

Number of test iterations. The more iterations you specify for a test, the longer you make the transaction. Because each client measures the amount of time it takes the server to get a transaction and then send a reply, you may see lower TPS scores.

Number of iterations for the Processor test. The more iterations you specify as the Total Size value for the Processor test, the larger the CPU load that the test places on the system. By creating tests with different Processor test iterations, you can improve the way ServerBench stresses the processor subsystem on servers with different processor power.

Disk test file initial size parameter. You can't consider the effect of this parameter without also taking into account the number of clients in the test. By multiplying the value in this parameter by the number of clients, you determine the size of your test area. For example, if you have 20 clients and you set this parameter to 1 MB, then your test area is 20 MB. If you set this parameter to 20 MB, then your test area is 400 MB. The main impact that the combination of the Disk Test File Initial Size and number of clients has is to determine how much of the server's disk you test. The larger this parameter and the more clients running the mix, the greater the amount of disk the tests cover, which increases the amount of time the server spends performing disk seek operations.

Disk test file I/O Range parameter. This parameter defines the contiguous range within the data file that ServerBench reads from or writes to. When you lower this parameter, ServerBench uses less of the disk file for file I/O. When you increase this parameter, ServerBench uses more of the file.

To get a rough indication of how much file cache your server would need to fully cache all the disk accesses, multiply the I/O Range parameter by the number of clients. Then compare that number to the actual size of the server's file cache. The smaller the server's file cache is compared to this value, the more disk accesses that will occur and the greater the stress on the server.

When you run ServerBench so that your I/O test area fits in cache, you may see better scores, but these aren't necessarily the most realistic results. In a real world environment, requests frequently miss the cache.

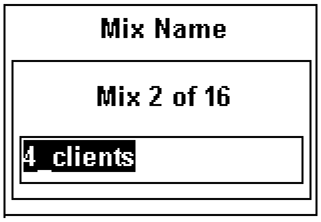
Our prescription is:

ServerBench's standard test suites set the I/O range parameter to 10 MB for a 20 MB disk test data file.

Details on the Mix Name parameter

You enter the Mix Name in the box under the title Mix Name in the Mix Definition window. This name can be any alphanumeric entry up to 15 characters long. Just above the mix name box, ServerBench tells you which mix this is in the test suite, such as Mix 2 of 16.

Figure 4-1: The Mix Name



Mix Name

Mix 2 of 16

4_clients

Related Topics:

What the Mix Name does

What the Mix Name does

This field is optional; however, we recommend that you always supply a descriptive Mix Name. ServerBench uses the Mix Name to identify the mix in the results tables, in the controller window as you run the test suites, and in the Mixes in Test Suite window. The field has no effect on your test results.

Details on the Duration parameters

The Duration parameters are:

- **Ramp up**. This is a fixed number of seconds at the beginning of the mix. ServerBench does not record any results from mix iterations that begin during the Ramp up period. This way your test results aren't skewed because the server load for the first few clients was very light, thus resulting in an artificially high score.

  Try to select a value that gives the server enough time to have reached a steady state. You can verify this by looking at the valid iterations column in Table 3 of the results. If all the valid iterations count for the clients in the mix are consistent, then the Ramp up period is long enough. (The value of Ramp up varies from mix to mix in the ServerBench standard test suites.)

- **Ramp down**. This is a fixed number of seconds at the end of the mix. ServerBench does not record any results from mix iterations that begin during the Ramp down period. This way your test results aren't skewed because the server load for the last few clients was very light, thus resulting in an artificially high score.

  Try to pick a time that is long enough to allow all of the clients to switch from the Running stage to the Sending Results stage of the test. (The value of Ramp down varies from mix to mix in the ServerBench standard test suites.)

- **Length**. This is the minimum number of seconds that the mix executes. (Because ServerBench always finishes the current iteration of a mix, a mix might run longer than the amount of time specified by the Length parameter.) The value for this parameter includes time for both the Ramp up and Ramp down periods.

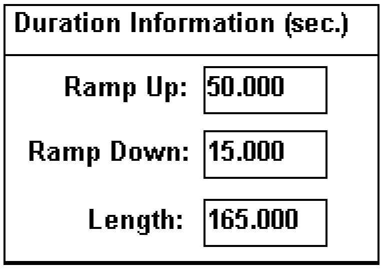
  (The value of Length varies from mix to mix in the ServerBench standard test suites.)

For more information on how you determine a value for these parameters, see the next three sections.

These three parameters apply to all the clients in a mix.

All timing on ServerBench is done in seconds.

Figure 4-2: Duration Information



Duration Information (sec.)

Ramp Up: 50.000

Ramp Down: 15.000

Length: 165.000

Related Topics:

What the Ramp up, Ramp down, and Length parameters do

Choosing values for Ramp up, Ramp down, and Length

Determining how long an iteration takes

What the Ramp up, Ramp down, and Length parameters do

Essentially, ServerBench executes a test mix in three parts:

Ramp up   Recording test information   Ramp down

The clients actually start sending requests to the server during Ramp up period. They continue sending requests until after the end of the Ramp down period (i.e., until they finish their last iteration of the test). However, a client only records TPS information for mix iterations that begin during the Recording test information period. If a mix iteration begins during a Ramp up or Ramp down period, the client does not record any information about the transactions it requests. If the mix iteration begins during the Recording test information period, the client counts how many times it requested each transaction and how long it took to get a response from the server.

NOTE: An iteration is one complete pass through the mix. For example, each time a client has requested all of the transactions in a mix, we say that it has completed one iteration of the mix.

By executing a test mix this way, ServerBench only measures your server when it is in a steady state. We like to measure your server's performance when it is at a steady state because that's when the server's subsystems are executing an appropriate workload for the test ServerBench is running. The server is no longer thrashing as a result of starting the tests and it has flushed out any residual caching effects that occurred as a result of creating test data files. In addition, all the clients are executing the tests, so the server does not have a light load. (A light load occurs when only a few clients are running tests, such as during the Ramp up or Ramp down periods when not all the clients have started the tests yet or some of the clients have ended the tests.)

The actual Recording information part of the test is the time that remains once you subtract the Ramp up time and the Ramp down time from the total time (i.e., Length) for the mix. For example, if you specify a test Length of 60 seconds with a Ramp up time of 5 seconds and a Ramp down time of 10 seconds, ServerBench will actually record information for only 45 seconds.

NOTE: Remember that the test may last longer than the amount of time you set for the length because ServerBench will not stop the test until the last mix finishes.

Choosing values for Ramp up, Ramp down, and Length

The unit of measure you define for the Ramp up, Ramp down, and Length parameters is always in seconds.

There's no fixed formula for deciding what values to supply for these parameters. What you're trying to do is set the Ramp up and Ramp down parameters so that ServerBench only records test results when your server is in a steady state.

As a rule of thumb, you might want to make the Ramp up and Ramp down periods last as long as it takes all the clients to perform one iteration of the mix. If your Ramp up value equals one iteration, then your server should be at a steady state when ServerBench starts recording test results because all the clients will be participating in the test. If your Ramp down value equals one iteration, then, if a client begins an iteration just before the beginning of the Ramp down period, all the other clients will still be requesting transactions of the server when that client finishes. In both cases, all the clients should be executing transactions during the period when ServerBench records results.

In general, you'll want the Length parameter to be long enough so that ServerBench can record the results from several iterations of the mix. The more iterations you have, the more likely you are to avoid gating errors. We recommend that you allow for at least 10 iterations of a mix. So calculate how much time you've specified for the Ramp up and Ramp down periods and then add enough time to that for the number of iterations of the mix you want ServerBench to perform.

NOTE: When you calculate a value for the Length parameter, remember to make sure it is greater than the sum of the Ramp up and Ramp down periods. Otherwise, because the clients ignore the results for any iteration of a mix that starts during Ramp up or Ramp down, the clients won't record any results for that mix.

Determining how long an iteration takes

To figure out how long an iteration takes, run a short test and then look at your results. Table 5 shows you the number of Ramp up iterations, Valid iterations (i.e., those iterations for which ServerBench recorded results), and Ramp down iterations. You can use this information to adjust the values you supply for these parameters.

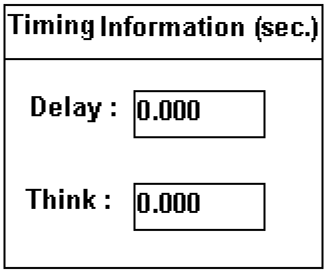
Details on the Timing parameters

The Timing parameters are two other general mix parameters that affect the time a test takes are:

- **Delay**. Enter the maximum amount of time in seconds (or fractions of seconds) that you want a client to wait before starting a test once the controller tells the clients to start. When you set the Delay time parameter, each client takes that value and generates a pseudo-random number. This way the Delay time can vary for each client.   As a result, when you set Delay time, you stagger the clients' initial requests to the server instead of overloading the server with a burst of requests at the beginning of each mix.   (The ServerBench standard test suite **SYS_60.TST** uses a Delay of 0 seconds.)

- **Think time**. Enter the number of seconds or fractions of a second you want a client to wait once it gets a response from the server before it issues another request to the server. (The ServerBench standard test suite **SYS_60.TST** uses a Think time of 0 seconds.)

Figure 4-3: Timing Information



Timing Information (sec.)

Delay : 0.000

Think : 0.000

These two parameters apply to all the clients in a mix.

Related Topics:

What the Delay and Think time parameters do

What the Delay and Think time parameters do

The Delay and Think time parameters help you regulate how much you stress your server.

When you enter a value for the Delay parameter, ServerBench staggers the clients' initial requests to the server. This way the clients don't overload the server with a burst of requests at the beginning of a mix.

Here's how the Delay parameter works: you specify a value in seconds for the Delay time. Each client then generates a pseudo-random number between zero and the Delay value in .1 second increments. When the clients receive a signal to start the next test mix, each client waits its prescribed Delay time and then begins executing the mix. This way the Delay times for the clients vary but do not exceed the amount of time specified by the Delay parameter. For example, if you set Delay to one second, each client will wait between .1 second and one second before sending its initial request to the server. If you set the delay to two seconds, each client will wait between .1 second and two seconds before sending its initial request to the server.

When you set a value for Delay time, make sure that value is shorter than the value for Ramp up.

The Think time is the amount of time you want a client to wait once it has received a response from the server before it sends another request to the server. For example, if you set Think time to 0.5, each client will wait half a second after receiving a response from the server before sending another transaction request to the server.

The shorter this amount of time you enter for Think time, the more the client stresses the server. This is because, in a normal work environment, most users send a request to the server, get a reply, do something with the data they receive, and then issue another request. They don't just take the data and then pelt the server with another request. You can use the Think time parameter to create tests that more closely reflect how users work with a server; however, you won't be stressing the server as much.

Details on defining transactions

The next part of setting up your mix is defining the transactions you want it to perform. A transaction can consist of a single test or several tests. The reason for using transactions instead of just sending tests to the server is that a transaction lets you bundle the tests together. This reduces the number of network requests, which can cause bottlenecks that aren't related to the server's performance.

When a transaction consists of only one test, we call it a singleton. These are the easiest transactions to create because you just enter the test mnemonic in the Transaction Name box.

Each time you set up a transaction, you specify parameters for the test you select. This means that you can have multiple tests with the same name, such as sequential write, but each test is different because its parameters differ from the parameters other tests are using.

The tests you can include in a transaction are:

- **Processor**. There is only one processor test and its mnemonic is P.
- **Disk**. Sequential read (SR), Sequential write (SW), Random read (RR), Random write (RW), and Append (A).
- **Network**. Server to Client (SC) and Client to Server (CS).

Related Topics:

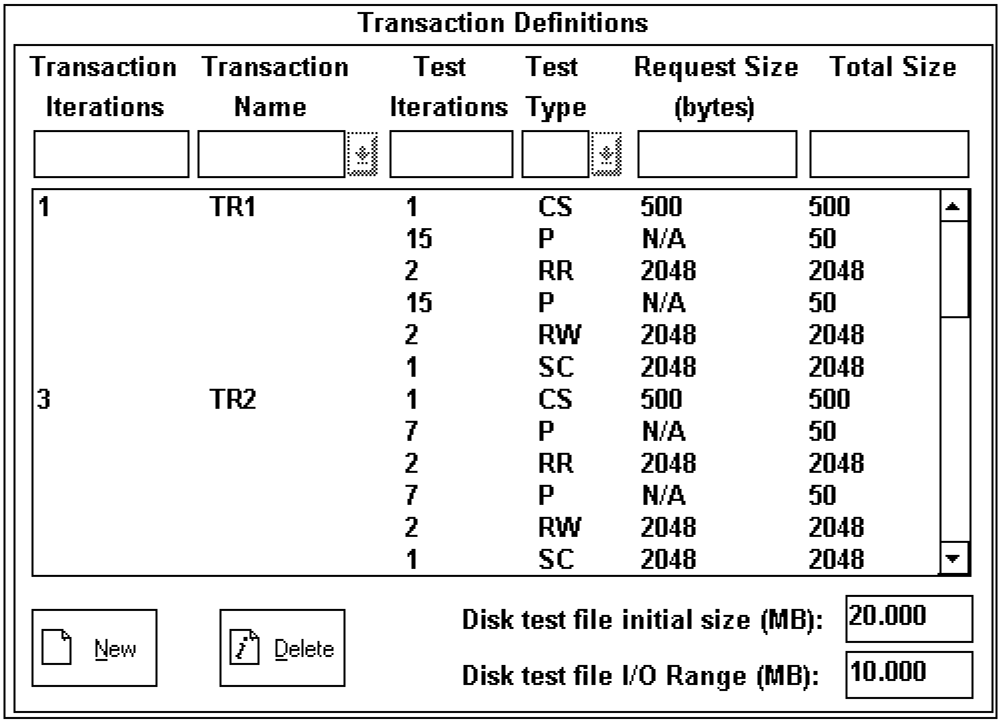
The basic steps for creating a transaction

Here's how the I/O range parameter works

The basic steps for creating a transaction

You define the transactions in the Transaction Definitions section of the Mix Definition window. If you've already created some transactions, you'll see them listed here.

Figure 4-4 Setting up transactions



Transaction Definitions

Transaction Iterations	Transaction Name	Test Iterations	Test Type	Request Size (bytes)	Total Size
1	TR1	1	CS	500	500
		15	P	N/A	50
		2	RR	2048	2048
		15	P	N/A	50
		2	RW	2048	2048
		1	SC	2048	2048
3	TR2	1	CS	500	500
		7	P	N/A	50
		2	RR	2048	2048
		7	P	N/A	50
		2	RW	2048	2048
		1	SC	2048	2048

Disk test file initial size (MB): 20.000

Disk test file I/O Range (MB): 10.000

To create a transaction:

1. Choose the New button. ServerBench moves your cursor to the Transactions Iterations box.

   NOTE: When you're creating several transactions, you'll notice that ServerBench always places the new transaction after the currently highlighted transaction.

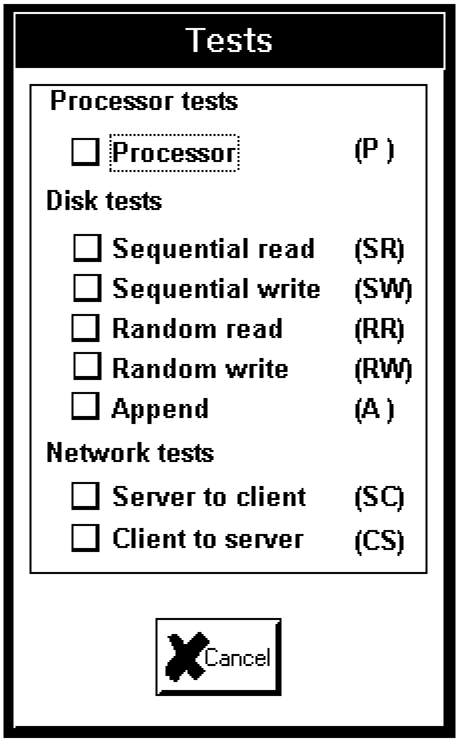
2. Under Transaction iterations, enter the number of times you want each client to request the transaction during one iteration of a mix.

3. Enter the name of the transaction in the box labeled Transaction Name. You can enter any name you like or you can use an existing mnemonic for a ServerBench test. If you choose one of the mnemonics, such as SR, ServerBench will assume you want the transaction to be a singleton; i.e., a single test with the parameters you specify. To see a list of the test mnemonics, choose the down arrow next to this box.

4. Enter the number of times you want the server to execute the test within that transaction in the Test Iterations box. If you are creating a singleton, leave this box blank. ServerBench will only execute the test once.

5. If you did not enter a test mnemonic as your transaction type, then enter in the Test Type box the mnemonic for the test you want ServerBench to run in this transaction. To see a list of the test mnemonics, choose the down arrow next to this box.

Figure 4-5: The tests



Tests

Processor tests

□ Processor (P)

Disk tests

□ Sequential read (SR)

□ Sequential write (SW)

□ Random read (RR)

□ Random write (RW)

□ Append (A)

Network tests

□ Server to client (SC)

□ Client to server (CS)

ServerBench automatically displays this box if you enter an incorrect mnemonic in the Test Type box.

6. Enter a value in bytes for the Request Size. The disk tests and the network tests require this parameter (if you're creating a processor test, ServerBench automatically puts N/A in this box). This is the amount of data ServerBench passes to your server's operating system at one time (i.e., this is the value the ServerBench application on the server uses for the size parameter when it requests file I/O or network I/O). For example, if you select a disk random read test with an Request Size of 512 bytes, ServerBench will read data from your server's disk in increments of 512 bytes.

   NOTE: If the value you supply here is larger than what your network will handle in a single packet, then your network will break this packet into a smaller size.

7. Enter a value for the Total Size. If you're setting up a Processor test, this value will be 1 to 2800 iterations. For the other tests, this value will be 1 to 16777215 bytes. For the Processor test, this is the number of times you want ServerBench to perform the Processor test each time it executes that transaction test. For the disk tests and the network tests, this is the total amount of data that you want ServerBench to move for that test. For example, if your Total Size for the random read disk test is 51200 bytes and your Request Size is 512 bytes, ServerBench would read data from your server's disk in increments of 512 bytes until it had read a total of 51200 bytes of data.

8. Enter a value from 0.001 MB up to 1024.000 MB for the Disk Test File Initial Size. This parameter tells ServerBench how big to make the data files it creates for the disk tests. During the Append test, these files can grow beyond this initial size.

9. Enter a value from 0.001 MB up to 1024.000 MB for the I/O Range Size. This parameter tells

ServerBench to set aside a contiguous section of the data file and only use that section for file I/O.

10. To add another test to your current transaction:

    a. Choose the New Test button.

    b. Go to the Test Iterations box; i.e., TAB to the Test Iterations box. (*Do not* enter anything in the Transaction Iterations or Transaction Name boxes, or ServerBench will set up another transaction.)

    c. Now follow steps 4 though 9.

    d. Repeat these steps until you have included all the tests you want to include in this transaction.

You can delete a transaction from your list of transactions by selecting the transaction (i.e., highlighting it) and then choosing Delete.

Here's how the I/O range parameter works

The I/O range parameter is another ServerBench feature designed to give you more control over the way you test your server. This parameter lets you increase the size of your test data files without having to increase the amount of file cache your server needs to perform the test.

The I/O range parameter sets aside a contiguous section of the test data file (i.e., a range) of the size you specify. ServerBench then performs all its I/O operations in this range; it does not perform file I/O operations anywhere else in the data file.

The starting point for the I/O ranges varies for each client. For example, if each client has a 20 MB data file with a 10 MB I/O range, then the I/O range for Client1's data file might look like:

5 MB data   <--   I/O range for 10 MB data   -->   5 MB data

while the I/O range for Client2's data file might look like:

7 MB data   <--   I/O range for 10 MB data   -->   3 MB data

With earlier versions of ServerBench, if you wanted to use more of your server's disk, you needed to increase the size of the test data files. As the size of the data files increased, they used up the available file cache quicker. This meant the ServerBench tests broke out of cache quicker. For example, four clients that each had a 20 MB data file would try to use 80 MB of file cache. The only way to decrease the amount of file cache was to reduce the size of the data files.

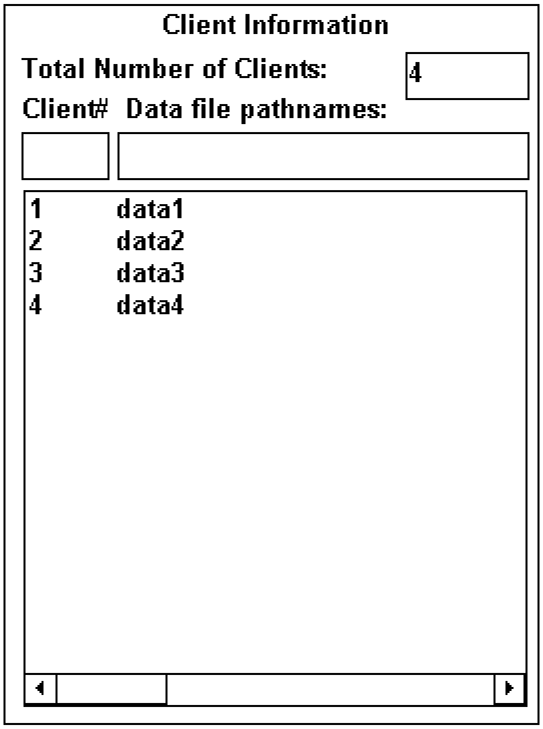
With ServerBench 3.0, you still must increase the size of your test data files when you want to test more of your disk; however, you can adjust the amount of file cache that the tests use. The I/O range parameter lets you determine how much file cache each client uses. Now, if you have 20 MB data files for four clients, you can specify an I/O range 7 MB. This means you would use 28 MB of file cache instead.

Details on entering the client information

There are two pieces of information that ServerBench looks for in the Client Information section of the Mix Definition window. They are:

- **Total Number of Clients**. Enter the maximum number of clients you want to have running this test mix. If you don't have this many clients connected, the mix will run with as many clients as you do have connected. (Each mix in the ServerBench standard test suites increments the number of clients running the tests.)
- **Data file path name.** This is the path name to the place on the server where you want ServerBench to create the test data files that the clients use for disk I/O during the disk tests. By default, ServerBench creates these files in the ServerBench directory on the server. The standard test suites use the file names **DATA1**, **DATA2**, and so on up to **DATA60**. If you enter a full path name in this field, make sure the directories you specify exist before you execute any disk tests.

Figure 4-6: The Client Information



Client Information

Total Number of Clients: 4

Client#	Data file pathnames:
1	data1
2	data2
3	data3
4	data4

You can also use the Mix Definition window to restrict the mix to specified groups of clients. See the section "Including client groups" for information on doing that.

Related Topics:

How to enter the client information

Including client groups

How to enter the client information

The data file path name gives the full path and file name on your server of the data file for each client. You must supply a unique data file for each client. To do this, make sure you have privileges for the directory you specify and that the directory path is valid. ServerBench verifies the data file path name when you run the test.

To tell ServerBench how many clients you want the mix to include and where to find the data file path names:

1. Enter a value for Total number of clients. This is the maximum number of clients that ServerBench will include in this mix.

   NOTE: If you don't have enough clients connected to the server, ServerBench will run the mix with fewer clients than the maximum you specified.

2 Supply the data file path names for each client. There are four ways you can do this. You can:

- **End the path name with an asterisk.** When you enter the path name this way, ServerBench treats the path name as template. This option lets you create paths for all clients at once. ServerBench will automatically append a number to that path name and enter it as the path name for each client up to the maximum number of clients. So, if you specified 60 clients, and you entered the path name **test***, ServerBench would create the path names **test1**, **test2**... up to **test60**.
- **Copy client paths from ASCII file.** You choose this option from the Advanced menu. This option tells ServerBench to copy paths from a text file that contains a list of path names. To create this file, enter each path name on a separate line and end the line with a carriage return. If the file contains more client path names than you specified, ServerBench resets the total client count to match the number of path names in the file. Thus, if you had specified a data file with 4 clients and you supply an ASCII file with 8 clients, ServerBench resets the total number of clients to 8 and enters all 8 path names.
- **Copy fields from another mix.** You choose this option from the Mix menu. This option lets you copy the path names from an existing mix. Once you choose this option, ServerBench will ask you to choose an existing test suite. Then it will ask you to select the mix you want to copy information from. Once you have the mix highlighted, click on the Copy client information button. If the mix has a different number of clients than you specified, ServerBench resets the total client count to match the number in the mix, regardless of whether that number is higher or lower than the one you specified. For example, if you specified a total of 8 clients and the mix only has 4 clients, the ServerBench resets the total number of clients to 4. However, if the mix has 12 clients, then ServerBench increases your total number of clients to 12 as well.
- **Type in each path name.** To do this, click on each client in the list box. ServerBench then displays that number in the Client# box and lets you enter a path name in the Data file path names box.

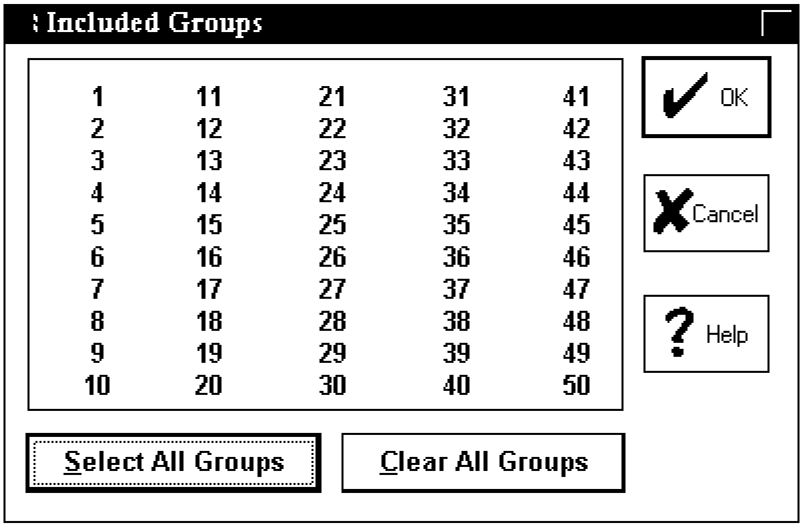
Including client groups

ServerBench lets you restrict the mix to certain client groups. You assigned group numbers to clients when you installed ServerBench on the clients. (For more information on setting up clients, see the on-line ServerBench manual for your server operating system.)

You don't have to specify any group numbers for a mix to run.

You get to the Included Groups dialog box by choosing the Groups option from the Advanced menu. To restrict the mix to certain client groups, click on the group number or numbers.

Figure 4-7: Defining groups of clients



1	11	21	31	41
2	12	22	32	42
3	13	23	33	43
4	14	24	34	44
5	15	25	35	45
6	16	26	36	46
7	17	27	37	47
8	18	28	38	48
9	19	29	39	49
10	20	30	40	50

You can also use the Groups function buttons to:

- **Select all** groups of clients to be in the mix (this has the same effect as selecting no groups). If you plan to use most groups in the mix, you can choose Set all and then just click on the group numbers you don't want included. When you click on a highlighted group number, ServerBench deselects it.
- **Clear all** the groups you had specified. Clear all lets you remove the current group assignments. You can then just click on the group numbers you want ServerBench to include in this mix.

Related Topics:

How the client groups feature works

How the client groups feature works

When you use client group numbers, you can run sets of clients together during a test. This way you can increase your server's work load incrementally. For example, you may want to run a test suite with 20 clients, then 40 clients, and then 80 clients. This way you can spot where the knee in the performance curve appears. (Or you could just increase the number of clients in each mix and let ServerBench select which clients.)

You can also use the group feature to place clients in a single network segment in one group. Then, you can bring an entire segment on-line at once. To get information about work loads across segments, number the clients consecutively on each segment. Include equal numbers of clients from each group in the segment. Now create a mix that defines the number of groups you want and the number of clients from those groups.

If you specify any group numbers for the mix, ServerBench will only let clients with those group numbers run in the mix. For example, suppose you have groups 1 through 5, each of which has 10 clients. When you defined the mix, you specified that it include 50 clients but only from groups 1 and 2. Because you specified groups 1 and 2, ServerBench will only use clients in those groups. Since both groups combined only have 20 clients, 20 is the maximum number of clients that ServerBench will use for this mix, even though you specified a total of 50 clients when you set up the mix.

If you don't specify any group numbers or if you have more clients than the mix needs, the server decides which clients to use based on the client ID number. ServerBench always uses the lowest ID numbers first. This means that, if you have a client with the ID number 1, that client will always run the tests anytime you don't specify a group number.

Even when you specify a group number, ServerBench gives the clients with lower numbers priority over the clients with higher numbers. For example, suppose you're using clients from Group 1 and Group 1 includes clients with ID numbers 10 through 25. In this case, client 10 always runs a test, regardless of whether you specify one client in the mix, four clients, or 25 clients.

This means that, if you connect more clients than a mix uses, you will always get the same set of clients for a given mix.

Creating test mixes to stress your server

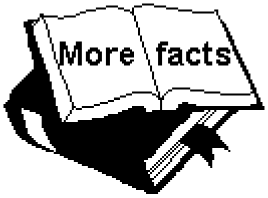
If your goal is to stress your server to the point where TPS drops off, then you'll want to do the following:

- Increase the initial size of the data file that the disk tests use.
- Increase the total size of the disk tests.
- Reduce the Think time parameter.
- Use a large value for the I/O range parameters.
- Increase the number of clients in the test.

Determining what to test on your system

As you look at the ServerBench tests, consider what would be the most useful information you can get about your system. For example:

- What subsystems do your primary software applications use the most? If you use applications that spend a lot of time accessing the disk, then you may want to spend more time running tests that use the ServerBench disk tests.
- Where are the bottlenecks in your system? To determine where your system encounters bottlenecks, you may want to run three different test suites, where each test suite focuses on a different server subsystem.
- Have you set the test parameters to provide a reasonable test of your system? For example, if you are using just one Ethernet for a test that uses 200 clients and no Think time, you won't get a valid score for your server. Instead, you'll be measuring the speed of a heavily congested Ethernet.
- Are there things you could do to optimize your server's performance? (See the on-line ServerBench installation manual for your server operating system.)
- What affect does changing a parameter have your results? You can experiment with changing system parameters and comparing the ServerBench results before and after the change.



Additional information:

To see what affect changing parameters has, make sure you only change one parameter at a time between ServerBench test runs. This way you can gauge the effect that parameter has on performance. The server parameters you may want to change are:

- Amount of server RAM.
- Number of CPUs
- Number of disks.
- Disk organization (mirroring versus striping).
- *Network topology.*

- How do different servers perform? By running the same test suites with the same settings on different servers, you can determine which server best meets your needs.

End of chapter

Overview

This chapter explains the screens you see when you run ServerBench on the controller and the clients. Both the controller and each client display graphical ServerBench screens. You use the screens that appear on the controller window to start ServerBench, run tests, monitor the tests as they run on each client, view results, create test suites, and perform other ServerBench tasks. The screens that appear on each client are for information purposes. They let you know which phase of the test each client is in (for example, running).

The main ServerBench window

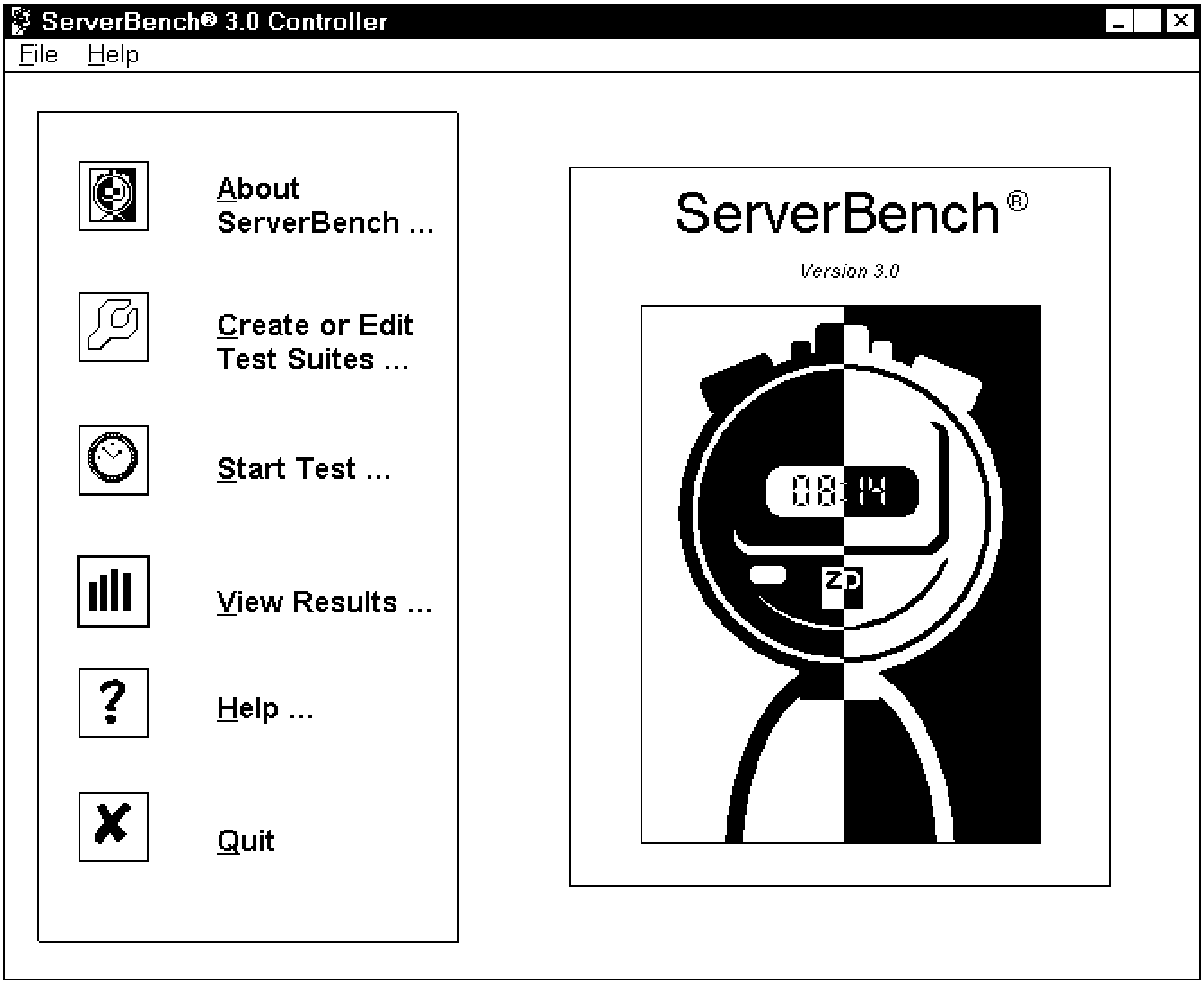
When you start ServerBench on the controller, the main ServerBench window appears. This window contains the ServerBench function buttons. From the ServerBench main window, you can:

- Get information about the ServerBench development team and re-read the ServerBench License Agreement.
- Create and edit test suite files.
- Start a ServerBench test.
- View test results.
- Use the on-line help facility.
- Exit ServerBench.

If you want to get to the controller window, choose the Start Test button. This brings up the controller window on top of the main ServerBench window.

NOTE: The main ServerBench window stays on your screen even when other windows appear.

Figure 5-1: The main ServerBench window



ServerBench® 3.0 Controller
File
Help
About ServerBench ...
Create or Edit Test Suites ...
Start Test ...
View Results ...
Help ...
Quit
ServerBench®
Version 3.0
08:14
ZD

The controller window

The controller window is probably the window where you will spend most of your time. This is the window you use to run the ServerBench test suites and to monitor the clients.

You get to the controller window by choosing the Start Test button (or selecting Start Test from the drop-down File menu) on the ServerBench main window. The controller window then appears on top of the main ServerBench window.

NOTE: If you want to return to the main ServerBench window from the controller window you can:

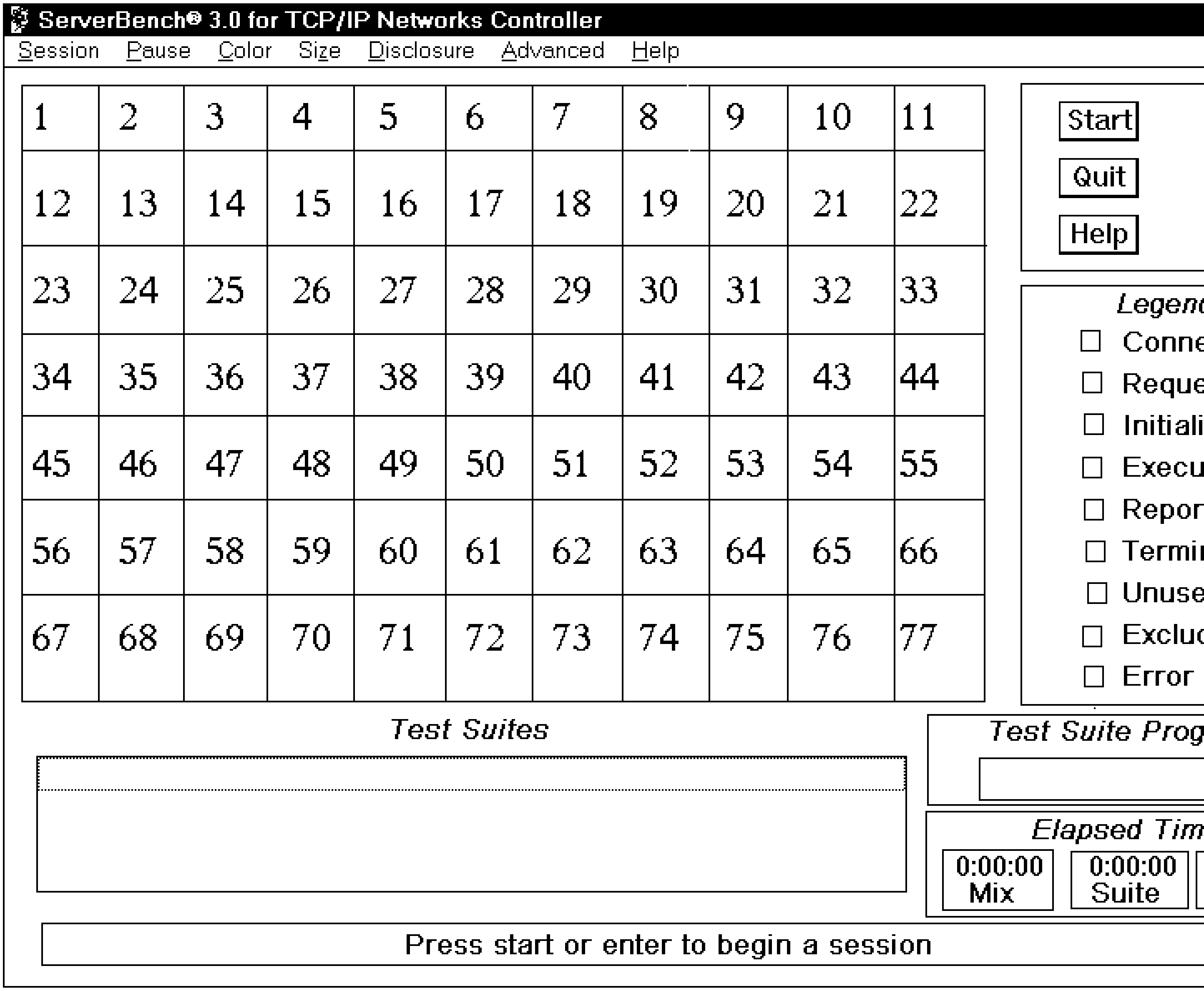
- Enter the key sequence ALT-TAB. This key sequence switches you between the controller window and the main ServerBench window.
- If any tests are running, choose Quit and select the Abort a test option. This brings you back to the "Do you want to select a test suite" dialog box. Say no.

  If no tests are running and you have the "Do you want to select a test suite" dialog box, say no.
- Minimize the controller window by clicking on the down caret in the upper right corner of the window. When you minimize the controller window, ServerBench saves its current state and continues to run any tests that were executing. When you maximize the window, you can view your test suites' status.
- If part of the main ServerBench window is visible, click on it to make it active.
- Use the right mouse button to click on the ServerBench icon in the upper right corner of the controller window.

From the controller window, you can:

- Select a test suite.
- Start a test.
- Halt a running test.
- Add a test suite.
- Specify whether ServerBench pauses between test mixes.
- Monitor the different stages the clients are at in the tests and get other client information.
- Monitor the progress of the test suite you are running.
- Check the status of the test suites you've selected during this test session.

Figure 4-2: The ServerBench controller window



1	2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21	22
23	24	25	26	27	28	29	30	31	32	33
34	35	36	37	38	39	40	41	42	43	44
45	46	47	48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63	64	65	66
67	68	69	70	71	72	73	74	75	76	77

0:00:00 Mix

0:00:00 Suite

Press start or enter to begin a session

This window contains several key parts:

- The client grid.
- The Legend of client activities.
- The Test Suites box.
- The status information.
- The function buttons.
- The menu options.

NOTE: If your controller's running Windows for Workgroups, you can run your test suites with the controller window minimized. The controller icon displays information about the running tests.

While the controller window is minimized, you can perform other tasks and look at results for mixes that have completed. However, if you try to view the results of a currently running mix while it is reading or writing to a file, ServerBench will abort the running test.

To avoid creating a problem with a running test, you can use the Pause between mixes option in the drop-down Pause menu and view the test results while ServerBench is displaying the Connecting clients box. (Make sure, though, that you *do not* choose the Delete log files option in the View Results dialog box if you plan to run the rest of the mixes in that test suite.)

Related Topics:

The client grid and color legend

The test suites box and test status information

The controller window's function buttons and menus

The Test Suite History window

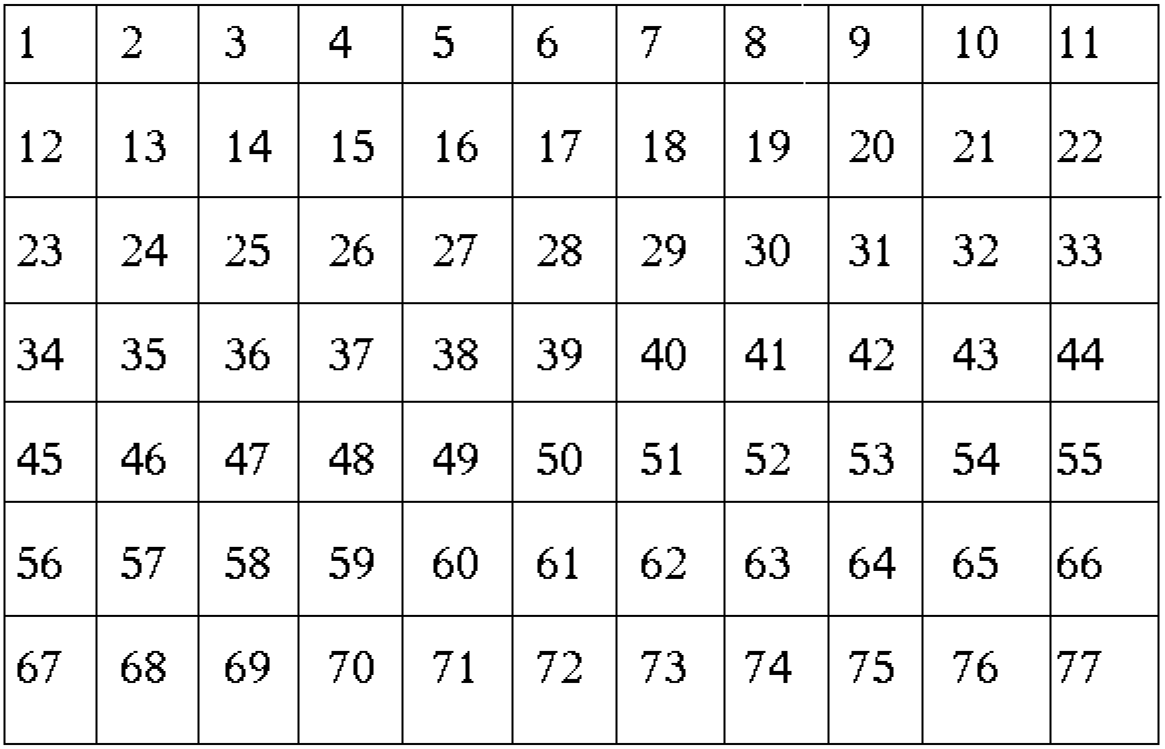
The Unattended Mode dialog box

The External Notification dialog box

The client grid and color legend

Probably the first thing you will notice in the controller window is the client grid. This grid contains a square (i.e., button) for each client that is running ServerBench. The number in the grid square corresponds to the client identification number. By clicking on a grid square, you can get information on that client.

Figure 5-3: The client grid in the controller window



1	2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21	22
23	24	25	26	27	28	29	30	31	32	33
34	35	36	37	38	39	40	41	42	43	44
45	46	47	48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63	64	65	66
67	68	69	70	71	72	73	74	75	76	77

You may also see numbers for clients that are not running ServerBench. There are two reasons for this:

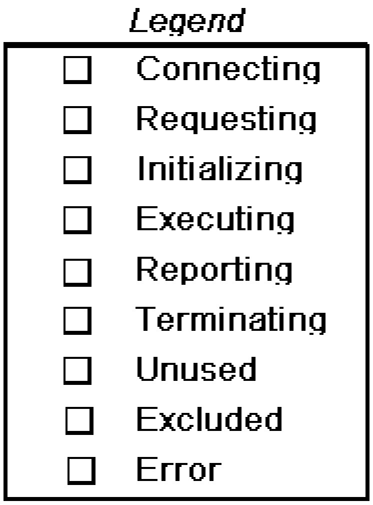
- ServerBench displays all numbers up to the highest identification number of connected client. This means that, if you only connect one client with an identification number of 160, ServerBench also displays numbers 1 through 159 in the grid.

- The client grid has only five sizes – 77, 160, 260, 400, and 1,000 – and displays all the numbers in one of these size ranges, except for 1,000. By default ServerBench displays the grid size that includes the highest identification number of a connected client. This means that if you only connect one client with an identification number of 165, ServerBench displays a grid with numbered squares that go up to 260.

  NOTE: ServerBench does not display the client numbers when you display the grid with 1,000 clients because the grid squares are too small.

Another feature of the grid is that it is color-coded. You can tell at a glance which stage a client is in by looking at the color of that client's square. The Legend of colors located to the right of the client grid tells you which color is associated with which client stage.

Figure 5-4: The color legend in the ServerBench controller window



Legend

- □ Connecting
- □ Requesting
- □ Initializing
- □ Executing
- □ Reporting
- □ Terminating
- □ Unused
- □ Excluded
- □ Error

The legend displays a color for each of the following client stages:

- Connecting. Default color is white.
- Requesting. Default color is yellow.
- Initializing. Default color is light blue.
- Executing. Default color is green.
- Reporting. Default color is dark blue.
- Terminating. Default color is magenta.
- Unused. Default color is gray.
- Excluded. Default color is black.
- Error. Default color is red.

When a client is at one of these stages, its square turns the color associated with that stage. So, if client 65 is running a test, its square is green (or whatever color you assigned to the Executing stage for this test run).

NOTE: Unless a client is connecting or issuing an error message, ServerBench updates the colors for all clients at once. As a result, when you monitor ServerBench's status from the controller, it will probably look like all the clients complete a stage at the same time even though this is not what is really happening. ServerBench updates all the clients at one time to avoid loading the server down with messages during a test.

You can use the drop-down Color menu to change colors associated with the stages. You can also change colors by clicking in the Legend box next to one of the stages. ServerBench will save any changes you make to the colors and then use those colors the next time you start a ServerBench session. (You can return to the default ServerBench colors by selecting the Load Default Colors option from the Color menu.)

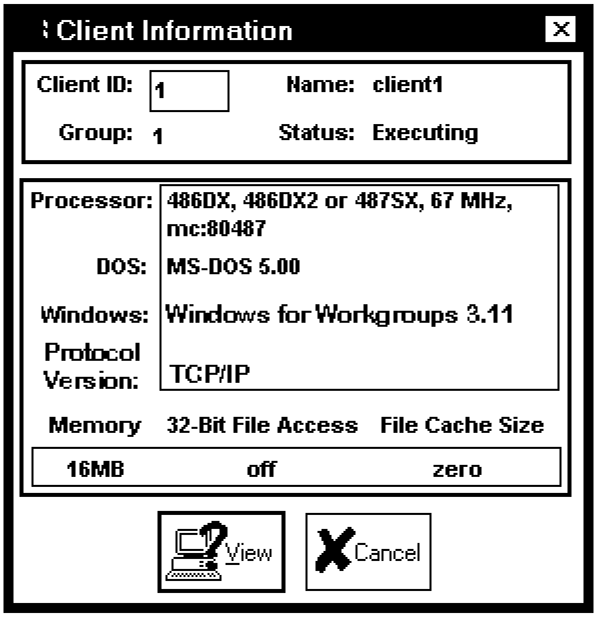
NOTE: ServerBench lets you change the colors in the Legend even when the test is running. This does not affect your server's score.

You can also use the client grid to get specific information on an individual client. When you click on the client's square, ServerBench displays a pop-up box that tells you the client's ID number, its name, its group number, and its status in real time (i.e., at that moment you select it). ServerBench updates the client information each time you select a client's square.

This box also includes the same information that is in the disclosure database about the client's processor, the version of DOS it is using, the version of Windows it is using, the amount of memory, it has, whether its using 32-bit file access, and what it's file cache size is.

If you'd like to see the status of another client, enter that client's number in the Client ID portion of this box or click on the appropriate client square.

Figure 5-5: Client information in the controller window



Client Information

Client ID: 1 Name: client1

Group: 1 Status: Executing

Processor: 486DX, 486DX2 or 487SX, 67 MHz, mc:80487

DOS: MS-DOS 5.00

Windows: Windows for Workgroups 3.11

Protocol Version: TCP/IP

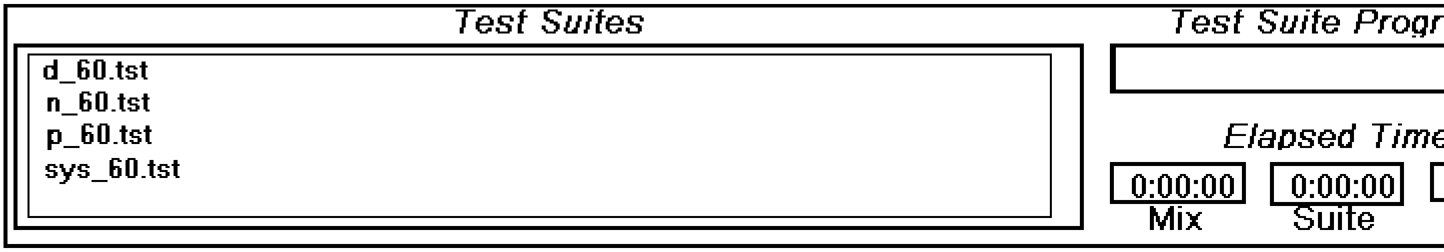
Memory	32-Bit File Access	File Cache Size
16MB	off	zero

The test suites box and test status information

You can use the information at the bottom of the controller window to monitor the status of the test you are running.

In the box in the bottom left of this window, ServerBench displays the names of the test suites you are running.

Figure 5-6:   Information on the test status



d_60.tst
n_60.tst
p_60.tst
sys_60.tst

In the bottom right of the controller window, ServerBench displays information on the status of the tests you are running. ServerBench tells you:

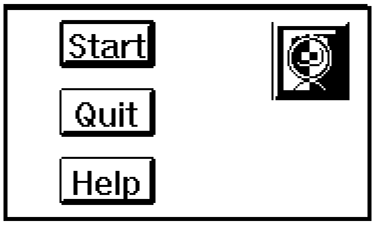
- How many of the mixes in the test suite have completed. (This is the first number in the box under the heading "Test Suite Progress.")
- The total number of mixes in the test suite. (This is the second number in the box under the heading "Test Suite Progress.")
- The time spent on the current mix.
- The time spent on the current test suite. This is the elapsed time spent on all the mixes, not just the time that has passed since you started this test suite. It does not include the amount of time ServerBench may have spent pausing between mixes.
- The total time spent on the test. This is the elapsed time spent on all the tests; it is not just the time that has passed since you started a test (i.e.,   ServerBench does not include the amount of time it spent pausing between mixes or test suites here). For example, suppose the test started at 2 p.m. and it's now 4 p.m. During this time, ServerBench spent 30 minutes pausing between mixes and between test suites. So the total time on the test is 1.5 hours, not 2 hours.

The controller window's function buttons and menus

The controller window also contains function buttons and drop-down menus you can use.

There are three function buttons in the upper right corner of the controller window. These buttons let you:

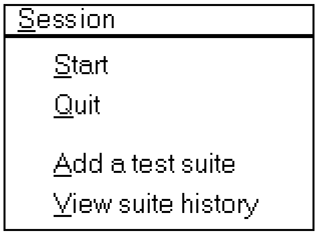
Figure 5-7: Function buttons in the controller window



- **Start**. Use this button to start a test.
- **Quit**. This button lets you interrupt or abort a running test. You can also use this button to return to the main ServerBench window. If any tests are running when you choose Quit and select the Abort a test option, ServerBench displays the "Do you want to select a test suite" dialog box. If no tests are running, choosing Quit causes ServerBench to close the controller window and display the main ServerBench window.
- **Help**. Clicking on this button brings up ServerBench's on-line Help.

There are seven drop-down menus that you can use from the controller window. They are:

- **Session**. From the Session menu, you can:



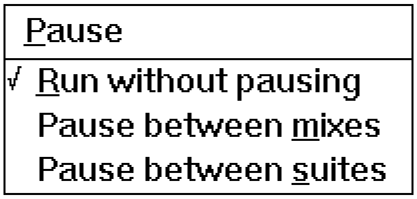
Start a ServerBench test session.

Quit (or abort) the controller window or abort a test.

Add a test suite.

View the test suite history window, which shows you the status of all the test suites you selected during this test session.

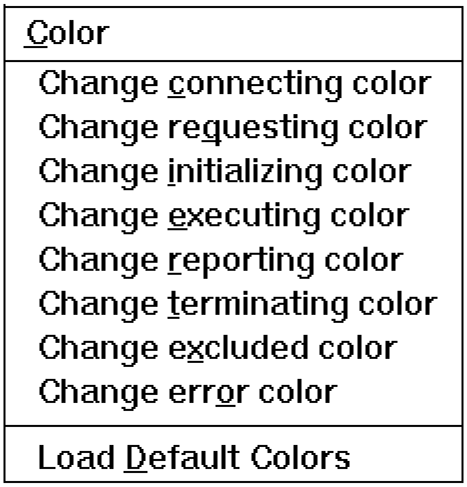
- **Pause**. You can select the Pause menu options at any point during a test run. From the Pause menu, you can:



When you specify a pause option, ServerBench runs the test mix or the test suite (whichever you specify) and then waits for you to tell it to continue with the next test mix or test suite. While ServerBench is pausing, you can connect additional clients, add another test suite to the list of selected test suites, or check on any clients that may be hung or displaying an error message.

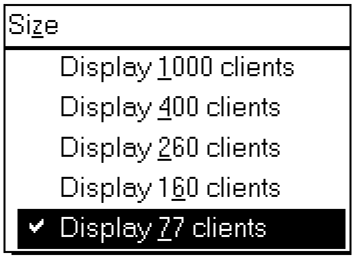
NOTE: If you are planning on running ServerBench without having someone there to monitor the tests (for example, overnight) *do not* select the pause option. Once ServerBench pauses, it will not continue until you choose OK in the Connect more clients dialog box. (ServerBench will warn you if you try to choose a pause option when you have unattended mode enabled.)

- **Color**. You can use the options in the Color menu to change the colors in the Legend located next to the client grid for your current ServerBench session. ServerBench will then use the colors you set until you change them. If you want to return to the ServerBench default colors, you can select the option Load Default Colors. Here are your options in the Color menu:



- **Size**. The Size menu lets you specify the number of squares ServerBench displays in the client grid. Your options are 77, 160, 260, 400, or 1,000. By default, ServerBench displays the grid that includes the highest client identification number.

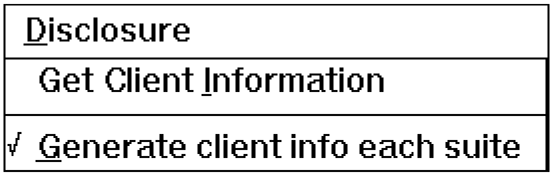
NOTE: When you display the grid with 1,000 squares, ServerBench does not display the client numbers; the grid squares are too small.



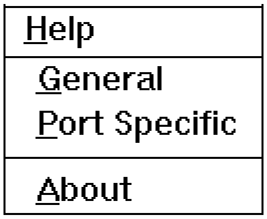
- **Disclosure**. If you run a test suite without having "Generate client info each suite" checked, then ServerBench won't save any client disclosure information. In addition, ServerBench won't display Table 5 of the results workbook because it won't have any disclosure information.

NOTE: We recommend that you always have this option selected when you run ServerBench.

The Get Client Information gives you the same information you get when you click on a client's square.



- **Help**. You can use the Help menu to start ServerBench's Help.



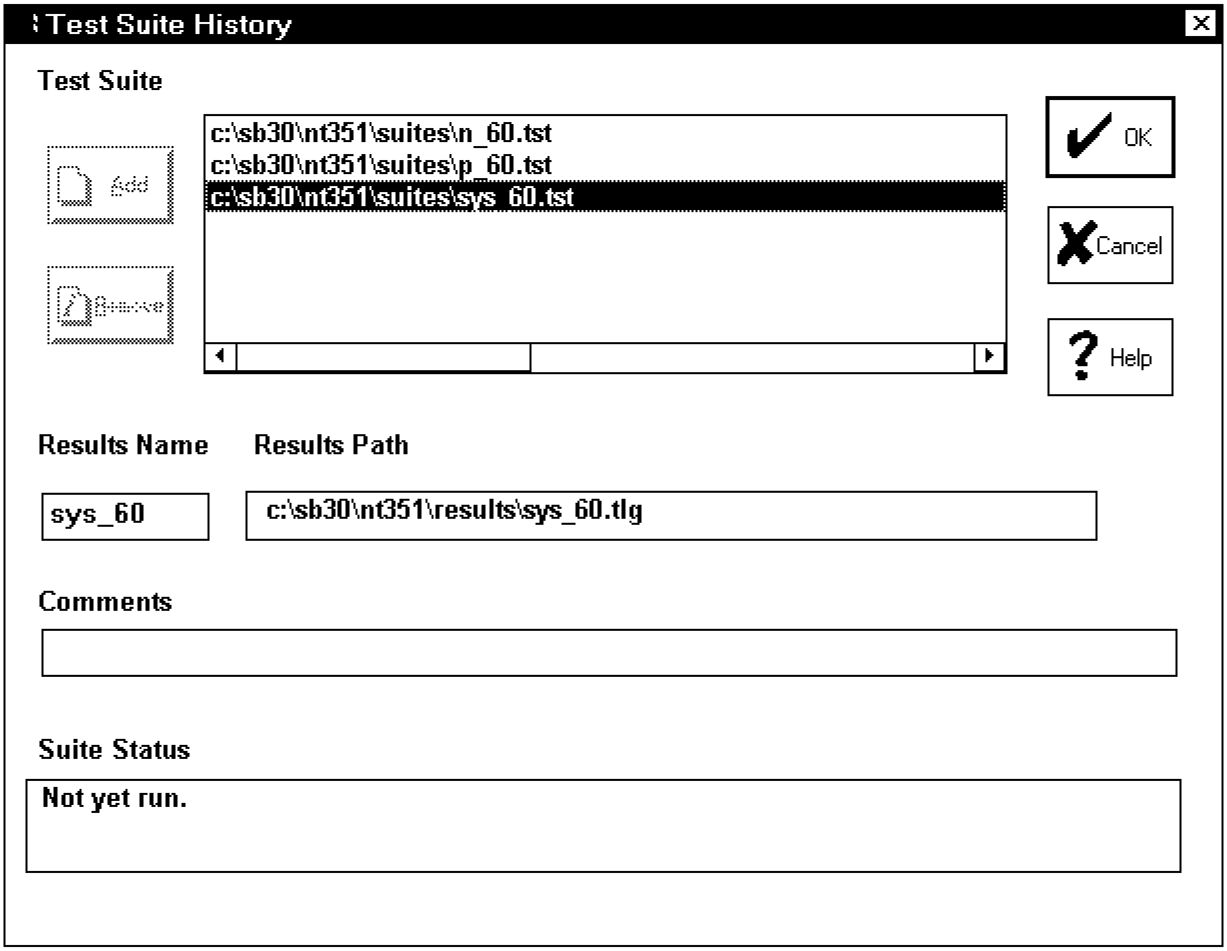
The General help contains most of the information you'll find in this manual. The Port Specific help contains most of the information you'll find in the on-line ServerBench manual for your server operating system. The About option takes you to the ServerBench About screen.

The Test Suite History window

You can use the Test Suite History window to check on the status of all the test suites you've selected during the current test session. When you click on a test suite name in this window, ServerBench displays the path name for that suite's results file. It also displays any comments you entered about that test suite as well as information about whether ServerBench skipped any mixes in that suite. If you request the status of a running test suite, ServerBench displays the status in real time; i.e., the current status of that test suite at that moment.

You can get to this window by choosing View Suite History from the drop-down Sessions menu in the controller window or by clicking on a test suite in the Test Suites box on the controller window.

Figure 5-8: Test Suite History window



Test Suite History

Test Suite

c:\sb30\nt351\suites\n_60.tst
c:\sb30\nt351\suites\p_60.tst
c:\sb30\nt351\suites\sys_60.tst

Results Name: sys_60

Results Path: c:\sb30\nt351\results\sys_60.tlg

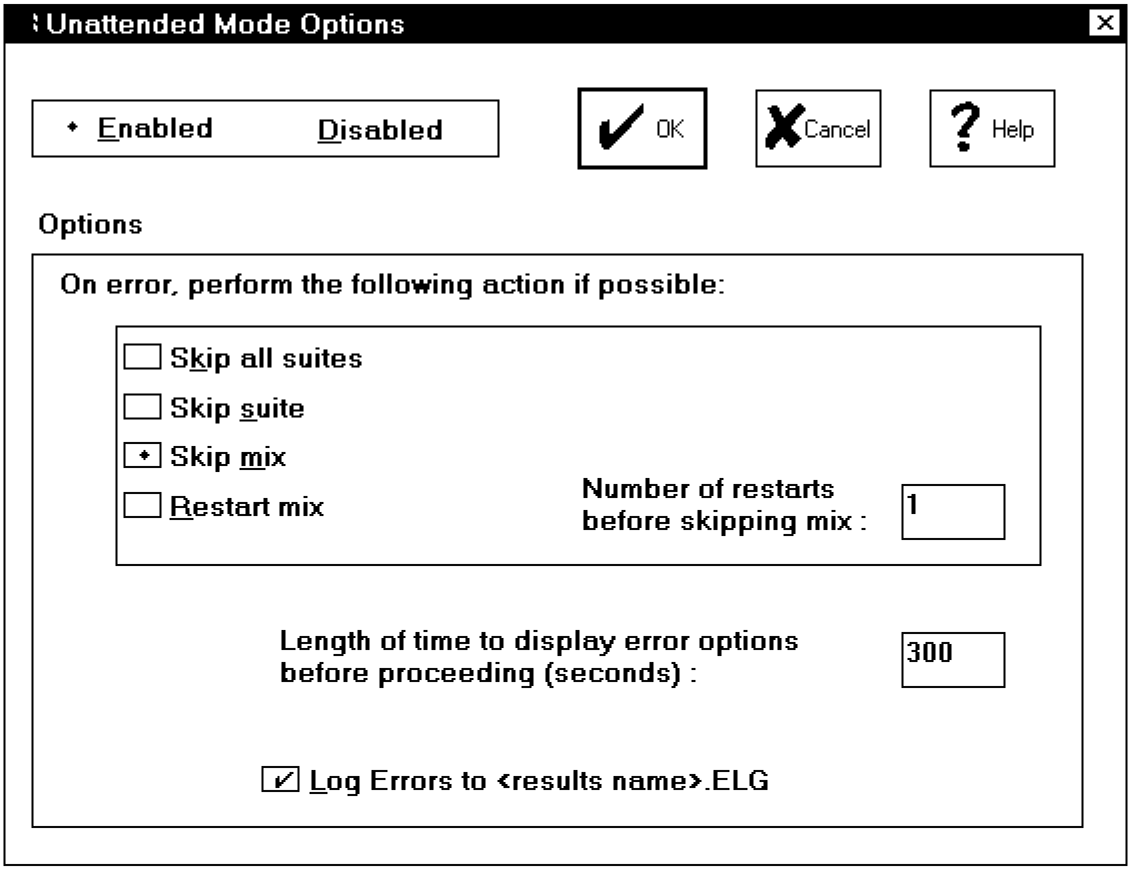
Comments:

Suite Status: Not yet run.

The Unattended Mode dialog box

ServerBench displays the Unattended Mode dialog box when you choose the option Unattended Mode from the Advanced menu in the controller window.

Figure 5-9:   The Unattended Mode dialog box



Unattended Mode Options

● Enabled   ○ Disabled

Options

On error, perform the following action if possible:

□ Skip all suites

□ Skip suite

● Skip mix

□ Restart mix

Number of restarts before skipping mix : 1

Length of time to display error options before proceeding (seconds) : 300

☑ Log Errors to <results name>.ELG

At this dialog box, you can:

Enable or disable unattended mode. (By default, unattended mode is enabled.)

Skip all test suites if an error occurs. This will end that test run.

Skip the current test suite. ServerBench will start running the next test suite, if there is one.

Skip the mix that is reporting the error. This is ServerBench's default.

Restart the mix reporting the error. If you select this option, you need to specify how many times you want ServerBench to try to restart the mix before skipping the mix. ServerBench resets the restart counter each time it starts a mix.

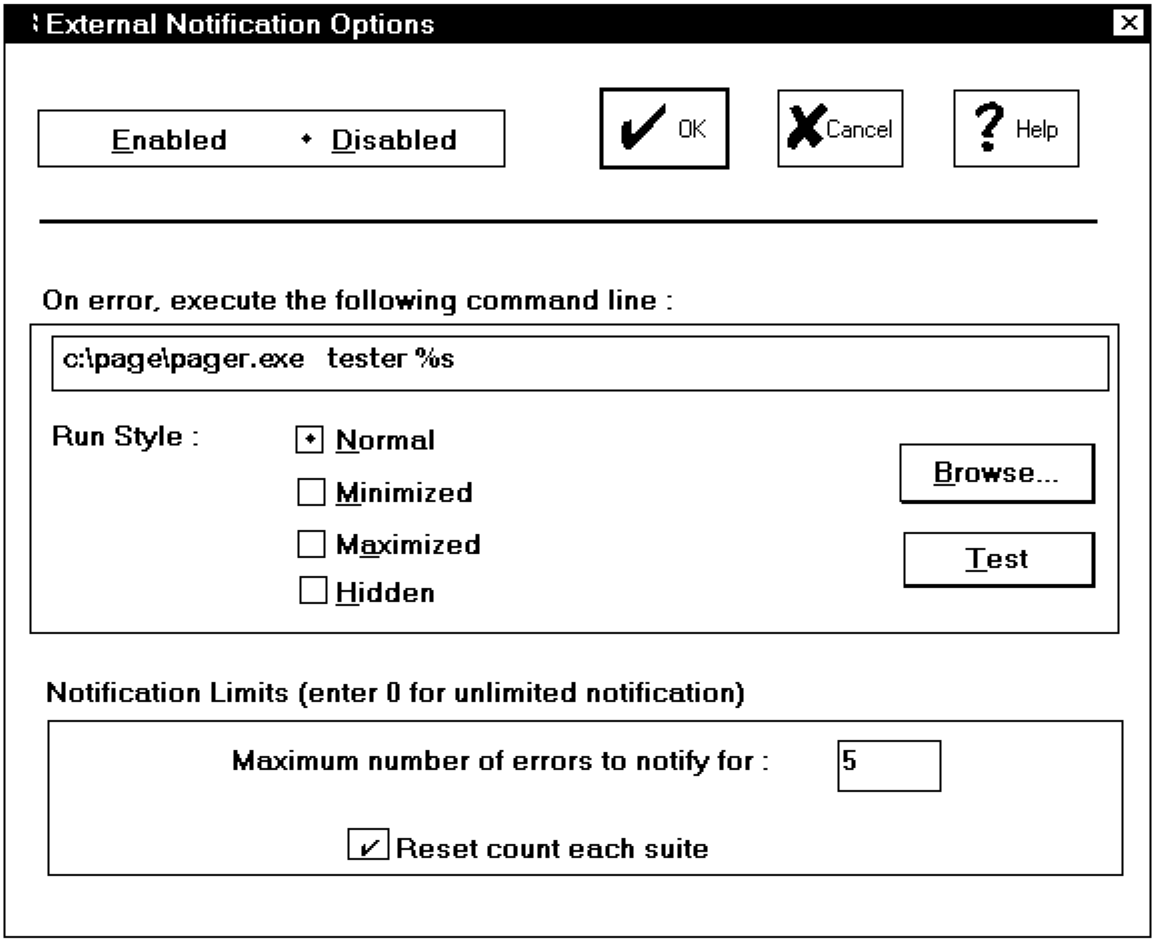
The amount of time you want ServerBench to wait before taking the action you specified. (We recommend that you set this to at least 60 seconds if you have external notification enabled. The paging program needs time to connect and disconnect.)

Write the error message to an error log (**.ELG**) file. By default, ServerBench writes the message to a file that uses the name ***<results-name>*.ELG** and saves this file in the controller's **RESULTS** subdirectory (or the results directory you specified when you selected the test suite). ServerBench only writes error messages to this log file when you have it running in unattended mode.

The External Notification dialog box

The External Notification dialog box appears if you choose that option from the Advanced menu.

Figure 5-10: The External Notification dialog box



Enabled ◆ Disabled

On error, execute the following command line :

c:\page\pager.exe tester %s

Run Style :
☒ Normal
□ Minimized
□ Maximized
□ Hidden

Notification Limits (enter 0 for unlimited notification)

Maximum number of errors to notify for : 5

☑ Reset count each suite

When you enable this dialog box, you can enter a command that the controller will execute if an error occurs. This command line can start any program, such as an executable, a **.bat** file, a **.pif** file, or an **.exe** file program. If you specify a pager program that dials a specified pager number and that program accepts alphanumeric pagers, you can actually send the error message to the pager.

If you include `%s` argument in this command line, the controller will replace that argument with the error message. For example, if you're executing a pager program that supports alphanumeric pagers, you can get it to display the error message when it pages you by entering a command line similar to:

```
c:\pager.exe tester %s
```

This command line tells the controller to execute the pager program and display the error message.

To see what will happen when the controller executes the command line, choose the Test option. This option lets you try the command line before starting the test suite. If you include **%s** in the command line, the string you'll see when you choose Test is:

```
SeverBench 3.0: Testing external notification feature... Success!
```

You can also specify the run style you want the controller to use when it executes the program you specified. You can select any of the standard Windows run styles: Normal, Minimized, Maximized, or Hidden.

In addition, you can tell the controller the maximum number of times you want it to execute this command line. If you enter 0 (i.e., unlimited), the controller will execute the command line you entered each time an error occurs. However, if you enter a number greater than 0, then, once the error count exceeds the notification count, the controller will no longer execute the command line.

If you like, you can also tell the controller to reset the notification count when it starts a new test suite.

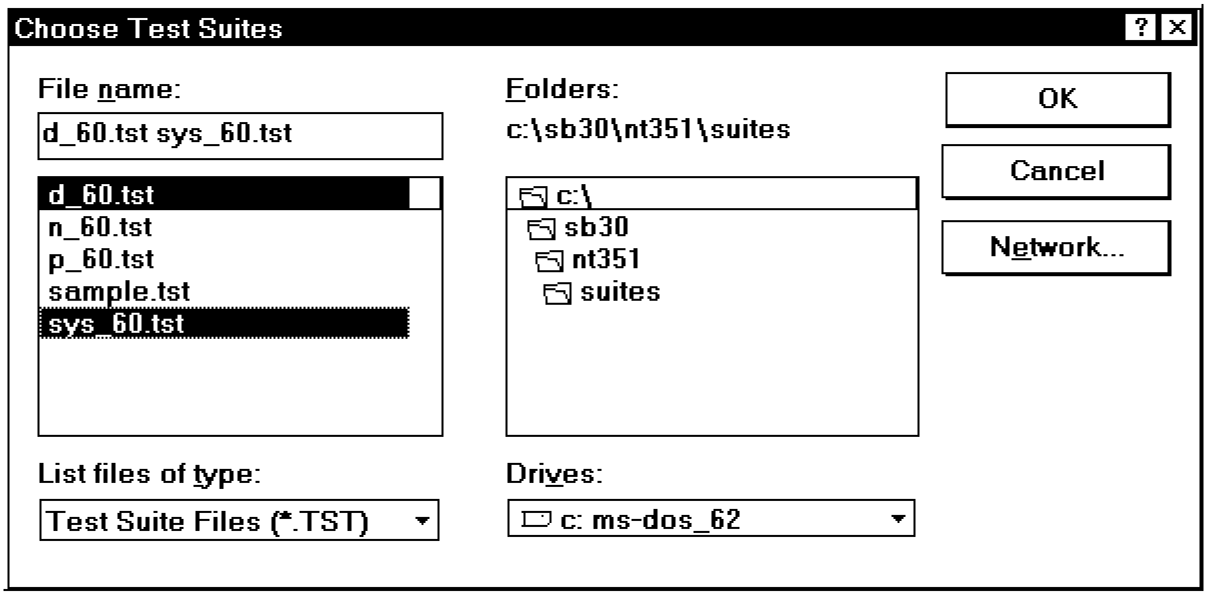
The test suite selection windows

Before you actually run a test suite, you must select it. ServerBench displays several windows to aid you in selecting the test suites you want to run.

When you tell ServerBench you want to choose a test suite to run, it displays the Choose Test Suite dialog box. Go to the directory that holds the test suite (or suites) that you want and highlight the names of the test suites you want to run. You can choose as many test suites from this window as you like. To select a range of test suites, click on the first suite and then, holding down the shift key, click on the last suite in the range. To randomly select test suites, click on the first test suite and then hold down the control (Ctrl) key each time you click on another test suite. Once you've selected the test suites you want, click on OK.

NOTE: When you select several tests, Windows arranges their names in the File name box in alphabetical order, not the order that you selected them.

Figure 5-11: The Choose Test Suites dialog box



File name: d_60.tst sys_60.tst

- d_60.tst
- n_60.tst
- p_60.tst
- sample.tst
- sys_60.tst

Folders: c:\sb30\nt351\suites

- c:\
- sb30
- nt351
- suites

List files of type: Test Suite Files (*.TST)

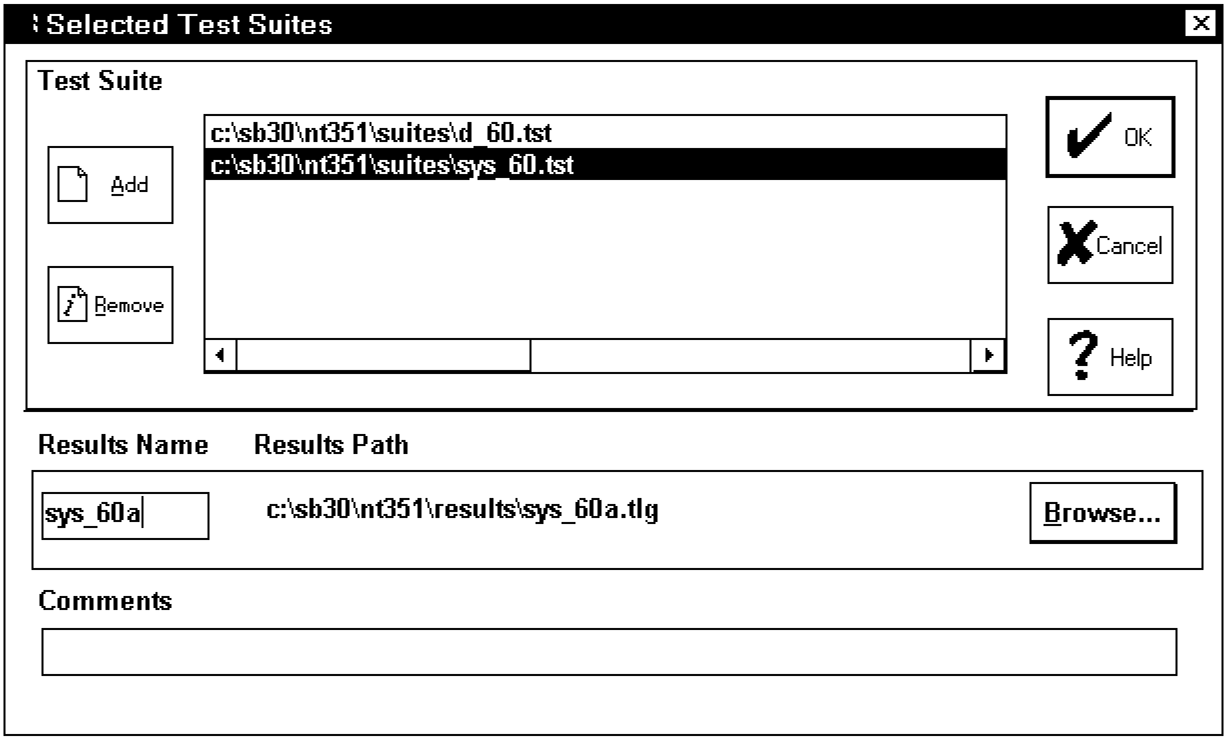
Drives: c: ms-dos_62

NOTE: You'll find ServerBench's standard test suite files in ServerBench's **SUITES** subdirectory in the controller's installation directory.

ServerBench takes you to the Selected Test Suites dialog box, which lists the suites you've already selected. Two special features in this window are that it lets you enter a path name for your results file and any comments you want to make about running this test suite. (If you don't enter a path name, ServerBench by default uses the test suite name with a **.TLG** extension. It places the path name in the **RESULTS** directory on the controller.)

This dialog box also provides you with an easy way to change the name of the results file so that ServerBench doesn't overwrite your previous results. You can type in a name for the results file in the Results Name box. If you want to change the path name to the results file, you can choose the Browse button. Each time you choose Browse, ServerBench displays the Select Results Path dialog box (see Figure 5-14). When you choose a path name in this box, ServerBench enters that path as the path name to your results file.

Figure 5-12: The Selected Test Suites dialog box



Selected Test Suites

Test Suite

c:\sb30\nt351\suites\d_60.tst
c:\sb30\nt351\suites\sys_60.tst

Results Name: sys_60a

Results Path: c:\sb30\nt351\results\sys_60a.tlg

Comments:

From this window you can also:

- **Add** more test suites to your current list of test suites. When you choose the Add button, ServerBench displays the Choose Test Suites dialog box. ServerBench always places the test suite (or suites) you're adding after the test suite that is currently highlighted.

- **Remove** one of the currently selected test suites. If you decide you don't want to run all the test suites you've chosen, you can use the Remove button to take the suites you don't want out of the list.

Once you have the suites listed that you want to include in this test run, click on OK.

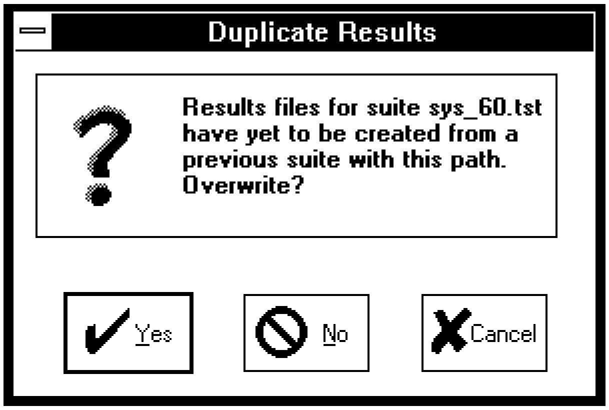
Related Topics:

Duplicate results names

Duplicate results names

ServerBench warns you if you've entered a name for your results file that another file is already using. For example you might see the warning that appears in Figure 5-13.

Figure 5-13: Duplicate Results warning box



Duplicate Results

Results files for suite sys_60.tst have yet to be created from a previous suite with this path. Overwrite?

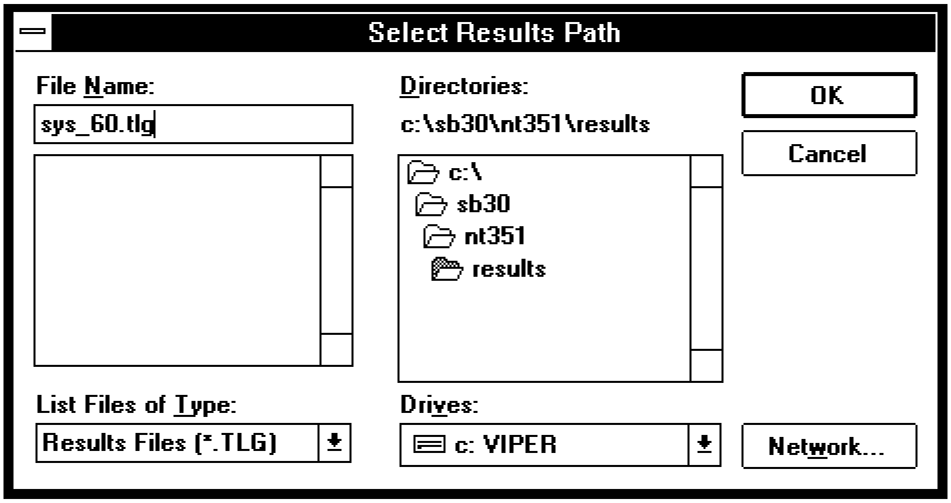
This warning means that you've selected two test suites (in this example, **sys_60.tst)** that have the same results path name. However, neither test suite has run yet. ServerBench also displays warnings if you enter the name for an existing results file.

Each time ServerBench displays a warning box, it gives you the option of choosing:

- Yes, which means you want to overwrite the first set of results with the second.
- No, which means you want to enter a new path name for the results file. (This is the same as choosing Browse in the Selected Suites dialog box).
- Cancel, which simply returns you to the Selected Suites dialog box.

When you choose NO, ServerBench displays the Select Results Path dialog box.

Figure 5-14: Select Results Path dialog box



Select Results Path

File Name: sys_60.tlg

Directories: c:\sb30\nt351\results

c:\
sb30
nt351
results

List Files of Type: Results Files (*.TLG)

Drives: c: VIPER

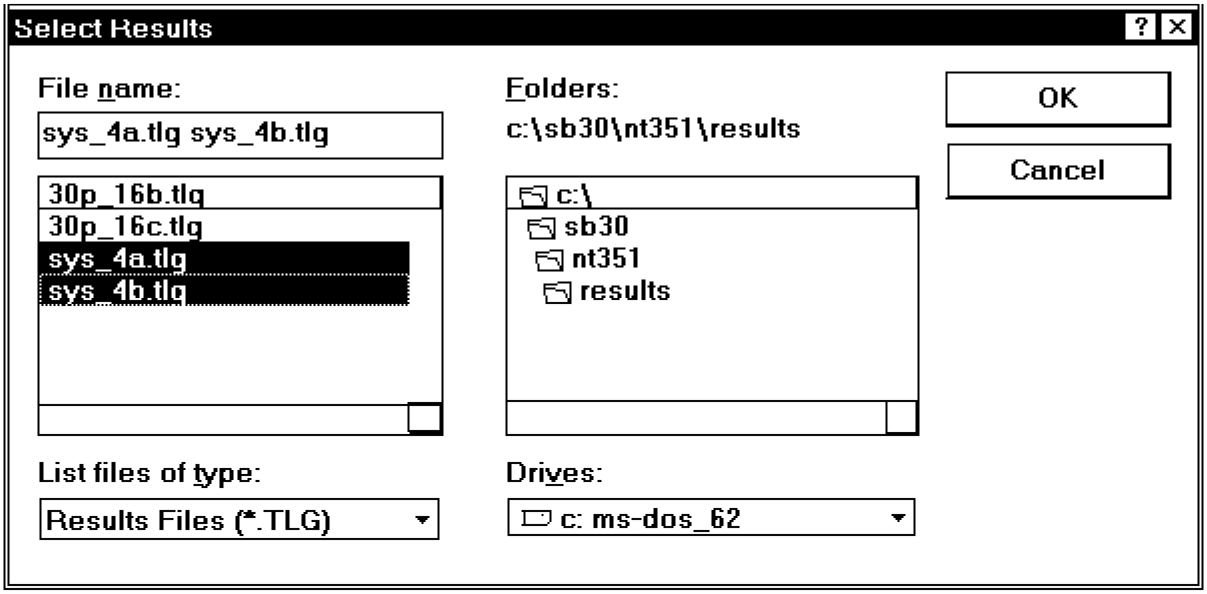
Under file name, ServerBench displays the current name of the results file. You can use this dialog box to specify a different directory for your results file. Because ServerBench uses sticky directories, it'll place the next results file you specify in this directory as well (unless you use the Browse button to specify a different directory.

The results dialog boxes

When you choose the View Results button from the main ServerBench window, ServerBench displays several dialog boxes.

The Select Results dialog box appears first. Go to the directory that holds your results and highlight the names of the results files you want to see. Each results file ends with a **.TLG** extension. You can choose as many results files from this window as you like. To select a range of files, click on the first file and then, holding down the shift key, click on the last file in the range. To randomly select results files, click on the first test suite and then hold down the control (Ctrl) key each time you click on another file. Once you've selected the files you want, click on OK.

Figure 5-15:   The Select Results dialog box



File name: sys_4a.tlg sys_4b.tlg

30p_16b.tlg
30p_16c.tlg
sys_4a.tlg
sys_4b.tlg

Folders: c:\sb30\nt351\results

c:\
sb30
nt351
results

List files of type: Results Files (*.TLG)

Drives: c: ms-dos_62

NOTE: By default, ServerBench places all the results files in the **RESULTS** subdirectory on the controller.

ServerBench now displays the Selected Results dialog box.

Related Topics:

The View Results dialog box

The View Results dialog box

You have lots of options in the View Results dialog box. For example, you can:

Work with your disclosure database. From this dialog box you can select and edit a disclosure database snapshot file.

Edit the disclosure logs ServerBench created and tell ServerBench to use them as the basis for the Sever and Client Disclosure tables in the Results workbooks.

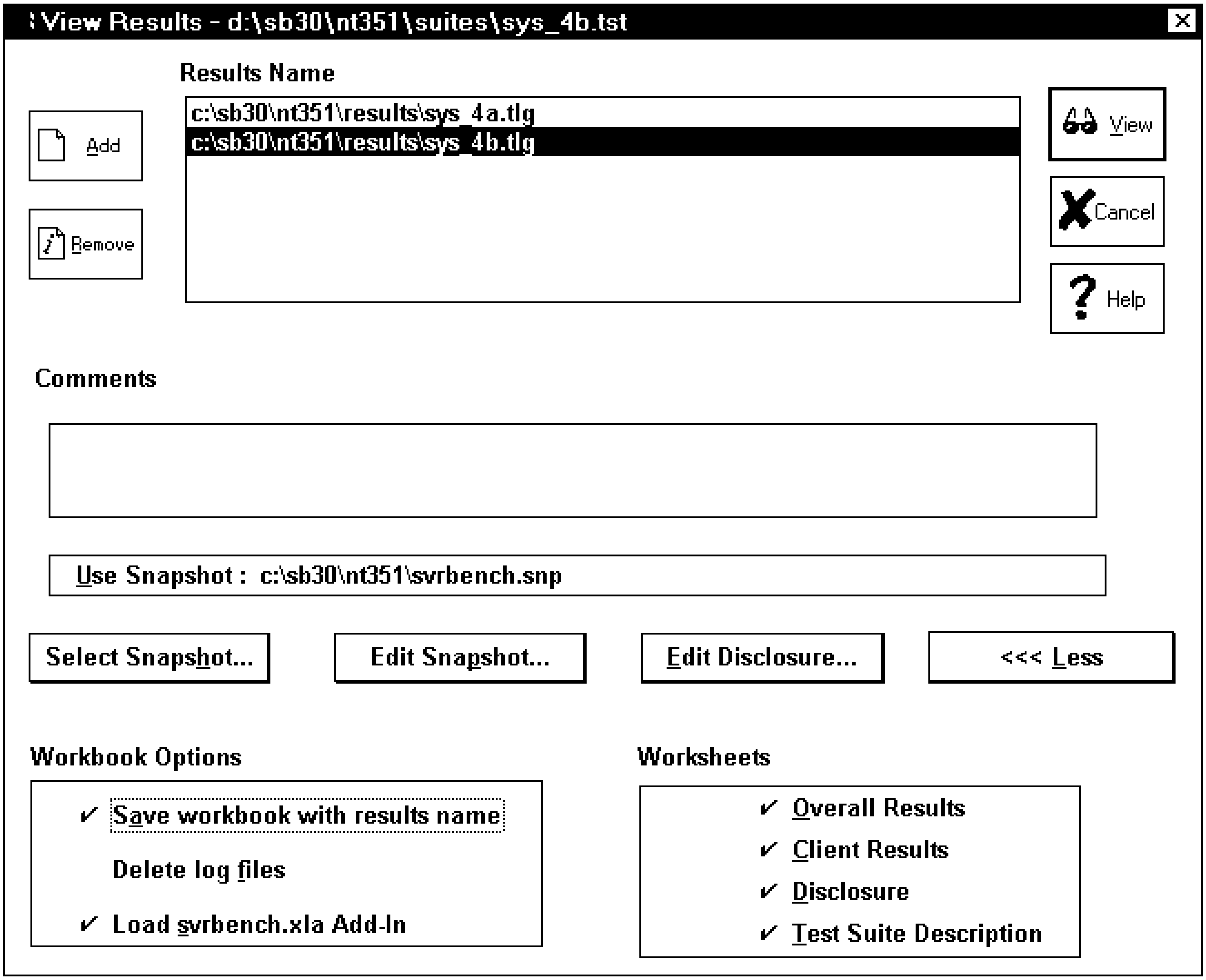
Select which results tables you want to look at. ServerBench will only create results tables for the categories you select.

Tip:

You can easily see which test suite you ran to get your results. Highlight a set of results in the Results Name box. ServerBench displays the name of the test suite associated with that set of results in the dialog box title bar.

Each option you select in this dialog box applies to all the results files listed under Results Name.

Figure 5-16: View Results dialog box



View Results - d:\sb30\nt351\suites\sys_4b.tst

Results Name

c:\sb30\nt351\results\sys_4a.tlg
c:\sb30\nt351\results\sys_4b.tlg

Comments

Use Snapshot : c:\sb30\nt351\svrbench.snp

Workbook Options

- ✓ Save workbook with results name
- Delete log files
- ✓ Load svrbench.xla Add-In

Worksheets

- ✓ Overall Results
- ✓ Client Results
- ✓ Disclosure
- ✓ Test Suite Description

Here's what you can do with the buttons in this dialog box:

- **Add** more results files to your current list of files. When you choose the Add button, ServerBench displays the Select Results dialog box.
- **Remove** one of the currently selected results files. If you decide you don't want to view all the results files you've chosen, you can use the Remove button to take the files you don't want out of the list.
- **Use Snapshot**, if checked, tells ServerBench to use the snapshot file displayed in this box as the basis for creating the Server Disclosure and Client Disclosure results tables. If you don't check this option, ServerBench uses the information in the **.DLG** logs to create those tables.
- **Select Snapshot** lets you choose a different database snapshot file. ServerBench uses the information in the database snapshot file to create the Server Disclosure and Client Disclosure results tables. You need to supply the information in these tables if you want to publish your results. You can create one snapshot file that contains all the information on your test setup and then use it for each set of results as long as your test setup stays the same.
- **Edit Snapshot** lets you modify the server and client disclosure information stored in the snapshot file specified by the Use Snapshot option.
- **Edit Disclosure** lets you modify the information that ServerBench captured about your server and

clients. When you run a test suite, ServerBench places information it captures about the server and the client in the **.DLG** log files that it creates for that set of results. ServerBench displays this information in the Server and Client Disclosure forms when you choose Edit Disclosure. If you also have the Use snapshot option checked, ServerBench includes the information from that snapshot file in these forms as well as the **.DLG** information. You can add the **.DLG** information to your current snapshot file by checking option Update snapshot file... at the bottom of the Server and Client Disclosure forms. If you don't check this option, ServerBench won't save the information on the disclosure forms to the snapshot file.

- **Less/More** lets you choose how much of the View Results dialog box ServerBench displays. By default, ServerBench only displays the top portion of this dialog box; it doesn't display the Workbook and Worksheet options. If you choose More, ServerBench displays the entire dialog box and the name on the button switches to Less. Then, when you choose Less, ServerBench displays the abbreviated dialog box.

In addition to setting up your disclosure information in this dialog box, you can also tell ServerBench how to set up your results workbooks. You can choose:

- Which workbook options you want. Under Workbook Options, ServerBench lets you specify:

  a. **Save Workbook with results name**. When checked, this option tells ServerBench to automatically save each set of a results to an Excel workbook that has the same name as the results file. For example, if your results are called **SYS_4A.TLG**, Excel would save them to a workbook called **SYS_4A.XLS**.

  b. **Delete Log files**. When you check this option, ServerBench will delete the results log files it uses to generate the results tables after it creates the results spreadsheets. Then, if you don't save the spreadsheet using Excel (or didn't tell ServerBench to automatically save the spreadsheet), you won't be able to view those results again. In addition, if you didn't tell ServerBench to create all the results tables for that set of results, you won't be able to generate them later.

  c. **Load svrbench.xla Add-in**. When you check this option, ServerBench automatically loads its **SVRBENCH.XLA** Add-In module for Excel. This module lets you create a TPS summary graph and a variance summary graph. It also enables you to easily print the graphs or tables from within the ServerBench workbooks without printing other forms of data. When you select this option, ServerBench adds a ServerBench option under Excel's Data menu.

Tip:

*While you can permanently install the **SVRBENCH.XLA** module through Excel itself, we recommend that you install this module from the View Results window. This way, it won't affect your daily Excel work. If you install the module permanently in Excel, you'll notice that your Excel start-up time is slower even when you're not generating ServerBench results.*

- Which results tables you want to view. When you check the category, ServerBench displays those tables when it creates the results workbooks.

Once you have all the results files you want to view and have set up database and workbook options, click on View and ServerBench will display your results.

NOTE: If you have a Excel session currently running, ServerBench will use that session of Excel to display the results workbooks. Otherwise, it will start a session of Excel.

The creating and editing test suites screens

When you choose the Create or Edit Test Suites button from the main ServerBench window, ServerBench displays a series of dialog boxes.

Related Topics:

Create or Edit Test Suites dialog box

The Mixes in Suite dialog box

Adding existing mixes to a test suite

The Reorder mixes dialog box

The Mix Definition window

Duplicating mix fields across a test suite

Copying mix information into the Mix Definition window

Create or Edit Test Suites dialog box

The first dialog box to appear is the Create or Edit Test Suites dialog box.

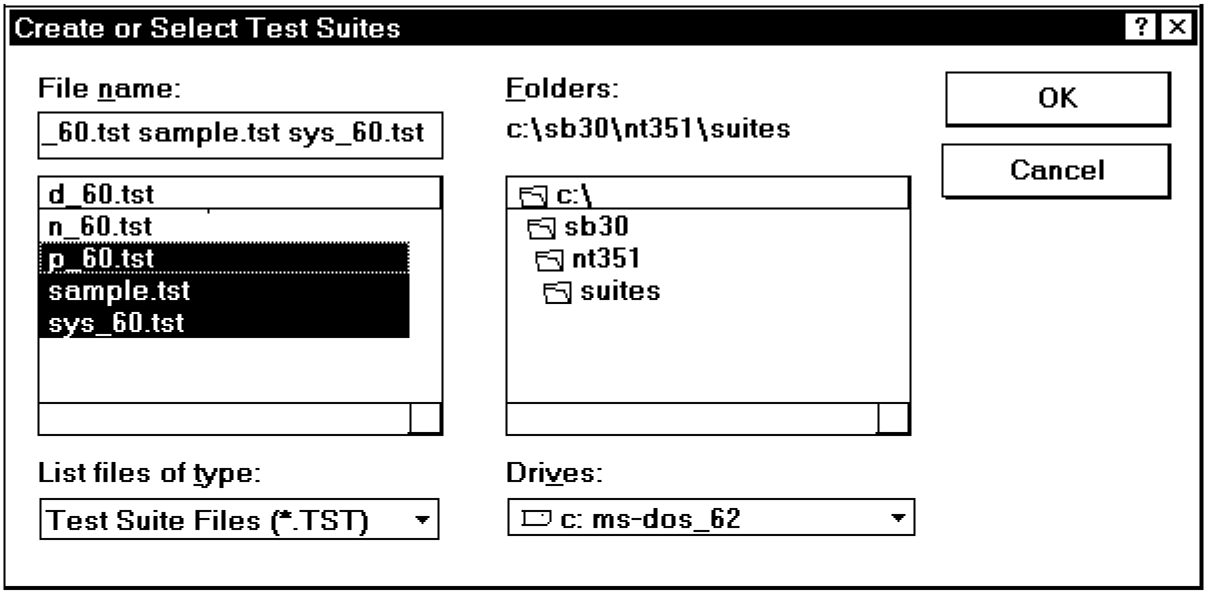
If you want to modify an existing test suite, go to the directory that holds the test suite (or suites) that you want and highlight the names of the test suites you want to edit. You can choose as many test suites from this window as you like. To select a range of test suites, click on the first suite and then, holding down the shift key, click on the last suite in the range. To randomly select test suites, click on the first test suite and then hold down the control (Ctrl) key each time you click on another test suite.

If you want to create a new test suite, enter the name of the test suite in the File name box. ServerBench will ask you if you want to create a test suite file.

Once you've selected the test suites you want, click on OK. ServerBench displays the Mixes in Suite dialog box.

NOTE: When you select several tests, Windows arranges their names in the File name box in alphabetical order, not the order that you selected them.

Figure 5-17: Create or Edit Test Suites dialog box



File name: _60.tst sample.tst sys_60.tst

d_60.tst
n_60.tst
p_60.tst
sample.tst
sys_60.tst

Folders: c:\sb30\nt351\suites

c:\
sb30
nt351
suites

List files of type: Test Suite Files (*.TST)

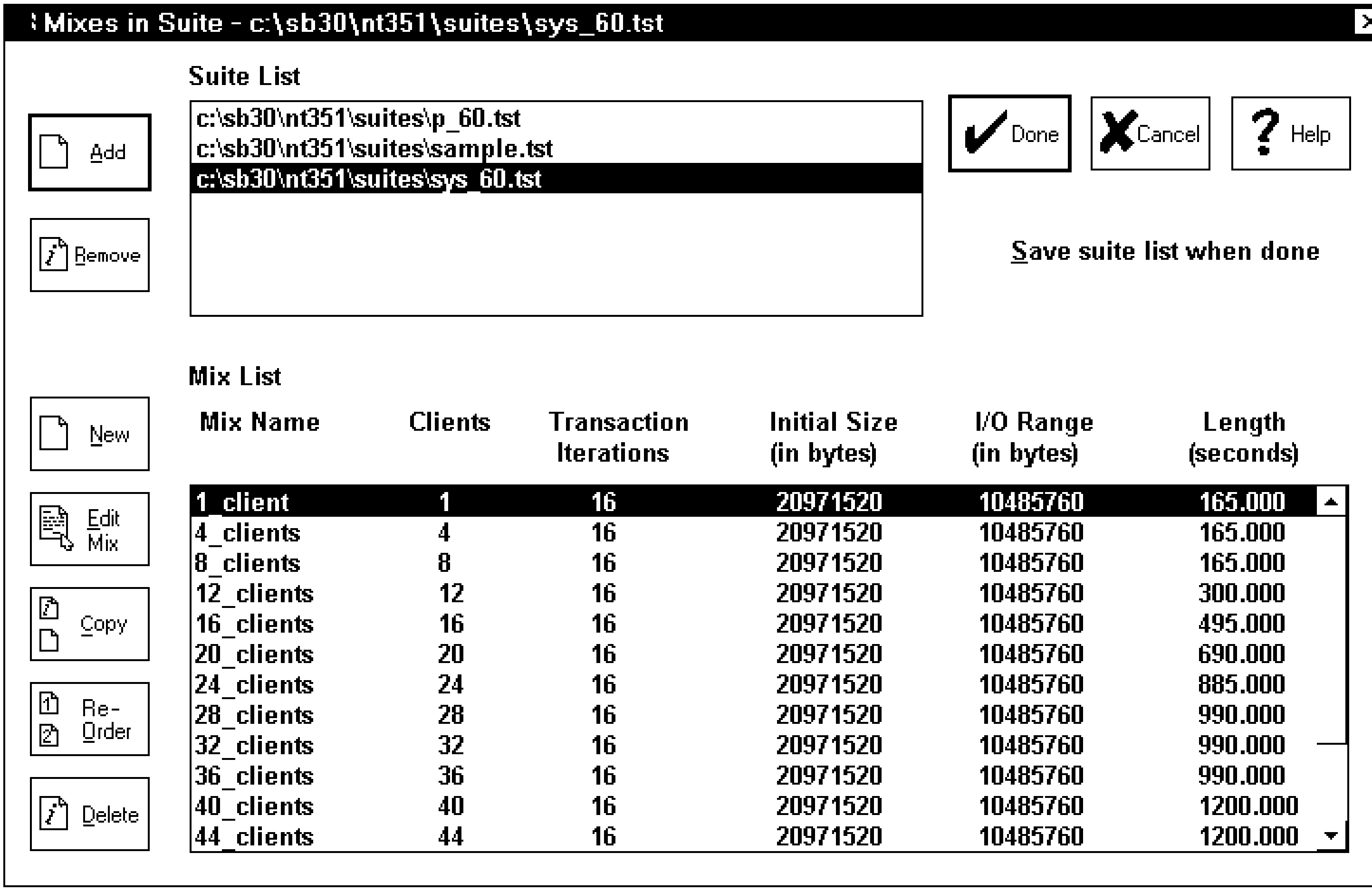
Drives: c: ms-dos_62

The Mixes in Suite dialog box

The Mixes in Suite dialog box offers you a number of options. It displays information on each mix in a test suite. It also lets you reorder the mixes and add mixes from other test suites. In addition, it lets you get to the Mix Definition window where you can create new mixes or modify existing mixes..

When you highlight a test suite in the Suite List, ServerBench displays all the mixes in that test suite in the Mix List. ServerBench tells you the values of some of the parameters that are set for that mix, such as the mix name, how many clients it has, how many transaction iterations the mix performs, the initial size of each client's data file, the value of the I/O range parameter, and the value of the Length parameter.

Figure 5-18: Mixes in Suite dialog box


Mixes in Suite - c:\sb30\nt351\suites\sys_60.tst
Add
Remove
Done
Cancel
Help
Save suite list when done
New
Edit Mix
Copy
Re-Order
Delete

Suite List

c:\sb30\nt351\suites\p_60.tst
c:\sb30\nt351\suites\sample.tst
c:\sb30\nt351\suites\sys_60.tst

Mix List

Mix Name	Clients	Transaction Iterations	Initial Size (in bytes)	I/O Range (in bytes)	Length (seconds)
1_client	1	16	20971520	10485760	165.000
4_clients	4	16	20971520	10485760	165.000
8_clients	8	16	20971520	10485760	165.000
12_clients	12	16	20971520	10485760	300.000
16_clients	16	16	20971520	10485760	495.000
20_clients	20	16	20971520	10485760	690.000
24_clients	24	16	20971520	10485760	885.000
28_clients	28	16	20971520	10485760	990.000
32_clients	32	16	20971520	10485760	990.000
36_clients	36	16	20971520	10485760	990.000
40_clients	40	16	20971520	10485760	1200.000
44_clients	44	16	20971520	10485760	1200.000

NOTE: If you are creating a new test suite file, the list of test mixes in this window will be empty.

This dialog box also includes numerous function buttons. Here's what you can do with the buttons in this dialog box:

- **Add** more test suites to this list. When you choose the Add button, ServerBench displays the Create or Edit Test Suites dialog box.
- **Remove** one of the currently selected test suites. If you decide you don't want to edit all the test suites you've chosen, you can use the Remove button to take the suites you don't want out of the list.
- **New** takes you to the Mix Definition window and lets you create a mix. ServerBench places this new mix after the currently highlighted mix in the test suite.

- **Edit Mix** displays the highlighted mix in the Mix Definition window. You can then change the values for that mix.
- **Copy** lets you copy one or more mixes from another test suite into the currently highlighted test suite. It inserts the mixes you're adding after the highlighted mix in the Mix List. When you choose the Copy button, ServerBench displays the Select Test Suites dialog box. Once you select a test suite and click on OK, ServerBench displays the Copy Mixes from... dialog box. Choose the mixes you want to copy by highlighting them. Now click on OK. ServerBench inserts the mixes you've highlighted into your current test suite. For more information, see the section "Copying mixes from an existing test suite" later in this chapter.
- **Reorder** lets you change the order of the mixes in the current test suite. When you choose the Reorder button, ServerBench displays the Reorder Suite Mixes dialog box. For more information, see the section "Reordering mixes" later in this chapter.
- **Delete** removes the mix you have highlighted from the test suite file.

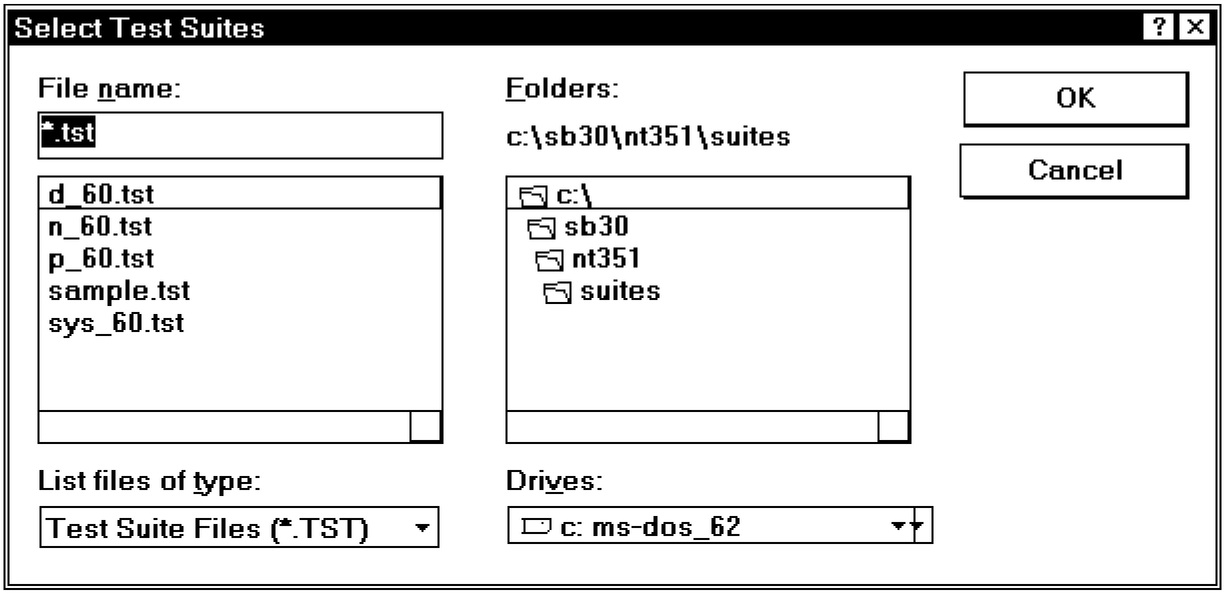
Adding existing mixes to a test suite

When you choose the Copy button from the Mixes in Suite dialog box, ServerBench lets you specify an existing test suite. It then displays all the mixes in that suite. You can highlight the mixes you want to add to your current test suite in the Mixes in Suite dialog box, and ServerBench will place those mixes after the current mix in the Mix List.

NOTE: If you just want to copy some of the information from one mix directly to the Mix Definition window for you to edit, use the Copy fields... option in the drop-down Advanced menu in the Mix Definition window. See the section "Copying mix information into the Mix Definition window" later in this chapter.

ServerBench displays the Select Test Suites dialog box when you choose the Copy button.

Figure 5-19: The Select Test Suites



File name: *.tst

d_60.tst
n_60.tst
p_60.tst
sample.tst
sys_60.tst

Folders: c:\sb30\nt351\suites

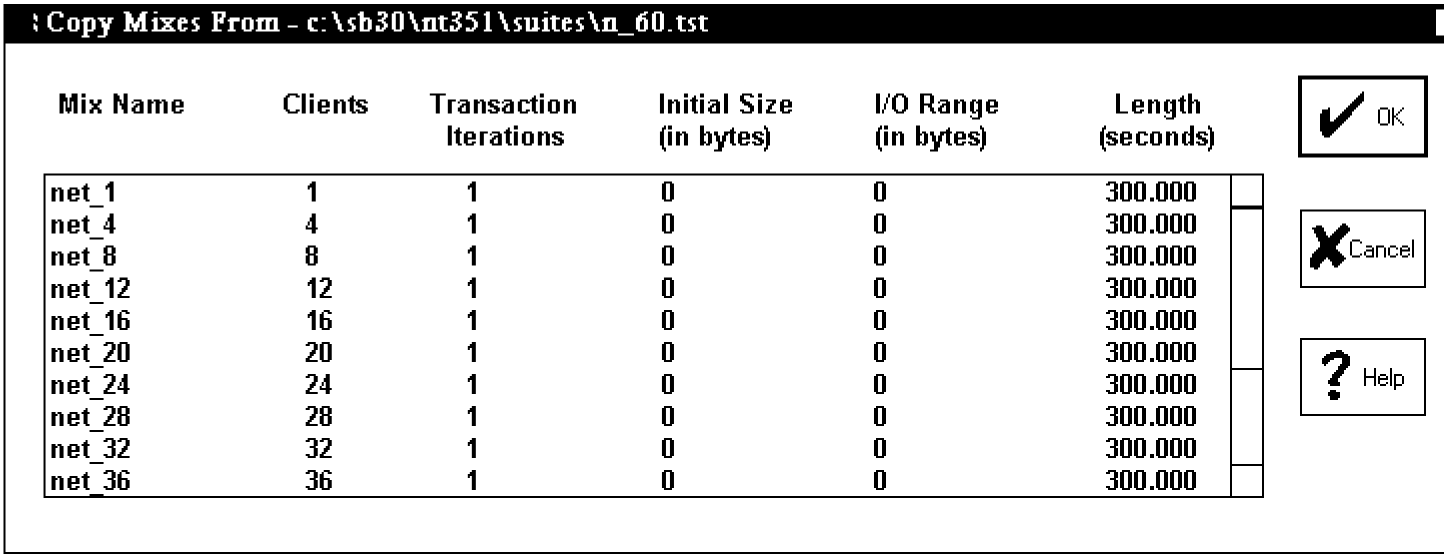
c:\
sb30
nt351
suites

List files of type: Test Suite Files (*.TST)

Drives: c: ms-dos_62

Go to the directory containing the test suite you want and select that suite. When you press OK, ServerBench displays the Copy Mixes From dialog box.

Figure 5-20 The Copy Mixes From... dialog box



Copy Mixes From - c:\sb30\nt351\suites\n_60.tst

Mix Name	Clients	Transaction Iterations	Initial Size (in bytes)	I/O Range (in bytes)	Length (seconds)
net_1	1	1	0	0	300.000
net_4	4	1	0	0	300.000
net_8	8	1	0	0	300.000
net_12	12	1	0	0	300.000
net_16	16	1	0	0	300.000
net_20	20	1	0	0	300.000
net_24	24	1	0	0	300.000
net_28	28	1	0	0	300.000
net_32	32	1	0	0	300.000
net_36	36	1	0	0	300.000

This dialog box displays information about the mixes in that test suite. It gives you the mix names, the number of clients in each mix, the number of transactions each mix performs, the value for the Disk test file initial size parameter, the value for the I/O range parameter, and the value of the Length parameter.

Highlight the mix or mixes you want to copy and then click on OK.

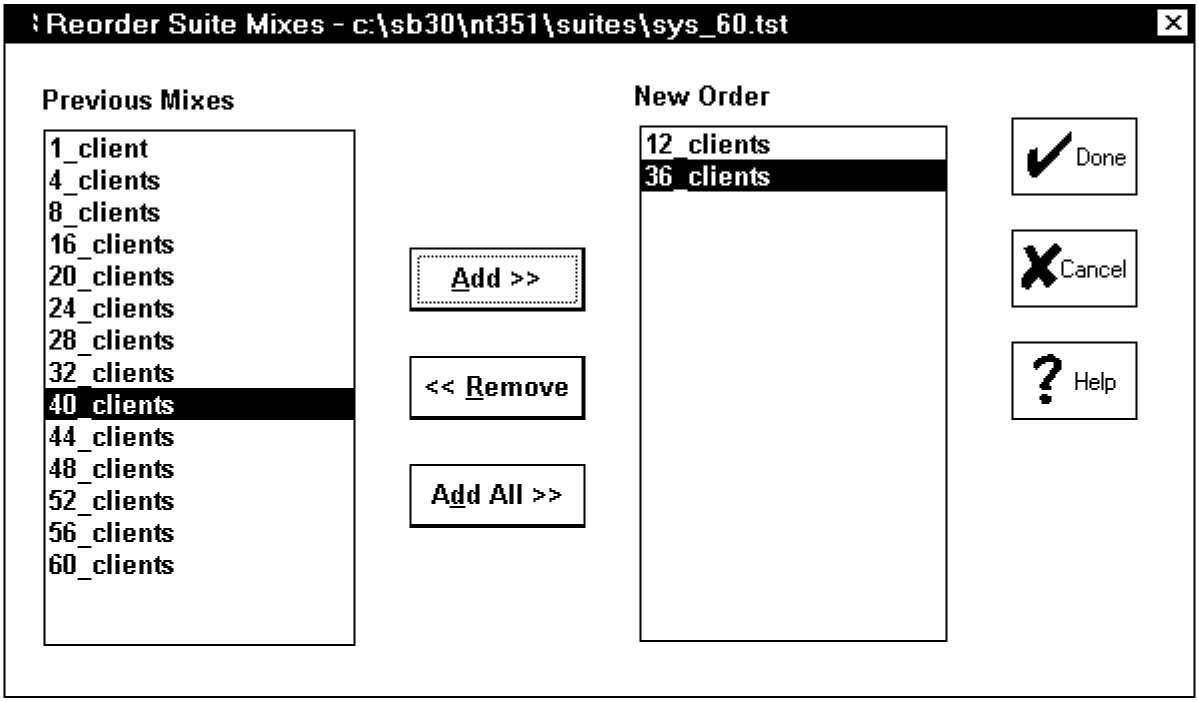
ServerBench returns you to the Mixes in Suite dialog box and places the mixes you've chosen in the Mix List after the highlighted mix.

The Reorder mixes dialog box

When you choose the Reorder button in the Mixes in Suite dialog box, ServerBench displays the Reorder Suite Mixes dialog box. This dialog box contains a list of all the mixes in the currently selected test suite. You can move these mixes one at a time or in groups to the New Order box. Highlight the mix (or mixes) you want to move and then click on the Add button. ServerBench inserts the mixes in the New Order box. If you already have some mixes in the New Order box, ServerBench places the ones you're adding after the highlighted mix. You can repeat this process as often as you need to.

To remove a mix from the New Order list, simply highlight it and then click on the Remove button. ServerBench places the mix back in the list of previous mixes.

Figure 5-21: Reorder Suite Mixes dialog box



Reorder Suite Mixes - c:\sb30\nt351\suites\sys_60.tst

Previous Mixes

1_client
4_clients
8_clients
16_clients
20_clients
24_clients
28_clients
32_clients
40_clients
44_clients
48_clients
52_clients
56_clients
60_clients

New Order

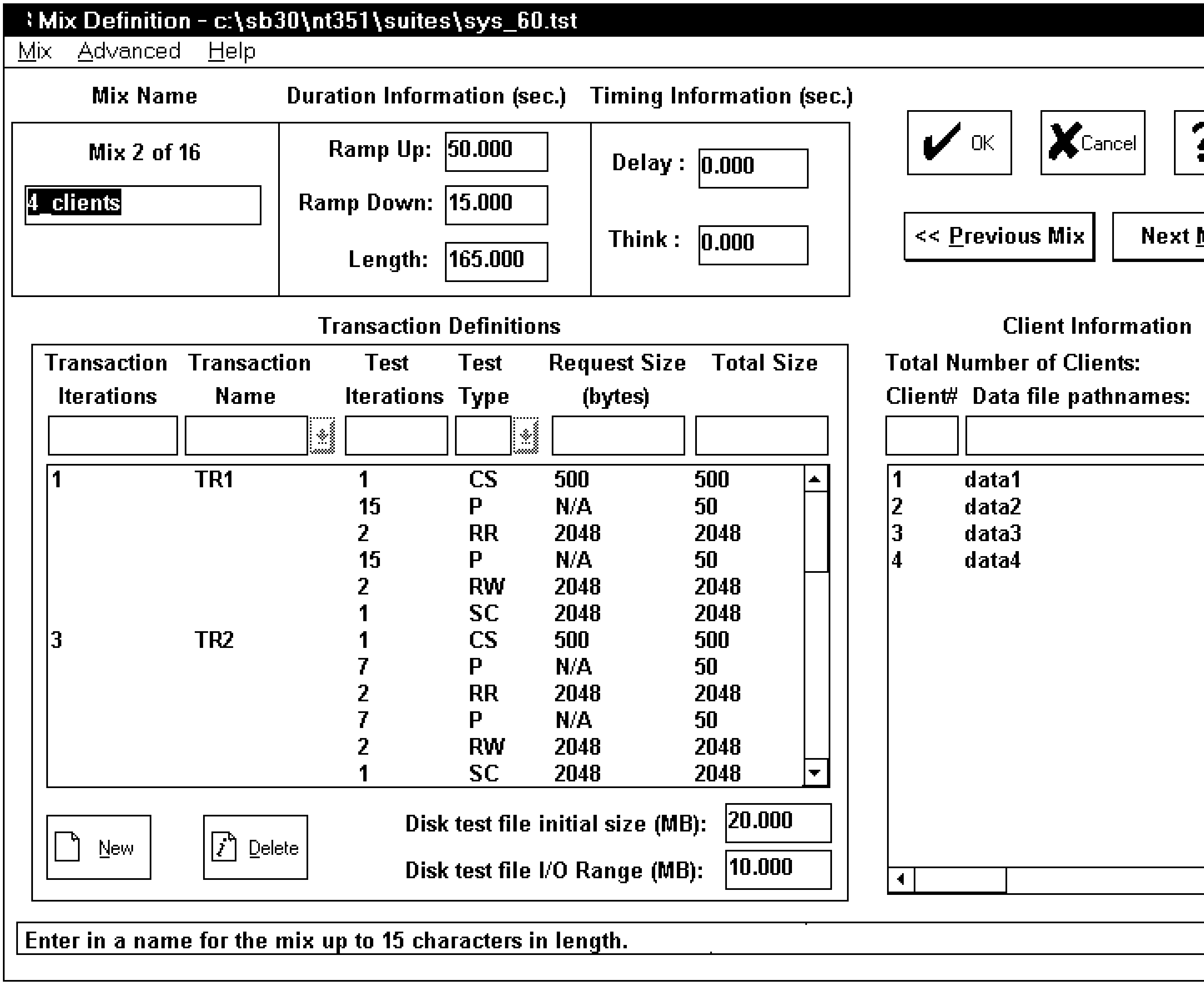
12_clients
36_clients

The Mix Definition window

You get to the Mix Definition window from the Mixes in Suite dialog box. To bring up the Mix Definition window, you can choose either the New button or the Edit Mix button. You can also get to the Mix Definition window by double-clicking on a test suite name in the Suite List or on a mix name in the Mix List.

From the Mix Definition window, you can create mixes or modify existing mixes. There are multiple parts to this window.

Figure 5-22: The Mix Definition window



Mix Name

Mix 2 of 16

4_clients

Duration Information (sec.)

Ramp Up: 50.000

Ramp Down: 15.000

Length: 165.000

Timing Information (sec.)

Delay : 0.000

Think : 0.000

Transaction Definitions

Transaction Iterations	Transaction Name	Test Iterations	Test Type	Request Size (bytes)	Total Size
1	TR1	1	CS	500	500
		15	P	N/A	50
		2	RR	2048	2048
		15	P	N/A	50
		2	RW	2048	2048
		1	SC	2048	2048
3	TR2	1	CS	500	500
		7	P	N/A	50
		2	RR	2048	2048
		7	P	N/A	50
		2	RW	2048	2048
		1	SC	2048	2048

Disk test file initial size (MB): 20.000

Disk test file I/O Range (MB): 10.000

Client Information

Total Number of Clients:

Client#	Data file pathnames:
1	data1
2	data2
3	data3
4	data4

Enter in a name for the mix up to 15 characters in length.

At this screen you enter the following mix information:

Tip:

We explain all these parameters in Chapter 10 "Creating Your Own Test Suites."

- **The Mix name**. This is the name ServerBench displays when it lists the mixes in a test suite. Enter an alphanumeric name that is up to 15 characters long.

- **Duration Information**. Define the following parameters:

  Ramp up. Enter the amount of time in seconds for the ramp up period. ServerBench ignores test results from mix iterations that start during the Ramp up period (which occurs at the beginning of a mix). This way you avoid having your test results skewed because the server load at the beginning of the test was very light.

  Ramp down. Enter the amount of time in seconds for the ramp down period. ServerBench ignores test results from mix iterations that start during the Ramp down period (which occurs at the end of a mix). This way you avoid having your test results skewed because the server load at the end of the test was very light.

  Length. Enter the total amount of time in seconds you want ServerBench to run. The value for length needs to be greater than the sum of the values for Ramp up and Ramp down and all the transaction iterations; otherwise, ServerBench will never record any results. In general, the value for Length is the minimum amount of time the test will run. If ServerBench is in the middle of a mix iteration when the Length value expires, ServerBench will continue the test until it finishes the current iteration. (One iteration of a mix means that each client executes each transaction in the mix.)

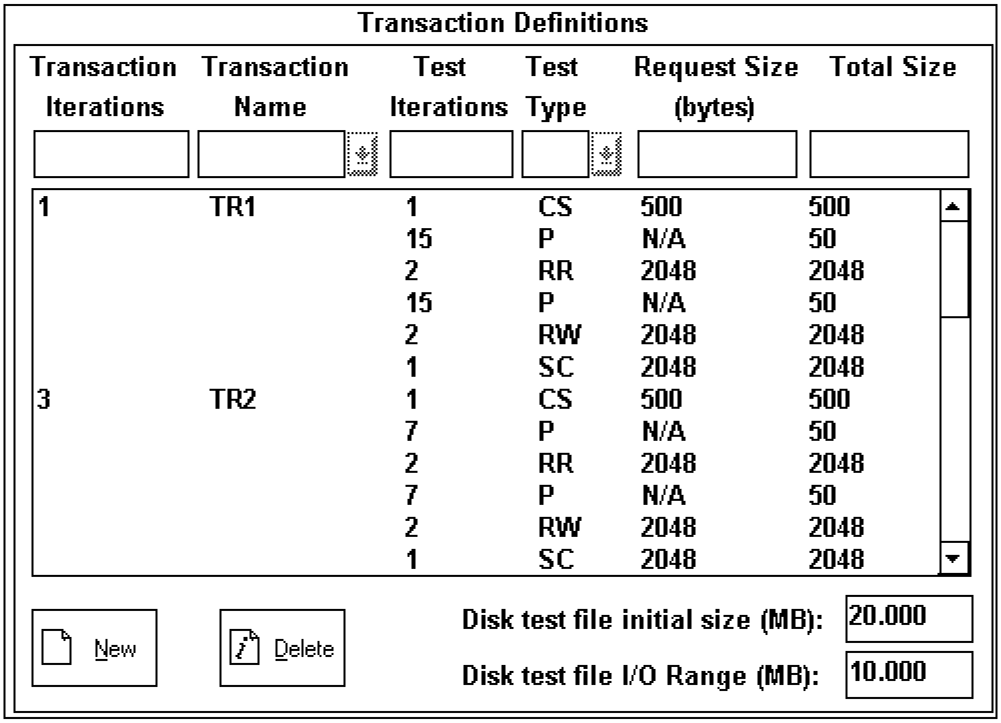
- **Timing Information**. Define the following parameters:

  Delay. Enter the maximum amount of time in seconds (or fractions of seconds) that you want a client to wait before starting a test once the controller tells the clients to start. When you set the Delay time parameter, each client takes that value and generates a pseudo-random number. This way the Delay time can vary for each client. As a result, when you set Delay time, you stagger the clients' initial requests to the server instead of overloading the server with a burst of requests at the beginning of each mix. The default value for Delay is 0, which is the value ServerBench's standard test suites use.

  Think time. Enter the number of seconds or fractions of a second a client will wait after receiving a transaction response before it sends another request. The default value for Think time is 0, which is the value ServerBench's standard test suites use. The smaller the value for Think time, the more the test stresses the server.

- **Transaction Definitions**. Define the following parameters:

Figure 5-23: The transaction portion of the Mix Definition window



Transaction Definitions

Transaction Iterations	Transaction Name	Test Iterations	Test Type	Request Size (bytes)	Total Size
1	TR1	1	CS	500	500
		15	P	N/A	50
		2	RR	2048	2048
		15	P	N/A	50
		2	RW	2048	2048
		1	SC	2048	2048
3	TR2	1	CS	500	500
		7	P	N/A	50
		2	RR	2048	2048
		7	P	N/A	50
		2	RW	2048	2048
		1	SC	2048	2048

Disk test file initial size (MB): 20.000

Disk test file I/O Range (MB): 10.000

Transaction iterations. Enter the number of times you want ServerBench to execute that transaction during one iteration of the test mix.

Transaction Name. Enter the name of the transaction in the box under this heading.

Test Iterations. Enter the number of times you want ServerBench to repeat the test during the transaction.

Test Type. Enter the mnemonic for the test you want to include in this transaction in the box under this heading. You can see a list of mnemonics by choosing the down arrow next to this box.

Request size. For the disk and network tests, you must enter the amount of data in bytes that you want ServerBench to move as a unit at one time.

Total size. For the disk and network tests, enter the total amount of data in bytes that you want ServerBench to move during that test. For the processor, enter the total number of iterations you want that test to perform.

Disk Test file initial size (MB). Enter the size you want each client's Disk test data file to be when ServerBench creates it on the server.

Disk Test file I/O Range (MB). Enter the size that you want ServerBench to use for the I/O range. This value specifies how much of the disk file ServerBench will use when performing disk operations.

New. Use the New button to create a transaction or to add another test to the current transaction. If you want to add a test to an existing transaction, you'll need to tab over to the Test Iterations field. If you enter any information in the Transaction Iteration or Transaction Name fields, ServerBench will create a new transaction.

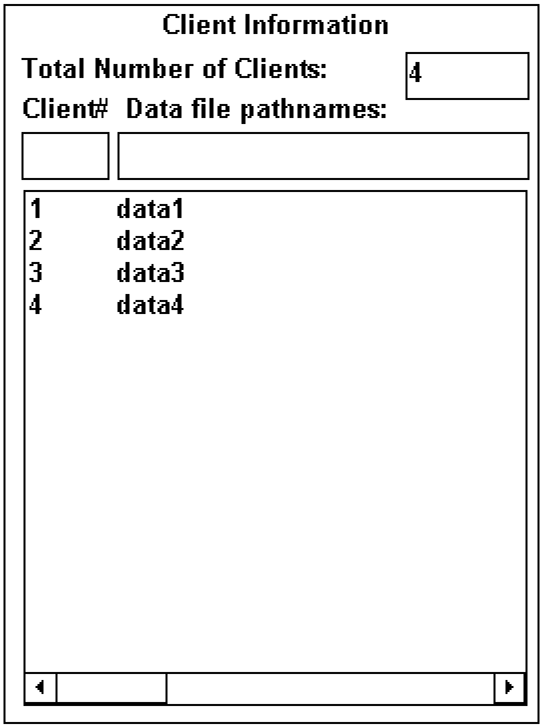
Delete. Use the Delete button to remove a highlighted test.

- **Client Information**. Define the following parameters:

  Total number of clients. Enter the number of clients that you want to participate in this test mix.

  Client # and Data file path names. ServerBench uses the data file path name to determine where to create the test data files for the disk read and write tests. You can use these boxes to edit the data file path name. If you enter a path name that ends in an asterisk (*), ServerBench will automatically append a number to that path name and enter it as the path name for each client up to the maximum number of clients. So, if you specified 60 clients, and you entered the path name **test***, ServerBench would create the path names **test1**, **test2**... up to **test60**.

Figure 5-24: The client portion of the Mix Definition window



Client Information

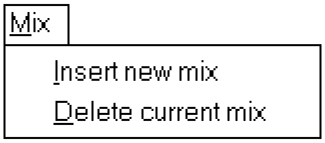
Total Number of Clients: 4

Client#	Data file pathnames:
1	data1
2	data2
3	data3
4	data4

A special feature of the Mix Definition window is that you can scroll through all the mixes in the test suite. In the Mix Name section of this window, ServerBench tells you which number this mix is in the test suite out of the total number of mixes. To move between the mixes, just choose the Previous or Next buttons.

The Mix Definition window also provides several drop-down menus that you can use in working with mixes. These menus are:

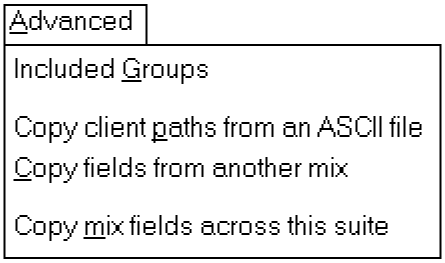
- **Mix.** From drop-down menu you can:



Insert new mix. This option lets you create a new mix for this test suite. The new mix always comes after the currently displayed mix.

Delete current mix. This removes the highlighted mix from the test suite.

- **Advanced**. From drop-down menu you can:


Advanced
Included Groups
Copy client paths from an ASCII file
Copy fields from another mix
Copy mix fields across this suite

Included Groups. When you choose this button, a pop-up dialog box appears that lets you select the group numbers you want included in this test mix. You can use the buttons in this dialog box to select all the clients or to clear all the group specifications you currently have entered.

Copy paths from an ASCII file. With this option, the Copy Paths From A File dialog box appears. At the dialog box, enter the name of the ASCII file where you list the path names for your clients. When you create this file, you should give it a **.PTH** extension. In the file, list the path name for each client on a separate line. For example, depending on the delimiters your operating system uses, you might enter something like:

```
C:\CLIENTS\CLIENT1 C:\CLIENTS\CLIENT2 C:\CLIENTS\CLIENT3 ... C:\
CLIENTS\CLIENT60
```

When you specify a directory in the path name, make sure that directory exists before you execute the test.

Copy fields from another mix. You can use this option to copy all or part of the information from an existing mix into the Mix Definition window. This option first displays the Copy Test Suite File dialog box and then the Copy a Mix dialog box. For more information on using this option, see the section "Copying mix information between test suites" later in this chapter.

Copy mix fields across this suite. When you choose this option, ServerBench displays the Duplicate Mix Fields Across Suite dialog box. For more information, see the section "Duplicating mix fields across a test suite" later in this chapter.

Duplicating mix fields across a test suite

The drop-down Advanced menu in the Mix Definition window contains the option Copy mix fields across this suite. When you choose this option, ServerBench displays the Duplicate Mix Fields Across Suite dialog box. This dialog box lets you copy the parameter values in the current mix to all the mixes in that test suite. You can copy the value for one parameter or multiple parameters.

The options you can choose at this dialog box you are:

- **Duration information.** You can specify the Ramp up, Ramp down, and/or Length parameters.
- **Timing information.** You can choose the Think and/or Delay parameters.
- **Transaction definitions.** If you choose the Transaction Definitions option, ServerBench replaces the transactions in the other mixes with the transactions in the current mix. However, it does not affect the client information in the mixes. Make sure you want all the transactions to be the same in all the mixes before you choose this option.

  If you choose the Disk test file initial size and/or the Disk test I/O range, ServerBench updates all the mixes so that they have the same value as the parameter (or parameters) you've chosen.
- **Client information.** If you choose All information, ServerBench makes the client information in all the mixes identical. You will now have the same number of clients in each mix with the same path names. Suppose you have three mixes in a test suite. Mix1 has 8 clients, Mix2 has 16 clients, and Mix3 has 32 clients. You're currently editing Mix2. If you choose All information, then each mix in this test suite with have 16 clients. All the path names and group designations in the three mixes will match the path names and group designations you specified in Mix2.

  To change only one part of the client information without affecting the number of clients in the mix, choose either Pathnames or Groups. If you choose the Pathnames option, ServerBench will change only the client path names in the other mixes and only for as many clients as the current mix has. In other words, if the current mix is Mix2 with 16 clients, ServerBench would only change the first 16 path names in Mix3. The path names for clients 17 through 32 would remain untouched.

Related Topics:

A word of caution about duplicating mix fields

A word of caution about duplicating mix fields

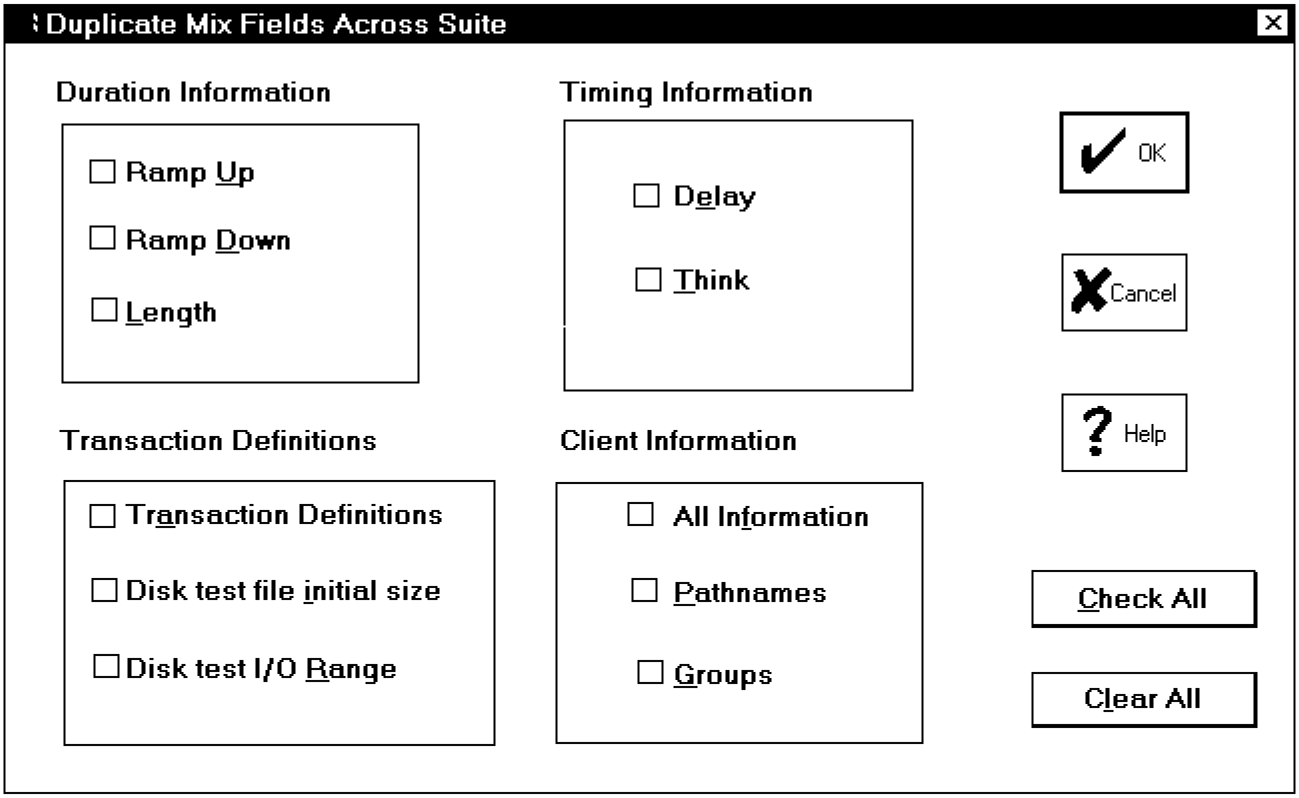
ServerBench's Duplicate Mix Fields Across Suite feature can be very handy, but you need to keep in mind how the settings you copy can affect the other mixes in the test suites. Here're some things to watch out for:

- **Null pathnames and blank Disk test parameters.** If you're copying information from a Processor test or a Network test to mixes that include Disk tests, make sure you don't copy the client path names, the value for the Disk Test File Initial Size parameter, or the value for the Disk test I/O Range parameter. Because the Processor and Network tests don't use those parameters, you'd be copying null pathnames and blank values for the two Disk test parameters. As a result, if you copy this information to a mix containing Disk tests, you'll be creating an invalid mix.

  The opposite is also true. If you choose only the option Transaction Definitions when you're at a mix containing Disk tests and you're copying this information to mixes containing Processor or Network tests, you'll again be creating invalid mixes. This is because ServerBench will add the transaction information (i.e., the Disk tests) to the mix containing the Processor and Network tests, but it won't copy any of the client information. So, once more, you'll have disk tests with no client path names or values for either the Disk Test File Initial Size parameter or the Disk test I/O Range parameter.

- **Dependent fields.** If you're only changing one parameter, you need to consider how that parameter works with other parameters. It's possible that, if you globally update only one parameter across all the mixes, you may cause an error condition for another mix. For example, suppose you change one field, such as Length, without changing the other fields that work with Length − Ramp up and Ramp down. Your current mix (Mix 1) uses Ramp up and Ramp down periods of 30 seconds and a Length of 90 seconds. Mix2 uses Ramp Up and Ramp Down values of 1 minute each. If you tell ServerBench to change the Length value for the mixes to match the value for Mix1, you'll create an error condition for Mix2 because the Length of the test will in Mix2 no longer exceed the combined values of Ramp Up and Ramp Down.

Figure 5-25: Duplicate Mix Fields Across Suite dialog box



Duration Information

- □ Ramp Up
- □ Ramp Down
- □ Length

Timing Information

- □ Delay
- □ Think

Transaction Definitions

- □ Transaction Definitions
- □ Disk test file initial size
- □ Disk test I/O Range

Client Information

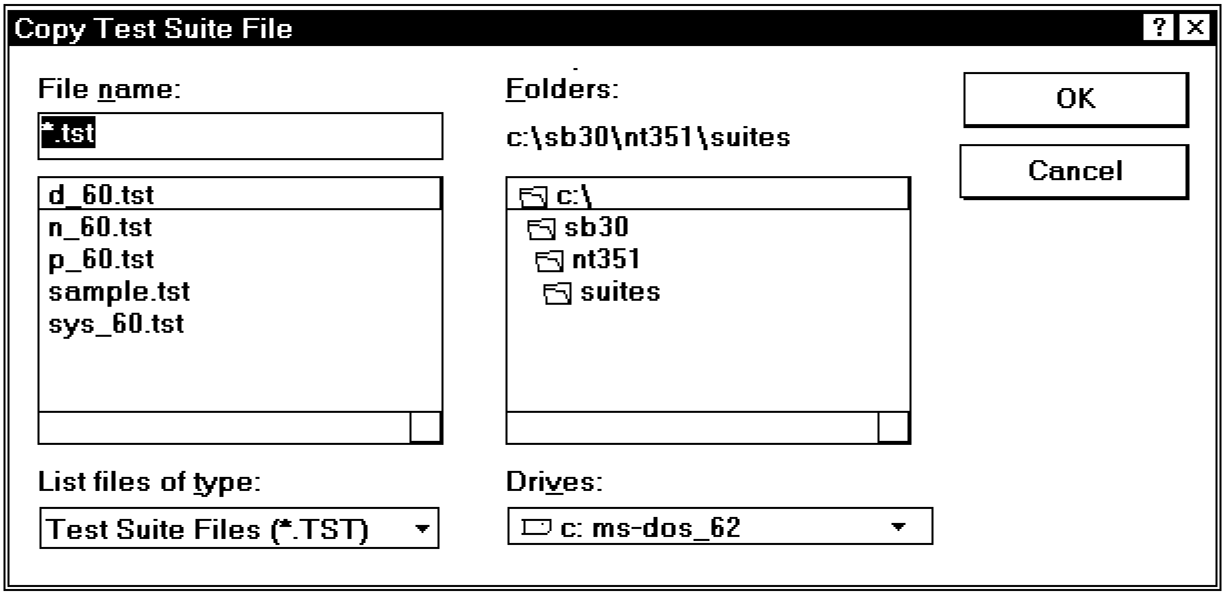
- □ All Information
- □ Pathnames
- □ Groups

Copying mix information into the Mix Definition window

You can use the Copy fields from another mix option in the Advanced menu to replace the current values in the Mix Definition window with the values in an existing mix. When you choose this option, ServerBench lets you copy either one entire mix or the values you specify.

After you select the Copy fields from another mix option, ServerBench displays the Copy Test Suite File dialog box. Highlight the test suite you want to use. You can only select one test suite.

Figure 5-26: Copy Test Suite File dialog box



File name: *.tst

d_60.tst
n_60.tst
p_60.tst
sample.tst
sys_60.tst

Folders: c:\sb30\nt351\suites

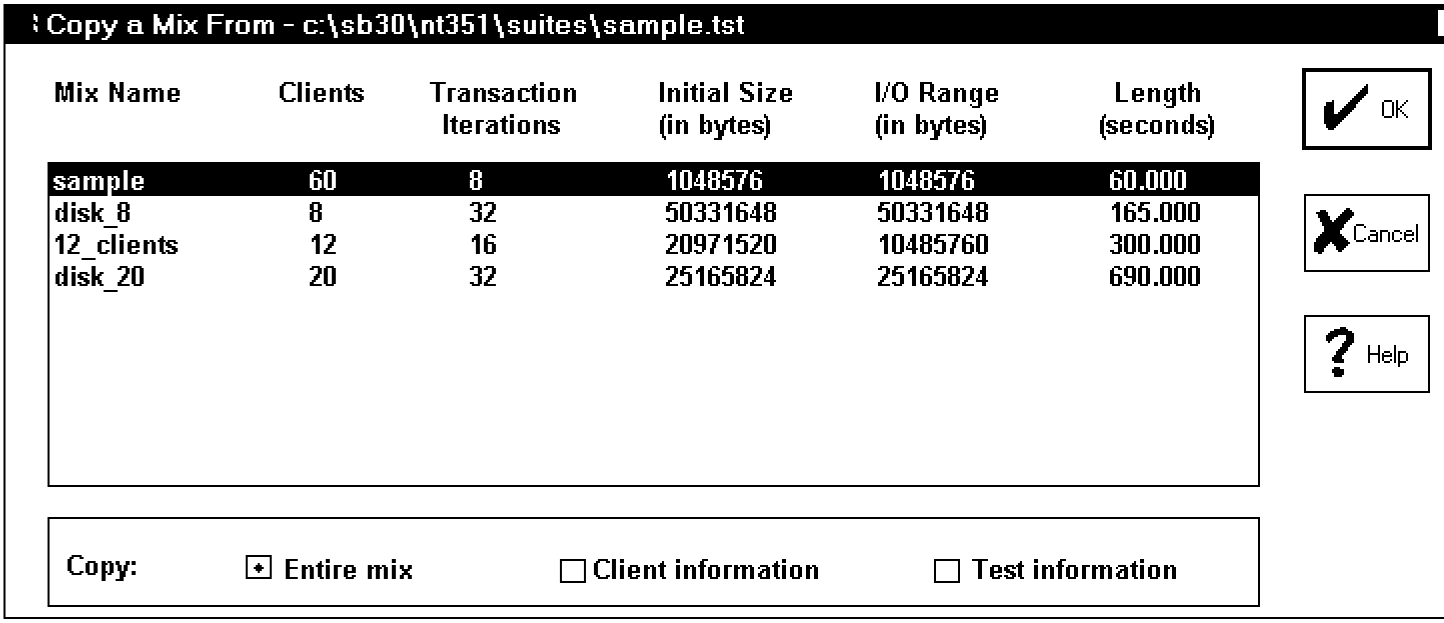
c:\
sb30
nt351
suites

List files of type: Test Suite Files (*.TST)

Drives: c: ms-dos_62

Once you click on OK, the Copy a Mix from dialog box appears.

Figure 5-27: Copy a Mix from... dialog box



Mix Name	Clients	Transaction Iterations	Initial Size (in bytes)	I/O Range (in bytes)	Length (seconds)
sample	60	8	1048576	1048576	60.000
disk_8	8	32	50331648	50331648	165.000
12_clients	12	16	20971520	10485760	300.000
disk_20	20	32	25165824	25165824	690.000

Copy: ☒ Entire mix □ Client information □ Test information

This dialog box displays information about the mixes in that test suite. It gives you the mix names, the number of clients in each mix, the number of transactions each mix performs, the value for the Disk test file initial size parameter, the value for the I/O range parameter, and the value of the Length parameter.

You now have three options

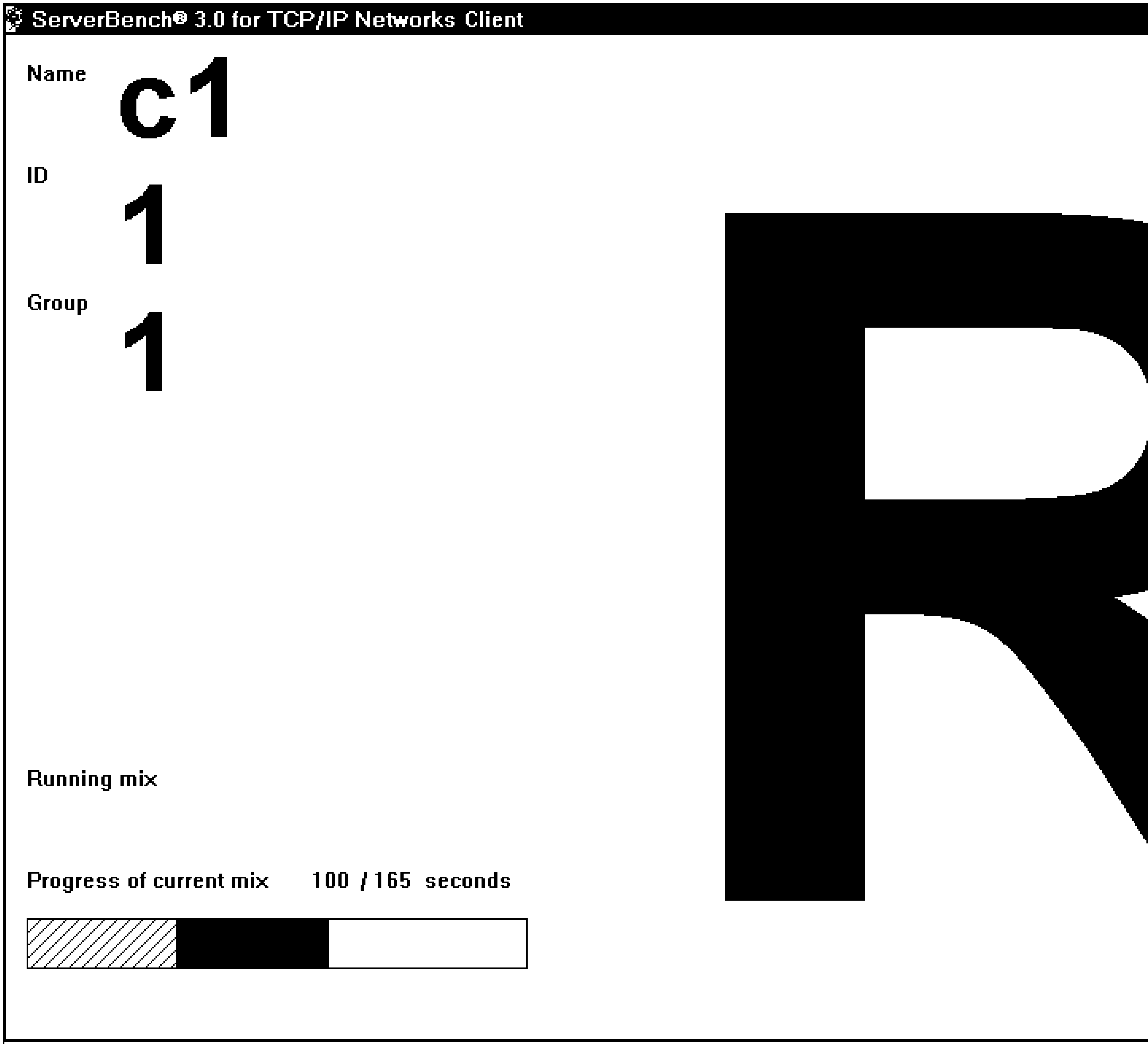
- **Entire mix**. ServerBench copies all the values in the selected mix to the Mix Definition window.
- **Client information**. ServerBench copies only the client information to the Mix Definition window.
- **Test information**. ServerBench copies only the transaction definitions to the Mix Definition window.

NOTE: If you want to copy multiple mixes at one time, use the Copy mixes button on the Mixes in Suite dialog box. (See the section "Adding existing mixes to a test suite" earlier in this chapter.)

The Client window

ServerBench also displays a Client window on each client. You can use the information in this screen to monitor the state of each client that is running a test.

Figure 5-28: The Client window



Name c1

ID 1

Group 1

Running mix

Progress of current mix 100 / 165 seconds

The key feature of the Client window is a very large letter indicating the client's state. The purpose of the letter is to let you see at a glance the stage the client is in. This ability can be very handy when you're running a large number of clients. For example, when the client is running a test mix, ServerBench displays the letter "R" in the Client window. The client states are:

B Blocked. ServerBench blocks a client between states to prevent one client from advancing to the next stage before the other clients are ready. For example, if Client 1 completes the test first, ServerBench will block Client 1 from sending its results until all clients have completed the test.

I Initializing. The client is initializing the current test mix.

R Running. The client is running the current mix.

S Sending. The client is sending its log to server.

D Done. The client has finished sending its log to server.

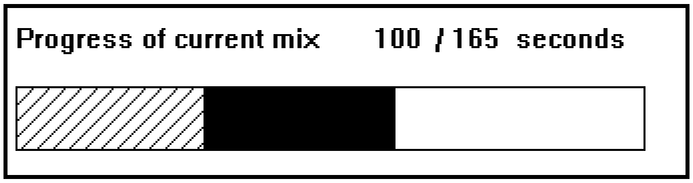
X Excluded. The client is excluded from the mix.

E Error. An error has occurred.

This screen also displays the client's identification number and group number. At the bottom of the screen, the client displays its current status or any error messages.

In addition, the Client window includes a bar graph that shows the progress of the current mix. When the bar is full, the mix is complete. ServerBench uses hash marks in the bar to let you know when the client is executing transactions that it is not counting (i.e., transactions that start during the Ramp up or Ramp down period). When the client is recording transactions, ServerBench fills the bar with a solid black color. For example, in Figure 5-29, the current mix has been running for 100 seconds and is scheduled to run at least 65 seconds more. Almost one-third shows hash marks, indicating that those transactions that have occurred so far began during Ramp up, so the client did not record them. The solid black color tells you that the client is now recording the transactions that complete.

Figure 5-29: Sample progress bar that appears in the Client window



Progress of current mix 100 / 165 seconds

ServerBench displays the Client window on the monitor of each client that is running the benchmark.

End of chapter

Overview

This chapter tells you how to get in touch with ZDBOp. You may want to get in touch with ZDBOp because you have questions or comments about ServerBench, you've encountered an unexpected problem with ServerBench, or you want to know how to get a copy of it or the other benchmarks Ziff-Davis provides.

If a problem occurs, look here for possible solutions

If a problem occurs while you're running ServerBench, here're some places to look for solutions:

- Check the on-line ServerBench manual for your server operating system. It contains some troubleshooting tips as well as any error messages you might see.
- Look in the **README.TXT** file for your ServerBench platform. This file contains any late-breaking information that did not make it into the manual. You'll find a copy of this file in the controller installation directory and on the CD-ROM in the directory that contains ServerBench for your operating system.
- Check the FAQ file for ServerBench, which we post on ZD Net™/CompuServe® Edition. (ZD Net is a Ziff-Davis on-line service. Access to CompuServe is available for a fee.)
- Read the messages in the ZDBENCH forum on ZD Net/CompuServe Edition to see if anyone else has had a similar problem.

Here's how you get in touch with ZDBOp

If you have technical questions about ServerBench that you can't find answers to in any of the documentation, fill out the Problem Report that is at the back of this manual and send it to us. If you have an Internet e-mail address, please include it when you get in touch with us. Here're some ways to get in touch with ZDBOp:

- If you have a modem and communications software, you can reach ZDBOp via its forum (GO ZDBENCH) on ZD Net/CompuServe Edition.

  The address for our World Wide Web page is:

  `http://www.zdnet.com/zdbop`

  You can also send an on-line version of the Problem Report to us via e-mail to:

  `zdbopwebmaster@zd.com`

- You can fax the Problem Report and any questions and comments you have directly to ServerBench Technical Support at the ZDBOp fax number 919-380-2879.
- You can mail the Problem Report and any questions and comments you have to ZDBOp at the following address:

  Ziff-Davis Benchmark Operation 1001 Aviation Parkway, Suite 400 Morrisville, NC 27560

  Attn: ServerBench Technical Support

Related Topics:

Getting copies of benchmarks from ZD Net

Requesting the benchmark CD-ROMs from ZDBOp

Getting copies of benchmarks from ZD Net

You'll find copies of ServerBench and other Ziff-Davis benchmarks on ZD Net/CompuServe Edition and ZD Net/World Wide Web Edition.

NOTE: You'll need to pay your usual connection fees when you download the benchmarks.

When you download ServerBench, you'll need to download three files for your server platform − one each for the server program, the controller program, and the client program. We provide a different set of files for each server platform ServerBench runs on.

To identify these files, we use the following naming scheme:

<*port-name*>**s**.<**exe** or **tar**> for the server file.

<*port-name*>**co**.**exe** for the controller file.

<*port-name*>**cl**.**exe f**or the client file.

where *port-name* is an abbreviated name we use to indicate that server operating system for that platform of ServerBench.

Requesting the benchmark CD-ROMs from ZDBOp

We distribute our server and PC benchmarks free of charge on CD-ROMs. The Server Benchmarks CD-ROM contains ServerBench and NetBench® (Ziff-Davis' industry-standard file server benchmark program). The Ziff-Davis PC Benchmarks CD-ROM contains WinBench® 96 , and Winstone® 96. You can also get a copy of the Ziff-Davis Macintosh Benchmark CD-ROM, which contains MacBench® 3.0 (a Mac OS system benchmark program). To get a benchmark CD-ROM, *mail* or *fax* the Benchmark Request form at the back of this manual to ZDBOp.

End of chapter

Problem Report Form for ServerBench

Information about you:

Name:

____

Address:

____

Company:

____

Phone or Fax: ______________________E-Mail Address____________________

Describe your ServerBench setup:

Please send us information on the version of ServerBench you're using, your operating system name and version number, the network operating system name and version number, the network protocol and version number. If you've filled out the disclosure database information ServerBench uses to create results tables 4 and 5 (server and client disclosure), you can simply send us those tables.

ServerBench version number:_______ for server operating system:____________________

Server:____

Controller:____

Client:____

Network operating system:____

Network protocol:____

Details of the problem:

Which test suite were you running? (If you were running a test suite you created or a modified standard test suite, send us a copy of the test suite.)

____

____

Please write any server error messages you received here: ______________________

____

____

Please write any controller error messages you received here and the stage the test suite was at here:

____

____

Please write any client error messages you received and the stage the client was in here:__________

____

____

Can you reproduce the problem? ______ Other comments: ______________________

____

Please send this form to:

Fax number: **(919) 380-2879**
or Mail: Ziff-Davis Benchmark Operation 1001 Aviation Parkway, Suite 040 Morrisville, NC 27560 Attention: ServerBench Technical Support

Benchmark Request Form

Please check the boxes of the products you want:

___ **Ziff-Davis PC Benchmarks CD-ROM**

Contains the **Winstone®** and **WinBench®** benchmarks for desktop PCs.

___ **Ziff-Davis Server Benchmarks CD-ROM**

Contains **NetBench®** for file servers with DOS, Windows for Workgroups, and Mac™ OS system clients and **ServerBench®** for client/servers for the server platforms Windows NT™ Server 3.51 using Digital™ Alpha™, MIPS®, PowerPC™, and x86-compatible processors, SCO® UnixWare®, SCO OpenServer Release 5, OS/2® Warp Server, NetWare® 4.1, and NetWare 4.1 SMP.

___ **Ziff-Davis Macintosh Benchmark CD-ROM** Contains **MacBench®** for Mac™ OS systems and Power Macintosh™.

Please send these products to:

Name: ____

Company: ____

Address: ____

City: ______________________________ State: _______________ Zip: _________

Country: ____

Telephone: ___________________________ FAX: __________________________

We answer requests in the order we receive them. We ship all benchmarks via 3rd-class U.S. Please allow 4-6 weeks for delivery. For faster shipment, provide your Federal Express account information below:

Your Federal Express account number: ____________________________________

Check one: ___ priority overnight ___ standard overnight

Please return this form:

Fax to: **(919) 380-2879**

or Mail to: Ziff-Davis Benchmark Operation 1001 Aviation Parkway, Suite 400 Morrisville, NC 27560