LANtastic Performance Optimization

Performance Objectives

At the end of this module you will be able to do the following:

1. Identify the AILANBIO.EXE, REDIR.EXE, SERVER.EXE, and LANCACHE.EXE switches that can affect network performance.
2. Identify the RANGE and DEFAULT switch settings, in LANtastic v6.0, of the above programs
3. Discuss the data flow between a LANtastic network workstation and a server and the buffers involved.
4. Discuss and describe a simple test sequence that will enable users to monitor the effect of changes made to the switch settings that can affect network performance.

You will need the following:

Four networked computers with LANtastic v 6.0 installed.

The Performance Module diskette, which contains the batch files and LANtastic commands used to perform the test process.

Overview of LANtastic Network Dataflow

There are many factors that can affect performance in a LANtastic network. The hardware that makes up the physical aspect of the network is probably the best area in which to try to improve overall network performance. However, trying to tell a customer that the best way to speed up their network is to replace their 10 year old file server with a PC from this decade is not always well received.

As a preface to actually making changes to LANtastic switches, it is best to have a basic understanding of the flow of information through the LANtastic network.

The following sections describe the flow of data from a LANtastic workstation to a server and from a server to a workstation; specifically, the examples are of WRITE and READ requests. (The numbers in parentheses correspond to the LANtastic network data flow charts in Figs. 1 through 4.) The section emphasizes the primary buffers used in network data transmission and points out which REDIR and SERVER switches are used to specify the buffers' number and size.

Write Requests

On the Workstation (Fig. 1)

When an application on a workstation issues a WRITE command, the data is held in the application's internal buffer (1).

If the data file being written to on the server is opened in sharable mode (2), *or* if the data being written is greater than the size of the REDIR buffer (3), *or* if the write consists of random-access data (4), the redirector constructs a network command block (NCB) containing the LANtastic SERVER WRITE command and a pointer to the application buffer (6). From that point, the NetBIOS transmits the command block and the data (7) over the network hardware (interface card and cable).

If the data file on the server is opened in nonsharable mode (2), *and* the write is smaller than the REDIR buffer size (3), *and* the write consists of sequential-access data (4), the write is buffered by REDIR (5). [*NOTE: Printing is always buffered by REDIR.*] As with all buffers, the buffered data is not released until the buffer is full, *or* the write is flushed, *or* the file is closed, at which time the redirector constructs the NetBIOS command block (6) with a pointer to the REDIR buffer. Then the NetBIOS transmits the command block and the data (7).

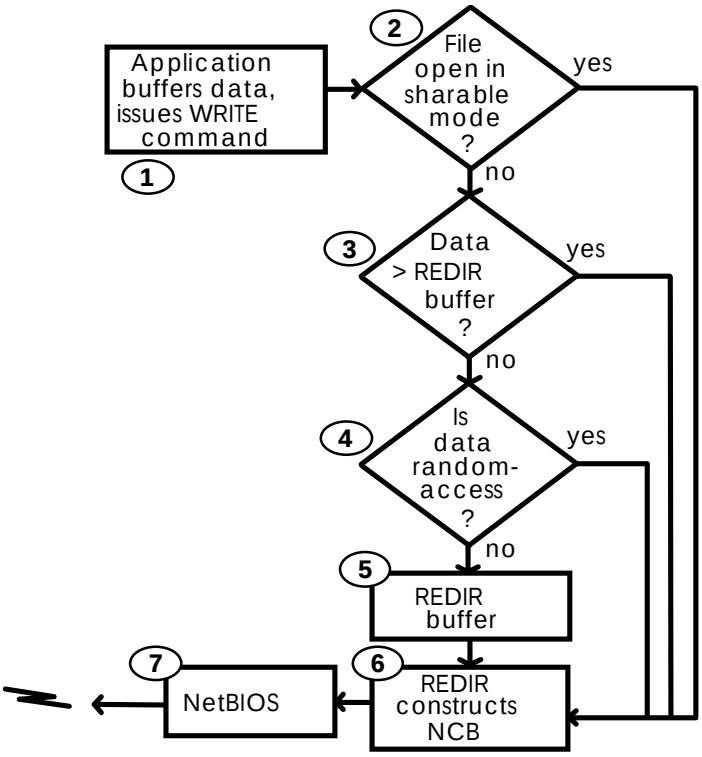


Fig. 1 -- WRITE Request: Data Flow on the Workstation

On the Server (Fig. 2)

The server's NetBIOS receives the command block and data over the network hardware (7) and SERVER writes them into the SERVER request buffer (8). (Any data that does not fit into that buffer is left in a NetBIOS buffer.) If the data is less than the size of the SERVER request buffer (9), the data is written to disk (10) and the procedure ends.

If the data is greater than the request buffer size, the system checks (11) to see if there is any data in the server request buffer besides the SERVER WRITE command. If there is, that data is written to disk (12), after which SERVER allocates a network buffer to receive additional data (13) and requests the additional data from the NetBIOS (14). Or, if there is no data in the request buffer (11) to be written to disk, SERVER allocates a network buffer to receive additional data (13) and requests the additional data from the NetBIOS (14).

There are actually two ways in which the NetBIOS can retrieve the additional data (14): by looking to see if there is any data left in the NetBIOS buffer and, if there is, retrieving it, or, if there is none left, by retrieving the additional data from the workstation's NetBIOS. In either case, it writes the additional data into the network buffer (15). SERVER then writes the additional data to disk (16). Steps 14-17 are repeated until there is no more data to be retrieved.

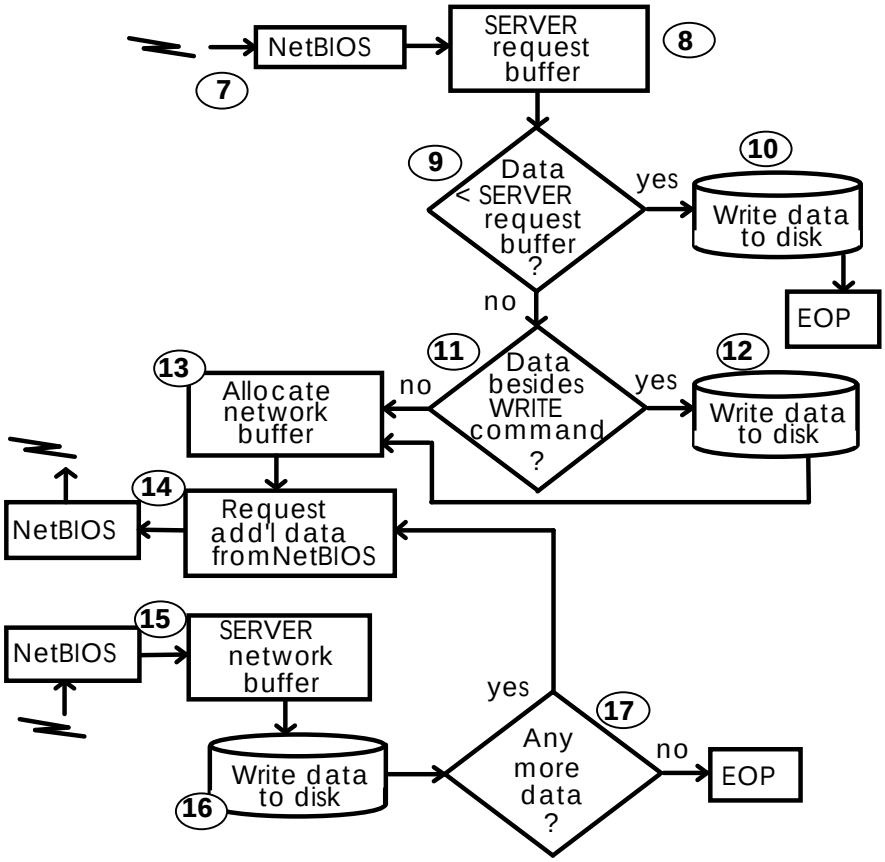


Fig. 2 -- WRITE Request: Data Flow on the Server

Examples of WRITE Requests

1. Suppose your application is writing a 512-byte file to the server (1), but the application buffer can only deal with 256 bytes at a time. The data is sequential-access data and the file is opened in nonsharable mode. The REDIR buffer is 512 bytes (the minimum size), so REDIR buffers the first 256 bytes of data and then waits for more data (see 2, 3, 4, 5 on flow chart). As soon as the REDIR buffer is full, REDIR constructs the NCB and hands it off, with the data, to the workstation's NetBIOS, which sends it to the server.

   The NCB and the 512 bytes of data come to the server's NetBIOS (7); the SERVER request buffer is only 50 bytes in size, so SERVER can put only the command information (for example, 14 bytes) and 36 bytes of the data into the request buffer. The remaining 472 bytes of data are held in a NetBIOS buffer until they are retrieved or another request on the system flushes them.

   The total amount of data in this WRITE request is not smaller than the request buffer (9), and there is data in the buffer besides the WRITE command (11), so SERVER writes the 36 bytes of data to the hard disk (12), allocates a network buffer (13), and requests additional data from the NetBIOS (14). If the 472 bytes of data are still in the NetBIOS buffer, the NetBIOS delivers them to the SERVER network buffer (15), and SERVER writes them to the disk (16).

For the following examples, trace the flowcharts in Figure 1 and Figure 2 and determine which buffers would be utilized in each data exchange.

2. The application issues a WRITE command for 5K bytes of random-access data; the file is opened in nonsharable mode. The REDIR buffer is 10K. The SERVER request buffer is 15K; the SERVER network buffer is 5K.

3. The application issues a WRITE command for 10K bytes of random-access data; the file is opened in nonsharable mode. The application buffer is 2.5K bytes. The WRITE command is 14 bytes. The REDIR buffer is 10K. The SERVER request buffer is 14 bytes; the SERVER network buffer is 5K.

Read Requests

On the Workstation (Fig. 3)

In preparing to read data from the server, an application on a workstation allocates an internal buffer and issues a READ request (1).

If the data file being read from on the server is opened in sharable mode (2), *or* if the data being requested is greater than the size of the REDIR buffer (3), *or* if the requested data is random-access data (4), REDIR constructs a NetBIOS command block (5) asking for the actual data requested by the application and containing the LANtastic SERVER READ command. The NetBIOS transmits the request over the network hardware (6) to the server. When the actual data requested by the application is received by the NetBIOS on the workstation (6x), it is written to the application's internal buffer (8).

If the data file on the server is opened in nonsharable mode (2), *and* the READ is smaller than the REDIR buffer size (3), *and* the READ is asking for sequential-access data (4), REDIR checks to see if the requested data is in the REDIR buffer (7). If it is there, the requested data is written to the application buffer (8). If the data is not in the REDIR buffer, REDIR constructs an NCB requesting one REDIR buffer's worth of data from the server (9). Then the NetBIOS transmits the command block over the network hardware to the server (10). When the buffer's worth of data is received by the NetBIOS on the workstation (10x), it is written to the REDIR buffer (11). From there, the actual data requested by the workstation is written to the application's internal buffer (8). The remainder of the data in the REDIR buffer is retained for the next READ.

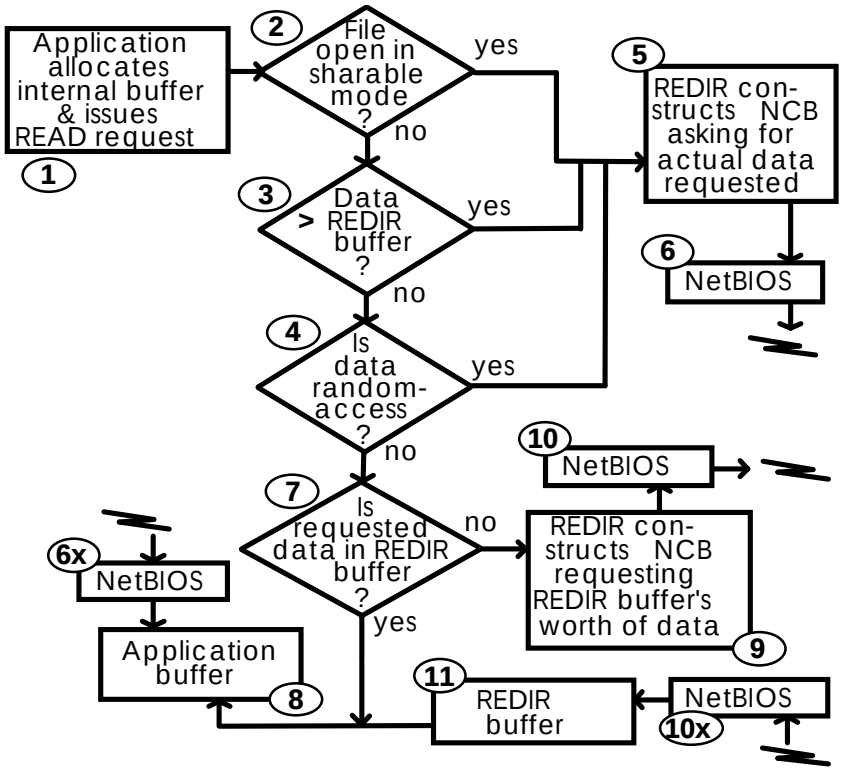
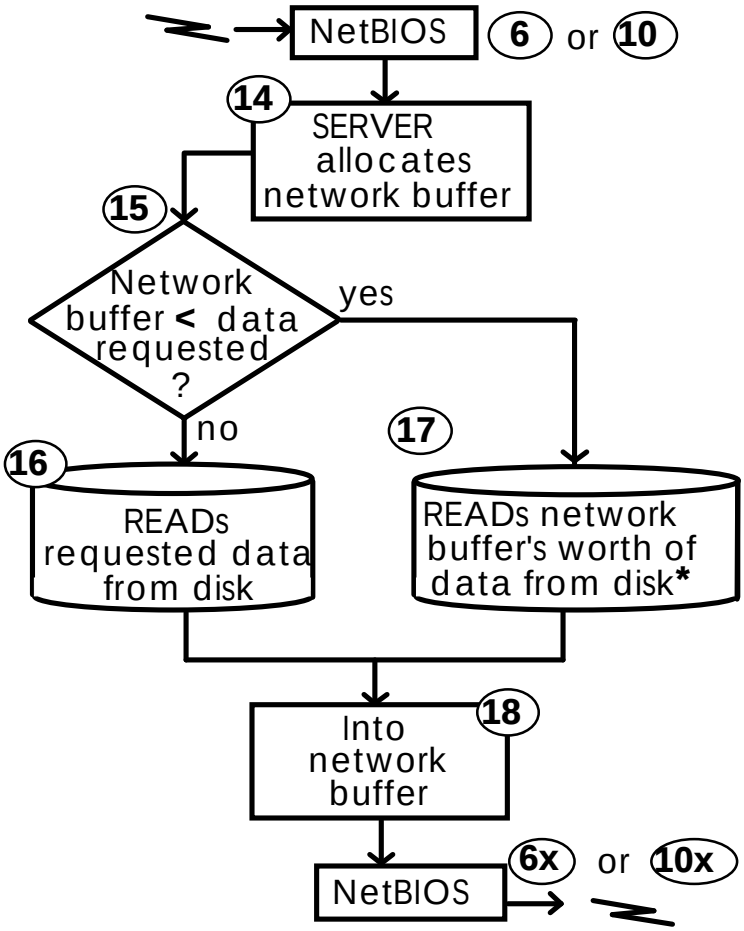


Fig. 3 -- READ Request: Data Flow on the Workstation

On the Server (Fig. 4)

The server's NetBIOS receives the command block from the workstation over the network hardware (6/10). SERVER allocates a network buffer (14) and determines if the network buffer is smaller than the data requested by the NCB (15). If it is *not* smaller, REDIR reads from the disk the requested data (17)--either the actual data requested by the application or a complete REDIR buffer's worth of data. SERVER reads the data into the network buffer (18) and the NetBIOS transmits it over the network (6x, 10x) to the workstation's NetBIOS.

If SERVER discovers (15) that the SERVER network buffer *is* smaller than the data requested by the NCB, SERVER reads one network buffer's worth of data from the disk (17) into the buffer (18), and the NetBIOS transmits it over the network to the workstation (6x/10x). The server continues to read and transmit one SERVER buffer's worth of data (17, 18, 6x/10x) until the workstation's NCB request has been satisfied.



17, 18, 6x/10x repeat until NCB request has been satisfied.

Fig. 4 READ Request: Data Flow on the Server

Examples of READ Requests

1. In this example, assume that an application has issued a READ request for 36,000 bytes of sequential data from a file opened in nonsharable mode on the server. Because the REDIR buffer is only 8K in size (3), REDIR constructs an NCB asking for the actual data requested by the application (5), and the NetBIOS sends the request out over the network hardware to the server's NetBIOS (6).

On the server, the NetBIOS receives the READ request (6) and SERVER allocates a network buffer for the data (14). In this case, the network buffer size is only 24K (15), so SERVER does the best it can to satisfy the workstation's request for 36,000 bytes, by reading 24K of data from the disk (17) and writing it to the network buffer (18). SERVER hands off that 24K to the NetBIOS, which sends it on to the workstation (6x). SERVER goes back to the disk and reads the remaining 12K of data (17) into the network buffer (18), from where the NetBIOS sends it on to the workstation.

Meanwhile, back on the workstation (6x), the NetBIOS receives the first 24K and SERVER writes it to the application buffer (8). Those steps are repeated with the next 12K of data as it arrives.

For the following examples, trace the flowcharts in Figure 3 and Figure 4 and determine which buffers would be utilized in each data exchange.

2. An application issues a READ request for 24 bytes of sequential data from a file opened in nonsharable mode. The REDIR buffer is 1K.

3. An application issues a READ request for 36K of sequential data from a file opened in nonsharable mode. The REDIR buffer is 10K. The SERVER network buffer is 24K.

4. An application issues a READ request for 1M of sequential data from a file opened in nonsharable mode. The REDIR buffer is 24K. The SERVER network buffer is 24K.

5. An application issues a READ request for 50 bytes of random-access data from a file opened in nonsharable mode. The REDIR buffer is 5K. The SERVER network buffer is 5K.

Buffers Used in Network Data Transmission

The primary buffers just addressed are the application buffer(s) and REDIR buffer(s) on the workstation, and the SERVER request buffer(s) and SERVER network buffer(s) on the server.

1. On the Workstation:

   LANtastic network workstations have one buffer that can affect performance, that is the REDIR, or Redirector, buffer.

   The REDIR buffer is controlled by two switches:

   - BUFFERS =
   - SIZE =

2. On the Server:

   A LANtastic server (DOS/Windows-based) has two buffers that can be used to affect performance. These buffers are:

   a. SERVER REQUEST BUFFER

      The server request buffer is controlled by two switches:

      - LOGINS= (or Maximum users in the NET_MGR menu)
      - REQUEST_SIZE=

   b. SERVER NETWORK BUFFERS

The network buffers can be manipulated by the following switches:

- NETWORK_TASKS=
- NETWORK_BUFFER_SIZE=

LANtastic Switches

This section provides a listing of the commands, switches and arguments for the LANtastic programs that affect network performance. Each command name is followed by a description of its purpose, its correct syntax, and explanations of its optional arguments (if any). You may then see examples for using the command.

Many, but not all, of the command-line switches in the Artisoft LANtastic Network Operating System are addressed here. Attention is given to those switches whose value directly affects network performance. Since network performance is dependent upon which applications are running and what hardware is in use, absolute values cannot be recommended. The better you understand the system's functionality, the easier it will be to tune the system for optimal performance.

NOTE: The most commonly used LANtastic switches are also described in Appendix A of the LANtastic Installation and Management Guide.

1. Universal Switches

   Although these switches do not directly affect performance, they are very helpful in determining the exact syntax for valid switches and the range of values for each switch.

   a. DESCRIBE

      The DESCRIBE switch provides you with version information about the program. When you use the DESCRIBE switch, the program doesn't install itself.

   b. HELP or?

      The HELP or ? switch provides you with a list of command line switches for the program, along with the valid range of values for each switch. When you use the HELP or ? switch, the program doesn’t install itself.

   c. VERBOSE

      The VERBOSE switch provides detailed information about the configuration of a program after it loads.

2. AILANBIO.EXE Switches

   Default values for the AILANBIO switches can be determined by “hand-loading” AILANBIO.EXE with no switches, as follows:

   ```
   C:\LANTASTI> AILANBIO/VERBOSE
   ```

   This will bring up the following:

```
Adapter Independent AI-LANBIOS(R) V4.06 - (C) Copyright 1993 ARTISOFT Inc.
NodeRunner/SI AI-LANBIOS(R) driver V4.04 - (C) Copyright 1993 ARTISOFT Inc.

Command line                /verbose
Adapter number              0         Low level MPX number           C7
Maximum number of NCBs      32        Run burst time in ticks        10
Maximum number of sessions  32        System timeout in 1/2 seconds 8
Maximum number of names     16        Retry period in ticks          2
Default number of NCBs      32        ACK timeout in ticks           2
Default number of sessions  32        Number of buffers              1
Buffer size                 570       Initial send size              570
Routing protocol            LANtastic IPX packet size          N/A
Post level       3        Bytes of memory used          20912

                ---- AI-LANBIOS(R) Installed ----
```

To determine the valid range of values for AILANBIO.EXE, just type in:

```
C:\LANTASTI> AILANBIO/?
```

This will display a list of the valid switches and their range of values on your screen.

a. BUFFERS= Range 0 to 254 decimal (1)

The effect of this switch on system performance depends primarily on the applications using the NetBIOS. For example, there is no performance advantage to having more than one buffer for each NetBIOS session actively receiving messages, since only one message at a time can be received in each session. Also, when the message is received, if the size of the receiving application's request buffer (an AILANBIOS buffer) is at least as large as the sending application's initial-send buffer (also an AILANBIOS buffer), buffering is not needed. In the case of REDIR, this is always the case. It is also true of SERVER if the request size is set to at least as large as the workstations' initial send size.

The memory required for each buffer includes a few bytes of overhead (rounded up to 16 bytes), required to maintain status information.

b. INITIAL_SEND_SIZE= Range 1 to 65535 decimal (570)

The size of the initial-send buffer should not exceed the packet size, nor should it exceed the size of the SERVER request buffer. If it does, a NetBIOS buffer is used, which adds more overhead to the system and decreases performance. See the SERVER REQUEST_SIZE= switch for recommended values.

c. RUN_BURST= Range 0 to 254 decimal (10)

This switch determines the amount of time, in ticks, that AILANBIO can use for network activity until it has to release control back to DOS. Once it has taken control, AILANBIOS can run until it has run out of things to do, or until its runburst time has been exceeded, in which case it will exit and reschedule at the next opportunity. Be aware that setting this to a higher value may improve network performance but can adversely impact any activity locally. Generally, this value will be increased only on machines that are acting as dedicated servers.

NOTE: ALONE does not modify this value. You may wish to use a higher value with ALONE, since foreground performance on a dedicated server is not an issue.

d. SIZE= Range 1 to 4300 decimal (570)

This switch sets the size of each allocated AILANBIOS buffer. Set it the same as the largest initial send size of all the nodes establishing a session with this computer.

3. REDIR SWITCHES

Default values for the REDIR switches can be determined by “hand-loading” REDIR.EXE with no switches, as follows:

```
C:\LANTASTI>REDIR SWORDFISH/VERBOSE
```

Which will produce the following:

```
Command line             SWORDFISH/VERBOSE

Machine name             SWORDFISH       Maximum logins          2
Number of buffers        1               Buffer size             1024
Beep cycle               4               Beep delay              4
PING Support             ENABLED         Chain sends             ENABLED
SMB Support              DISABLED        NCP Support             DISABLED
                      Pop-up Messages - ENABLED
Pop-up duration          15              Pop-up line             5
Bytes of memory used     17632

           ---- LANtastic (R) Redirector Installed ----
LANtastic (R) Redirector V6.00/AI - (C) Copyright 1994 ARTISOFT Inc.
Serial Number BBB-200     - 200 node license.
```

To determine the valid range of values for REDIR.EXE, just type in:

```
C:\LANTASTI>REDIR/?
```

This will display a list of the valid switches and their range of values on your screen.

a. BUFFERS= Range 1 to 64 decimal (1)

This switch determines the number of REDIR buffers until REDIR is restarted.

REDIR buffers are used for sequential file reads and writes that are smaller than the REDIR buffer size. If your application program has more than one file open at a time for a sequential read or write, the number of buffers on the REDIR command line should be increased to match the number of files that the application can have open at one time.

If you are running applications that print to more than one printer simultaneously, the number of buffers specified should also include one for each print stream.

NOTE: Printing is <u>always</u> buffered by REDIR

Large reads and writes, random-access reads and writes, and files opened in shared mode are not buffered. (Refer to the flow chart, if necessary.)

b. SIZE= Range 512 to 32768 decimal (1024)

This switch determines the size of each of the REDIR buffers specified by the BUFFERS= switch. There is a limit of 32K for REDIR buffers. Thus, it is only possible to have a single 32K REDIR buffer. If the SIZE= switch is set to 8K, then BUFFERS= cannot be more than four.

REDIR buffers are used only for sequential reads and writes. If the amount of information requested is larger than the buffer size, the buffer is bypassed and the data is read directly to/from the user's application buffer.

If the amount of data requested is smaller than the buffer, it is buffered until the buffer is full and then it is transmitted. Thus, a series of small reads and writes are combined into one large read/write, reducing the number of network transmissions.

Large reads and writes are transmitted directly from the application bypassing the REDIR buffer and eliminating the overhead of moving data between the REDIR

buffer and the application.

4. Server Switches

   The SERVER command-line parameters can be set or changed in the Server Startup Parameters menu in NET_MGR. The command-line switches are dynamic and only affect the current session; NET_MGR parameters are a stored configuration and can be overridden but not permanently changed by command-line entries.

   You can obtain a list of the current switch settings for SERVER.EXE by entering the command:

   ```
   C:\LANTASTI> SERVER/VERBOSE
   ```

   This will bring up the following information:

```
LANtastic (R) Server V6.00/AI - (C) Copyright 1994 ARTISOFT Inc.
Serial Number BBB-200

Command line            /VERBOSE

Machine name            SWORDFISH      Adapters in use         1
Maximum open files      in CONFIG.SYS  Maximum logins          3
Network buffer size     2048           Network tasks           1
Initial request size    32             Run burst               2
Cached resources        2              File Lookup Cache       0
Random access cache KB  0              Record lock hold time   DISABLED
Floppy direct access    YES            Park Drive at Shutdown  NO
Remote program load     DISABLED
Notification            DISABLED       Auditing                DISABLED
Server Control          DISABLED       Run buffer size         N/A
                           Printing - DISABLED
Printer buffer size     N/A            Printer tasks           N/A
Immediate Despooling    DISABLED       Print Streams           DISABLED
RPD Support             DISABLED       Initial despooling      STOPPED
                           Security - ENABLED
Send server ID          YES            Login Accounts          ENABLED
Access Control Lists    ENABLED        File Level ACLs         DISABLED
Group ACLs              ENABLED
                     Internal SHARE - DISABLED
Name Space              N/A            SHARE Locks             N/A
Shutdown key            Ctrl-Alt-DEL   Bytes of memory used    27728

           ---- LANtastic (R) Server Installed ----
```

   To determine the valid range of values for SERVER.EXE, just type in:

   ```
   C:\LANTASTI>SERVER/?
   ```

   This will display a list of the valid switches and their range of values on your screen.

   a. LOCK_HOLD_TIME= One of the following choices:

      DISABLED 2 3 4 5 (9) 13 18 27 36 45 54 63 72 81 90 108 126 144 162 180

      This specifies the length of time, in ticks, that the server will wait for a record lock before assuming the request has failed. Setting a higher value in this field can increase server response time if you have a lot of user contention within a database. Empirical testing has shown that this value increases the time it takes to open a file initially, but this decrease in speed must be balanced against the time savings in holding a request. If database files are not used often on the network, disable the LOCK_HOLD_TIME.

   b. LOGINS= Range 1 to 300 decimal (5)

      The LOGINS= switch specifies the number of logins allowed at any given moment and, hence, the number of request buffers to be created. For each user, the system allocates one request buffer plus approximately 100 bytes. (See the SERVER REQUEST_SIZE= switch for more information on buffer size.)

c. NETWORK_BUFFER_SIZE= One of the following choices:

2K (4K) 6K 8K 10K 12K 14K 16K 20K 24K 28K 32K 40K 48K 56K

One network buffer is allocated for each network task. This setting represents the largest chunk of file I/O that the server can work with at one time. Larger values improve the performance of large file read/write requests, since they don't have to be broken down into many small chunks. For example, COPY and XCOPY benefit greatly from a larger network buffer size specification.

d. NETWORK_TASKS= Range 1 to 32 decimal (1)

The number of network tasks determines how many operations can be handled simultaneously by the server. In other words, the number of network tasks determines how many network buffers are setup and can be allocated for SERVER's use. If more requests than network tasks are received, the excess tasks have to be rescheduled for the next active period.

e. PRINTER_BUFFER_SIZE= One of the following choices:

(512) 1K 2K 3K 4K 5K 6K 8K 10K 12K 14K 16K 18K 20K 24K 28K 32K

This setting does not affect overall system performance, but increasing the size of the printer buffer can improve print despooling.

f. PRINTER_TASKS= Range 0 to 5 decimal (1)

The number of printer tasks should be set to no more than the number of printers attached to the server. Do not set printer tasks to more than one on a server running Windows. Printing can be slowed down in some instances with Windows, Quicken, or Clipper 5.0 running on the server if printer tasks are set higher than one.

g. REQUEST_SIZE= Range 14 to 2048 decimal (32)

This switch specifies the size of each request buffer on a server. At its default size (32 bytes), only the smallest requests are handled in one transaction (lock, unlock, close, read, etc.). An intermediate size of 60-100 bytes improves the performance of most path operations (open, find first, rename, delete). A larger value of 570+ bytes can handle all requests except large writes, and it has the added benefit of eliminating the need for NetBIOS buffering in AILANBIOS.

h. RESOURCE_CACHE= Range 1 to 50 decimal (1)

Each time a user tries to access data on a server, the resource must be checked to see if the user has the proper permissions. A resource cache allows the ACL information about a resource to be stored in memory. Storing this information in memory eliminates the server's having to look up this information on the disk. This improves the performance of path operations (open, find first, rename, delete, etc.) when multiple disk or printer resources are in use at the same time on the server. There is no advantage to setting this value higher than the number of concurrently used resources. The overhead is about 850 bytes per entry.

i. RUN_BURST= Range 1 to 255 decimal (2)

SERVER schedules on five events:

- Timer ticks
- NetBIOS request completion
- Foreground task idle (For example, an application waiting for a keystroke

calls INT28.)

- Foreground task keyboard wait
- Foreground task DOS calls

Once SERVER has taken control, it runs until it has run out of things to do, or until its runburst time has been exceeded, in which case it exits and reschedules.

This switch, like the AILANBIO RUN_BURST, determines the amount of time, in ticks, that server can use to process network requests. Setting this switch to a higher value will improve network performance but can adversely affect local performance.

j. SEEK_CACHE= One of the following choices:

(NONE) 1K 2K 4K 8K 12K 16K 20K 24K 28K 32K 40K 48K 56K 64K

The seek cache caches information about where a file is located on disk and where in the file a user is currently located. It does not cache data from the file.

Seek caches only significantly affect the performance of random-access file operations if the files being randomly accessed are relatively large, typically 100K or more, and they are kept open and are accessed frequently enough to collect useful cache data.

For these conditions, even a small cache is helpful, and the larger the cache, the more improvement is realized. No publicly available benchmarks have been performed, but Artisoft has seen throughput improvements as high as 25%.

You can compute the optimal size for the seek cache with the following formula:

- Add the size of any large, frequently used database file (in bytes) and divide this by your hard drive's cluster size (normally 4096). This gives the number of clusters used. Each cluster used requires 32 bytes, and each open file requires 32 bytes. Maximum performance is achieved when the cache size is ≥ 32*(number of clusters used + number of files).

5. Lancache Switches

The LANcache product is a track-buffering cache rather than a sector-buffering cache. It stores its track map in conventional memory. Its lookup tables are stored in conventional memory to make the lookups faster and enhance the overall performance of the cache.

The LANcache utility will not perform “after writes” if VERIFY is ON. VERIFY must be OFF before the LANcache program can perform AFTER_IO_DELAYs.

Default values for the LANCACHE switches can be determined by “hand-loading” LANCACHE.EXE with no switches, as follows:

```
C:\LANTASTI>LANCACHE/VERBOSE
```

This will bring up the following:

```
LANtastic (R) Cache (LANCACHE) V3.00 - (C) Copyright 1992 ARTISOFT Inc.

           Command line /VERBOSE
       Cache memory type XMS                   Long-write delay 12 sec
  Cache memory size (KB) 1020                   After-I/O delay 1 sec
 Windows cache size (KB) 255                  Maximum run burst 10
 Maximum cache size (KB) 1020            Bytes per cached track 17408
Conventional memory used 25936                    Tracks cached 60
          Background I/O ENABLED

Physical drive 0 -------
                Caching ENABLED
          Delayed writes ENABLED
              Read-ahead ENABLED

                ---- LANtastic Cache Installed ----
```

To determine the valid range of values for LANCACHE.EXE, just type in:

```
C:\LANTASTI>LANCACHE/?
```

This will display a list of the valid switches and their range of values on your screen.

a. AFTER_IO_DELAY= Range 0 to 3600 decimal (3)

   This specifies how long the system should pause after the last disk read or write before flushing the cache to the hard disk. This timer is reset after every read and/ or write request, which allows disk writes to be made during the system's idle time. Since this timer is reset for every access, it never expires on a busy system.

b. CACHE_SIZE= Range 16 to 16000 decimal (all available memory)

   Set this switch to the largest possible size; experiment to find the optimal setting. The default is whatever memory is available at the time the LANcache program is executed.

c. LONG_WRITE_DELAY= Range 0 to 3600 decimal (6)

   This value is the longest time (in seconds) that the LANcache program will run without flushing from the cache to the disk. The value should always be greater than or equal to AFTER_IO_DELAY=. The longer this period of time, the more data is cached and then written in a single write. There is a point at which the savings in caching is offset by the delay caused by the write. The optimal value is determined by experimentation.

Summary of Switch Settings for Performance Imrovement

So, after wading through all of this material, what have we got? There are really only a few switches that have a significant impact on network performance. This section will describe some general considerations for getting a boost in performance. Keep in mind that no network is exactly the same. So think of these recommendations in the same way you would tell someonehow to adjust the water temperature in a shower -- by giving a general recommendation for the position of the hot and cold knobs and then saying “adjust the temperature from there to suit your needs”.

The switches you should consider first, and their “starting” values for adjusting performance are:

1. AILANBIO.EXE:

   a. RUN_BURST

Setting the RUNBURST switch to a higher value will help to improve network performance, change this switch from the default value only of the computer will be a dedicated network server.

b. INITIAL_SEND_SIZE

Changing the INITIAL_SEND_SIZE switch by itself really does not affect overall network performance. When this switch is set to the same value as a server's Request buffer it can improve performance for database access, or other applications that move very small files in and out of the server.

Be sure to set this switch to the same value as the server's REQUEST_SIZE.

Determine the optimum size by taking the size of the largest database record, in bytes, and add approximately 60 bytes for overhead.

For situations where there is no database server on the network, trying setting this value anywhere from 1024 to 1500 (1500 bytes is the maximum packet size on an Ethernet network).

2. REDIR.EXE

a. BUFFERS=

Multiple buffers almost always improve performance, but testing has shown that more than 4 buffers starts to use up memeory without adding much more to performance. Try setting this switch to 2 and adjust as needed.

b. SIZE=

Set this switch to the same value as the AILANBIO INITIAL_SEND_SIZE and the SERVER REQUEST_SIZE. If there is no database server on the network, try setting this switch at 1024.

3. SERVER.EXE

a. NETWORK_BUFFER_SIZE=

Determine the average size of the data that will be sent across the network and set the NETWORK_BUFFER_SIZE= switch to that value. Determining this value may not seem very intuitive, but can be estimated. As a first setting, try using a value of 8192 unless your needs clearly indicate another setting should be used.

b. NETWORK_TASKS=

Generally, any setting above one task will help to improve performance. Usually, a setting of 3 will provide an increase in performance without using too much memory. In general, network performance will level off as the value for NETWORK_TASKS approaches 8.

c. REQUEST_SIZE=

This is the first buffer filled by any WRITES from a workstation. To improve performance, set this switch to the same value as the AILANBIO INITIAL_SEND_SIZE and the REDIR SIZE=switches. If there is no database server on the network try setting this switch at 1024.

d. RUN_BURST=

Setting this switch to a higher value will improve network performance at the expense of local performance. In general, set to 255 on dedicated servers. If the server is a dedicated print server, set this switch between 100 and 150.

e. RESOURCE_CACHE=

On each server, determine the number of most frequently used resources, and add one or two to that value to determine the proper setting for resource caches. Usually 2 or 3 is a good start.

f. REQUEST_SIZE=

Set to the same value as AILANBIO's INITIAL_SEND_SIZE and REDIR's SIZE=.

4. LANCACHE.EXE

Just do it! Setting up a cache is prbably one of the best things that can be done to help improve performance.

Configure computers for testing

1. Choose one of the computers to act as the server for this test process. If the computers use different processors -- choose the fastest computer to act as the server.
2. On the server, perform the following steps:
   a. Be sure that all of the SERVER STARTUP PARAMETERS are at the default setting by using NET_MGR.EXE
   b. Change the name of the server to HOST on the REDIR line of STARTNET.BAT
   c. Check the AUTOEXEC.BAT file to make sure that SERVER will load automatically and that there are no caching programs loaded through AUTOEXEC.BAT

d. Create three subdirectories named 1111, 2222, and 3333 and copy the either the Windows subdirectoryor the LANTASTI subdirectory into each of these subdirectories by entering the command:

```
XCOPY C:\WINDOWS\*.*/S/E c:\XXXX
```

(Where "XXXX" is 1111, 2222, or 3333)

e. Use NET_MGR.EXE to create three new resources called 1111, 2222, and 3333 which point to the new subdirectories and reboot the server.

3. On each of the other workstations perform the following:

a. Using the PERFORMANCE TESTING diskette, copy the file TEST.BAT from the floppy to the root of the hard drive on each workstation.

b. Save the original AUTOEXEC.BAT file by renaming the file with your initials as the extension, then copy the TEST.BAT file to AUTOEXEC.BAT

c. Edit the AUTOEXEC.BAT file on each of the workstations and replace the characters XXXX with either 1111, 2222, or 3333 depending on the workstation and Reboot each of the workstations.

Upon rebooting, the workstations should come up with the following information on their screens:

```
START YOUR ENGINES!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

You are now ready to begin the test process.

About the test process

Running the tests described here will demonstrate the effects that changes to the LANtastic network have on network performance. (These tests do not compare the LANtastic Network Operating System to other networks.)

Each time you restart the workstations the following AUTOEXEC.BAT is invoked to begin the test process:

```
@ECHO OFF
ECHO _[1;37;44m
CLS
PROMPT $G$P$  $
PATH=C:\WINDOWS;C:\;C:\DOS;C:\LANTASTI
IF NOT EXIST C:\TEMP MD C:\TEMP
ECHO Y | DEL C:\TEMP\*.*

CLS
C:\LANTASTI\NR
C:\LANTASTI\AILANBIO
C:\LANTASTI\REDIR #
C:\LANTASTI\NET USER XXXX
C:\LANTASTI\NET LOGIN \\HOST
C:\LANTASTI\NET USE/D T: \\HOST\XXXX
ECHO _[1;37;42m
CLS
C:\LANTASTI\NET SHOW
ECHO.
ECHO.
ECHO.
ECHO.
ECHO    START YOUR ENGINES !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
PAUSE>NUL
ECHO _[1;37;42m
CLS

REM RESET CLOCK TO MIDNIGHT
```

```
ECHO 00:00:00 |TIME
ECHO TEST HAS BEGUN ..................................... PLEASE WAIT
XCOPY T:\*.* C:\TEMP
REM RESET CLOCK TO MIDNIGHT
ECHO 00:00:00 |TIME
ECHO TEST HAS ENDED .....  PLEASE MARK TIME, MAKE CHANGES, THEN REBOOT .....
```

Between each test, you will change one or more switch settings on the server and/or workstations. Rerun the test after each alteration, and note the (potentially) changed elapsed times on log sheet at the end of this training module. The objective is to discover which combination of settings produces the fastest time.

The tests are conducted from each workstation simultaneously so that the results will indicate server performance while under a workload. You should record on the attached log sheet, the elapsed time for each round of testing, displayed on each workstation's screen at the conclusion of each round. Choose one workstation and note it's time or approximate an average time across the workstations and note that.

Running the Test Process

The following steps are used to describe the changes that you will make to the machines in your test network. You should make the changes, reboot all of the machines and run the test under each set of circumstances.

1. With the server running and all of the workstations at the beginning screen, press any key on each of the workstations' keyboards to begin the test at default settings.

   When the test has completed note the elapsed time on the log sheet at the end of this module in the DEFAULT SETTINGS section.

2. After completing the DEFAULT SETTINGS test -- go into the Server's NET_MGR SERVER STARTUP PARAMETERS and set the RUN_BURST value to 255.

   Restart the computers and run the test sequence. Note the results on the log sheet next to the variable "Server Run Burst=255"

3. After completing the SERVER RUN_BURST test -- go into the Server's NET_MGR SERVER STARTUP PARAMETERS and set the RESOURCE_CACHE to 5(five). DO NOT reset the RUN_BURST value.

   Restart the computers, run the test and note the elapsed time in the "Server Resource Cache=5 area.

4. Edit the server's AUTOEXEC.BAT file to load the LANCACHE program. Previous versions of LANCACHE had to be loaded after REDIR.EXE but before SERVER.EXE. This is no longer the case. LANCACHE may be loaded without the LANtastic Network Operating System.

   Restart the computers, run the test and note the elapsed time in the "Server's LANCACHE loaded" area.

5. Use the NET_MGR menu to change the SERVER STARTUP PARAMETERS. Set the NETWORK_BUFFER_SIZE to 56K.

   Restart the computers, run the test sequence and note the results

6. Use the NET_MGR menu to change the SERVER STARTUP PARAMETERS. Set the NETWORK_BUFFER_SIZE to 40K.

   Restart the computers, run the test sequence and note the results

7. Use the NET_MGR menu to set the NETWORK_BUFFER_SIZE to 20K.

   Restart the computers, run the test sequence and note the results

8. Use the NET_MGR menu to set the NETWORK_BUFFER_SIZE to 40K and set Network Tasks to 5.

   Restart the computers, run the test sequence and note the results

9. Use the NET_MGR menu to set the NETWORK_BUFFER_SIZE to 20K and set Network Tasks to 5.

   Restart the computers, run the test sequence and note the results

10. Use the NET_MGR menu to set the Request Size to 1500.

    Restart the computers, run the test sequence and note the results

11. On each of the workstations, edit the AUTOEXEC.BAT file so that the AILANBIO line reads as follows:

    ```
    C:\LANTASTI\AILANBIO\INITIAL_SEND_SIZE=1500
    ```

    No changes need to be made to the server

    Restart the computers, run the test sequence and note the results

12. On each of the workstations, edit the AUTOEXEC.BAT file so that the REDIR line reads as follows (with XXXX representing the station name, 1111, 2222, or 3333):

    ```
    C:\LANTASTI\REDIR XXXX SIZE=8192
    ```

    No changes need to be made to the server

    Restart the computers, run the test sequence and note the results

13. On each of the workstations, edit the AUTOEXEC.BAT file so that the REDIR line reads as follows (with XXXX representing the station name, 1111, 2222, or 3333):

    ```
    C:\LANTASTI\REDIR XXXX SIZE=4096
    ```

    No changes need to be made to the server

    Restart the computers, run the test sequence and note the results

14. On each of the workstations, edit the AUTOEXEC.BAT file so that the REDIR line reads as follows (with XXXX representing the station name, 1111, 2222, or 3333):

    ```
    C:\LANTASTI\REDIR XXXX BUFFERS=4 SIZE=8192
    ```

    No changes need to be made to the server

    Restart the computers, run the test sequence and note the results

15. On each of the workstations, edit the AUTOEXEC.BAT file so that the REDIR line reads as follows (with XXXX representing the station name, 1111, 2222, or 3333):

    ```
    C:\LANTASTI\REDIR XXXX BUFFERS=4 SIZE=4096
    ```

    No changes need to be made to the server

    Restart the computers, run the test sequence and note the results.

LANtastic Network Performance Log Sheet		
Test #	**Variable Changed**	**Elapsed Time**
1	Default Settings	
2	Server RUN_BURST=255	
3	Server Resource_Cache=5	
4	Server's LANCACHE loaded	
5	Network Buffer Size= 56K (Memory use of Server _____)	
6	Network Buffer Size= 40K (Memory use of Server _____)	
7	Network Buffer Size= 20K (Memory use of Server _____)	
8	Network Buffer Size= 40K (Memory use of Server _____) Network Tasks = 5	
9	Network Buffer Size= 20K (Memory use of Server _____) Network Tasks = 5	
10	Server Request Size=1500	
11	Workstation AILANBIO INITIAL_SEND_SIZE=1500	
12	Workstation REDIR SIZE=8192	
13	Workstation REDIR SIZE=4096	
14	Workstation REDIR BUFFERS=4 SIZE=8192	
15	Workstation REDIR BUFFERS=4 SIZE=4096	

LANtastic Performance Optimization

Training Workbook
Module NOS15
Revision 00
4/4/95

ARTISOFT CORPORATE MISSION

To be a leading developer and marketer of network solutions to workgroups, small businesses, and remote network users.

MISSION

To establish channel franchises through sales execution, support, and technical services on a global basis.

CUSTOMER SUPPORT DEPARTMENT MISSION

To Serve customers...
Better than yesterday
Better than our competitors
Better than our customers expect.

EDUCATION SERVICES DEPARTMENT MISSION

To provide Artisoft Technical Support and Sales associates with the best training and best information tools in the industry.

Special thanks to Rick Roth, Education Services Manager, Lorna Peck, Courseware Developer, and Eric Fogal Technical Support supervisor, for their valuable input.

PLEASE NOTE: This is a training document. It is not intended for use as a troubleshooting reference. Product features and concerns can and often do change without notice. Please refer to source documents or Folios for the most current product information.