Evaluation of Client/Server Configurations For Analytic Processing

Barry R. Cohen, Planning Data Systems, Inc.

ABSTRACT

I participated in a project to design and performance test a statistician's work station in a client/server environment. The mix included Oracle on HP/UNIX servers as the data warehouse, SAS Software for analytic processing, and both the HP/UNIX servers and Windows PC clients as candidate sites for the SAS program execution and data serving. The performance test results were presented in my SUGI21 paper. Much time was required in that paper to detail the hardware and software involved, and to simply present the performance results. Little time was available for critical analysis of the findings. I now build upon the information in that paper, taking a critical look at what the performance numbers tell us about setting up a client/server environment for data analysis with SAS Software. This includes examination of the real cost of moving data over a network at the time it is analyzed, and how that cost sometimes leads to avoiding the client/server set up in favor of one where all tasks are handled on one platform, be it a server platform or a PC platform. This paper should be of interest to people who are designing a client/server environment for analytical processing with SAS, where data set sizes range from tens of megabytes to perhaps several hundred megabytes, and are looking for some hard performance numbers and evaluation to guide them.

INTRODUCTION

Last year (prior to SUGI21) I participated in a project to design and performance test a client/server environment that would support statistical (analytic) processing. The hardware and software mix included Oracle on one or more HP/UNIX servers as the DBMS for the data warehouse, SAS® Software for analytic processing, and both the servers and Windows PC's as candidate sites for the computations and data serving. As the project began, I had hoped to benefit from the experiences of others and not research and test every possible combination of cooperative processing and file serving. But I had found only limited information in recent SUGI proceedings. Much had been written, but these papers tended to be theoretical, describing what could now be done and generally how to do it. Little seemed to be available that gave some sense, with hard numbers, of how capable these environments were of supporting the kinds of tasks we hope to assign them to. So I felt I could contribute to the SAS community by publishing some of my own project-related research findings. I did this for SUGI21 in my paper titled Client/Server Performance Testing: Practical Results You Can Use Right Away.

The SUGI21 paper discussed in detail the hardware and software involved in the performance testing, and then presented in detail the test result data. It was a large amount of material to present, and it did not leave much space within the 10 page limit to critically evaluate the results. I felt nonetheless that I was making the right choice in providing details with only a limited amount of interpretation because performance test results have very limited value to others unless accompanied by a thorough description of what is really being tested. So I decided shortly after SUGI21 to present a follow-on paper at SUGI22 in which I would build upon the work in the SUGI21 paper. Specifically, I would take a step back and provide a more critical evaluation of the results presented in the SUGI21 paper.

It is important to note that the current (SUGI22) paper is not intended to stand fully alone without the SUGI21 paper. The reader can get some benefit from reading this paper only; however, the greatest benefit will come from reading the SUGI21 paper first. In particular, I will not repeat herein the detailed description of the test environment. I will assume that as given, and proceed to re-present the test results with more evaluation of the meaning of these results.

PURPOSE

Client/server processing has been with us for a few years now, and I believe that most people who use SAS Software do now understand generally what it is about. I also believe that people working in a heavy data analysis environment (as opposed to a transaction processing environment, for example) understand generally the benefit of designing a client/server configuration that keeps the data on the same platform where it is being analyzed. There is, of course, more than a little irony in this design idea. The benefits of client/server processing are often described, at the top level, as coming from the fact that processing tasks can be distributed among a variety of machines, connected by a network, with each machine doing the task or tasks it is best suited for. However, it often seems that when doing statistical analysis on measurable amounts of data, you pay a high price in time to serve the data over a network from a file server to a separate number crunching machine. So high, in fact, that people wind up settling on a configuration for statistical analysis which consolidates the processing on one platform instead of distributing it, in order to keep the number crunching near the data.

My purpose today is to present a succinct picture of several of the client/server configurations that are available today for doing statistical processing with SAS Software. And as part of this picture, I want to present an interpretation of my performance test results which looks carefully at where the data resides and where the number crunching occurs. I hope to enhance our understanding of this issue with the hard numbers found in my results. There is, of course, no one universal truth about how a client/server environment that supports statistical processing should be set up. But I hope that by providing some hard numbers that demonstrate in practice how various configurations perform, I can give you a better sense of the cost and benefit of the configurations you consider.

ENVIRONMENT & CONFIGURATIONS

The overall hardware and software of the production environment for my project that was available to be configured in one client/server configuration or another is illustrated in Figure 1. This environment was detailed in my SUGI21 paper, including discussion of the HP T500 Data Warehouse server and the HP K200 Statistical Server. In addition, the SUGI21 paper discussed the smaller HP G50 and G40 servers which functioned as surrogates for the T500 and K200 respectively, until the new hardware arrived.

Figure 2 illustrates the wide area network (WAN) and local area network (LAN) connections among the sites involved. The two production servers reside at the "server site" and connect to each other on a FDDI ring. The clients from "statisticians site #1" connect to the servers on a WAN connection which is another FDDI ring. The clients from "statisticians site #2" connect to the servers on a T3 line. The clients connect to each other and to the WAN on an Ethernet LAN using 10BaseT technology and the TCP/IP protocol. Finally, during development, the G50 and G40 servers connected to each other on a WAN T1 line, and the clients connected to these surrogate servers on the same WAN T1 line.

The chart just below summarizes the network speeds of the various WAN and LAN connections. I present the network speeds both in the traditional Megabits per second and in Megabytes per minute. I will use the latter units later in the paper. Note that since 25 Mbits/sec of the full 45 Mbits/sec on the T3 line were reserved for data, I show that number in parentheses. (The remainder of the line was reserved for voice and video).

WAN/LAN	Mbits/sec.	Mbytes/min.
T1	1.4	10.5
T3	45 (25)	338 (188)
FDDI	100	750
Ethernet 10BaseT	10	75

Figures 3, 4, and 5 illustrate the client/server configurations tested. The legacy mainframe was also tested as a benchmark. They are all detailed in the SUGI21 paper. The "All UNIX Server" configuration (Figure 3) is named as such because the server is handling both the data and the analytic computations. The PC clients are really not clients but rather terminals of the server via terminal emulation software. So there is no network traffic engendered during analytic processing (except that for terminal emulation).

The "PC Client/UNIX File Server" configuration (Figure 4) uses the UNIX machine as a file server and does the analytic computations by running SAS Software on the PC Clients. So here there is much network traffic engendered during the analytic processing. And the "All PC" configuration (Figure 5) involves a pre-download of the data from the server to the PC's, and then does the analytic processing on the PC's. Here there is as much network traffic as in the PC Client/UNIX File Server configuration, but it occurs in one burst up-front instead of on a flow basis as the analysis program runs.

CRITICAL EVALUATION

Data Preparation Program

Let's focus first on the test results when running the Data Preparation Program in these three configurations. (This test program and the test Analysis Program are detailed in the SUGI21 paper). All the test results for the Data Preparation Program are presented in Table 1. Table 1 is a simplification of the same table in the SUGI21 paper. I have done some rounding and dropped off the information about how many individual tests went into the average number presented in the table. The Data Preparation Program runs in 13 minutes in the All UNIX Server configuration with the model K200, and in 62 minutes with the model G40. In contrast, both configurations that involve moving the 48MB of data over the network take 108 minutes to run the program. And just to clarify a bit, in the PC Client/UNIX File Server configuration, I wrote the 43MB file produced as output by the program to a file on the PC. Had I written it back to the file server, which I justifiably should have to remain consistent with the approach to processing in this configuration, the results would have been even slower.

So the first thing we can say here is that it does make a difference when the data must be transferred over the network. Fortunately, because the All PC configuration involved a prior step download which was timed separately, we are in a position to determine just how much difference it makes. Table 1 indicates that it took 15 minutes to download the 48MB of data. So even if the PC were able to execute the program as fast as the server, (which was not the case), there is still a 15 minute overhead to move the data over the network. Compared to the K200's total execution time of 13 minutes, for example, this means more than doubling the total execution time.

Next we can look at whether the speed of the network makes any difference. We might think that any performance hit we take for involving the network can be overcome by upgrading the network. The first problem with such thinking is that the network is not one of the more easily upgraded pieces of the environment. For most organizations, it is probably much easier to slip in a faster server or a faster PC than it is to slip in a T3 line where a T1 now exists. So if the speed of the network matters, one might have to engage a client/server configuration on the given network that reflects that, instead of changing the WAN connection.

The second problem with this thinking is that my test data indicates that the speed of the WAN and LAN were not the problem. In the All PC configuration, the 48MB data set was moved from the UNIX server to the PC in 15 minutes. This was a transfer rate of 3.2 Mbytes/min. This particular test was run on the T1 line between the G40 Statistical Server and the PC Clients at the development site (see Figure 2), and it also involved the Ethernet LAN. We know from the chart presented above that the T1 line is rated at 1.4 Mbits/sec which equates to 10.5 Mbytes/min. And we know from the same chart that the Ethernet LAN is rated at 75 Mbytes/min. So we moved the data at 3.2 Mbytes/min (48MB/15min) on a T1 line rated at 10.5 Mbytes/min, or at about 30% of its full rating. The suggestion here is that since we were not approaching the top rated speed of the line, the line's capacity was probably not the bottleneck. Rather, the bottleneck probably came from the overall set of activities associated with moving data over a network. These include activities required at the server end to package the data and deliver it to the WAN, plus activities required to re-package the data for transfer from the WAN to the LAN, and finally activities required to move the data off the LAN and onto the PC. Apparently, these activities cannot occur as fast as the lines can move the data once it is on them. Since I did not re-run this particular test with the T3 line or FDDI ring in place of the T1 line, I cannot do a simple comparison to confirm or negate my hypothesis. However, I did do such a comparison for the Analysis Program, and I will present that later in the paper. (And the Analysis Program result does bear out my hypothesis).

One more point bears mention. So far, I have been evaluating the results from executing single instances of the program. We cannot execute multiple instances on the PC (because the operating system cannot support it). But it might be the case if we executed instances on multiple PC's, that we would begin to see an additional slow down based upon heavier traffic on the network. If so, then the overall performance hit we would take for engaging the network, with multiple instances of the program, would be an even greater one than that seen for the single instance test.

So here are a few things we can say at this point from our evaluation of the test results for the Data Preparation Program:

1. Client/server configurations that somehow serve the SAS data sets from a file server on one platform to a SAS program executing on another platform will in general execute slower than if the data and program are on the same platform.
2. The magnitude of the performance hit can be viewed two different ways, either as absolute wall clock time or as a percent increase in total execution time.
3. As wall clock time, I paid a price of about 15 minutes to move a 48MB SAS data set over my given network.
4. As a percent increase in total execution time, if the PC were able to execute the program as fast as the K200 server (i.e., 13 minutes), then the 15 minutes to move that data would represent a 115% increase, or more than doubling the execution time.
5. Note, too, from point #4, that the PC actually could not execute the program nearly as fast as the server, aside from network considerations. So for some analysis processing, there may be another performance hit associated with choosing a client/server configuration that executes the program on the PC. The actual magnitude of this hit will depend upon the power of the server versus the power of the PC, and it will also reflect the ratio of I/O services to CPU services required by the program. For my environment, the PC's were better able to compete with the servers on CPU services than on I/O services. (This point is covered further when we review the results of the Analysis Program below).
6. Finally, if we tried to emulate the multiple instance scenario by executing the program on multiple PC's simultaneously, we might see an even greater performance hit due to increased network traffic than we see for the single instance scenario.

Configuration	Single Instance of Program	Multiple (2) Instances of Program
Mainframe	33 min.	no measurable change from single-instance
All UNIX - G40	62 min.	121 min - same physical disk 56 min - separate physical disks
All UNIX - K200	13 min.	32 min - same physical disk 17 min - separate physical disks
PC Client/UNIX File Server	108 min.	not possible with single Windows session
All PC	108 min. (92 min execution + 15 min to download 48MB of data)	not possible with single Windows session

Table 1
Data Preparation Program Test Results (Single and Multiple Instance Processing)

Analysis Program

Let's focus now on the test results when running the Analysis Program in these same three configurations. (This test program is detailed in the SUGI21 paper). All the test results for the Analysis Program are presented in Table 2. Table 2 is a simplification of the same table in the SUGI21 paper. I have again done some rounding and dropped off the information about how many individual tests went into the average number presented in the table. The Analysis Program runs in 1.3 minutes in the All UNIX Server configuration with the model K200, and in 1.6 minutes with the model G40. In contrast, the configurations that involve moving the 43MB of analysis data over the network take about 10.5 to 11 minutes to run the program.

So the first thing we can say here is that once again it does make a difference when the data must be transferred over the network. And again, because the All PC configuration involved a prior step download which was timed separately, we are in a position to determine just how much difference it makes. Table 2 indicates that it took 9.5 minutes to download the 43MB of data. So even though the PC was able to execute the Analysis Program about as fast as the server, (1.5 minutes in the All PC configuration versus 1.3 minutes in the All UNIX Server configuration), there is still a 9.5 minute overhead to move the data over the network. Compared to the K200's total execution time of 1.3 minutes, this means more than an eight-fold increase (11 min/1.3 min) in the total execution time.

Next we can look at whether the speed of the network makes any difference. Here, too, as in the case of the Data Preparation Program, my test data indicates that the speed of the WAN and LAN were not the problem. In the All PC configuration, the 43MB data set was moved from the UNIX server to the PC in 9.5 minutes. This was a transfer rate of 4.5 Mbytes/min (43MB/9.5min). This particular test was first run on the T1 line between the G40 Statistical Server and the PC Clients at the development site (see Figure 2), which of course also involved the Ethernet LAN. It was later run in the production environment between the "server site" and the "statisticians site #2" on the T3 line. We know from the chart presented earlier that the T1 line is rated at 1.4 Mbits/sec which equates to 10.5 Mbytes/min. And we know from the same chart that the portion of the T3 reserved for data was 25 Mbits/sec, which equates to 188 Mbytes/min. So we moved the data at 4.5 Mbytes/min on a T1 line rated at 10.5 Mbytes/min, or at about 43% of its full rating. And when we did the re-run on the T3 line, the total time for the program execution was essentially the same.

So, once again, as with the Data Preparation Program results, there is the suggestion that since we were not approaching the top rated speed of the T1 line, the line's capacity was probably not the bottleneck. Rather, the bottleneck probably came from the overall set of activities associated with moving data over a network. Once again, these include activities required at the server end to package the data and deliver it to the WAN, plus activities required to re-package the data for transfer from the WAN to the LAN, and finally activities required to move the data off the LAN and onto the PC. Apparently, these activities cannot occur as fast as the lines can move the data once it is on them. And since I did re-run this particular test with the faster T3 line in place of the T1 line, with essentially the same result, this additional test seems to confirm my hypothesis.

Finally, had we run this program on several PC's simultaneously, to emulate the multiple instance scenario that we were able to test in the All UNIX Server configuration, we might begin to see an additional slow down based upon heavier traffic on the network. If so, then the overall performance hit we would take for engaging the network, with multiple instances of the program, would be an even greater one than that seen for the single instance test.

So here are a few things we can say at this point from our evaluation of the test results for the Analysis Program. They are largely the same as for the Data Preparation Program:

1. Client/server configurations that somehow serve the SAS data sets from a file server on one platform to a SAS program executing on another platform will in general execute slower than if the data and program are on the same platform.
2. The magnitude of the performance hit can be viewed two different ways, either as absolute wall clock time or as a percent increase in total execution time.
3. As wall clock time, I paid a price of about 9.5 minutes to move a 43MB SAS data set over my given network.
4. As a percent increase in total execution time, the 11 minutes to execute the program (9.5 minutes to move the data plus 1.5 minutes to then execute the program), represents a 750% increase in execution time compared to the 1.3 minutes in the All UNIX server configuration.
5. This time essentially did not change when I ran the test on the faster WAN T3 line instead of the WAN T1 line. This seems to confirm my hypothesis that it is not the line's rated capacity that matters most. Rather, the rate limiting factor is probably the overall set of activities inherent in moving data onto, over, and off of a network line.
6. Note, too, from point #4, that for the Analysis Program, the power of the server versus the power of the PC was not another performance issue. In contrast, for the Data Preparation Program, I suspect that the higher ratio of I/O services to CPU services in the program led to a much better relative performance in the All UNIX

Server configuration aside from network considerations.

7. Finally, if we tried to emulate the multiple instance scenario by executing the program on multiple PC's simultaneously, we might see an even greater performance hit due to increased network traffic that we do not experience for the single instance scenario.

Configuration	Single Instance of Program	Multiple Instances of Program
Mainframe	0.6 min	no degradation observed
All UNIX - G40	1.6 min	2 by 2 users: 1.7 min 10 by 2 users: 5.1 min 20 by 2 users: 9.6 min
All UNIX - K200	1.3 min	2 by 2 users: 1.4 min 10 by 2 users: 5.2 min 20 by 4 users: 7.2 min
PC Client/UNIX File Server	10.4 min	not possible with single Windows session
All PC	11 .0 min (1.5 min execution + 9.5 min to download 43MB data)	not possible with single Windows session

Table 2
Analysis Program Test Results

ADDITIONAL THOUGHTS

Other Configurations

Another client/server configuration could involve replacing the HP model K200 as the environment for the statistical server with an NT server on the same LAN as the PC clients. In this case, the data would be extracted from the Oracle database on the T500 data warehouse server directly to the NT server, and there would be no UNIX environment for analytic processing. All analytic processing would occur in the PC-LAN environment. A potential benefit from this approach is that the data could be served faster to the clients in the PC Client/UNIX File Server configuration since it only has to move over the LAN from the NT server to the client PC. There would be no WAN involved. And if so, we would probably not have to even consider the All PC configuration because it would not offer much potential advantage over the PC Client/UNIX File Server configuration.

But my team saw two drawbacks to this configuration which led us to reject it. First, we knew that we had some analytic processing which was so I/O intensive, such as our Data Preparation Program, that we just did not feel the PC's would be able to compete with the UNIX server when executing it. We even considered upgrading the I/O subsystem of the PC clients, or building a series of "SAS compute servers" on the NT servers. But these ideas seemed to have more costs when compared to engaging a UNIX statistical server. And a second drawback here was that while the NT server might work well for the PC clients on the same LAN, we still had to deal with the problem of making this data available to clients on another LAN at the other statisticians' site. We felt that as performance for the other site lagged, that the users would start making multiple copies of the data in their local environment. And this would introduce an issue of version control and quality assurance that we wanted to avoid.

I also want to briefly mention one other client/server configuration that was not tested. It concerns having the PC clients remotely submit SAS code to the UNIX server, which then serves the results back to the LOG and OUTPUT windows of the PC-based SAS session but keeps any SAS data sets in the server environment. This configuration is discussed in my SUGI21 paper in the section titled "The Configuration Not Tested". The only point I want to make here about it is that it is quite similar to the All UNIX Server configuration in performance and it would not change any of the findings I am presenting in this paper.

Client/Server for Analytic Processing?

At this point it is fair to ask whether the concepts of client/server processing are really relevant to analytic processing, such as that done by statisticians with SAS software? And if not, why not? I first want to clarify the frame of reference for the question. In my overall processing environment, the analytic processing is only one of many processing activities occurring. Many other related applications use the clinical data warehouse on the T500 server, and they are comprised of some processing that occurs on the T500 server and other processing that occurs on the client. So I am not raising this question about the overall environment in which analytic processing occurs. Rather, I am asking it only for the more narrowly focused analytic processing that occurs within a larger environment.

I think that the concepts of client/server processing are potentially relevant to analytic processing, but configurations that serve the data over a network to a client that then handles the computations may not be successful. The outcome will be a function of:

1. the volume of data being served;
2. the ratio of the time needed to serve the data to the total time needed to execute the programs involved;
3. the ability of the clients to handle both CPU services and I/O services as compared to the servers;
4. the expectations for performance in the particular organization involved;
5. to a certain extent, the capacity of the WAN and LAN involved, but perhaps to a lesser extent than initially thought.

One more comment about this might help us understand why certain client/server configurations which have such appeal in other types of processing have less appeal here. Analytic processing is distinguished from some other processing, such as transaction processing, in that it tends to involve processing entire sets of data at one time as opposed to retrieving and processing only small subsets of larger data sets (such as one or two customer records from a large customer database). In the latter case, there is little impact if the data to be processed is on the other side of a network because it is a small volume to transport over the network. However, in the analytic processing case, we tend to need to transport full sets of data instead of just individual records, and we are impacted by the transport time in a way that we are not with other types of processing.

CLOSING COMMENT

Design of a client/server environment is difficult because of the many processing options available and tradeoffs among them. I hope that I have provided you information in this paper to understand the tradeoffs and weigh them with the aid of some hard performance numbers.

REFERENCES

Cohen, B.R. (1996), "Client/Server Performance Testing: Practical Results You Can Use Right Away", Proceedings of the Twenty-First Annual SAS User Group International Conference, 21.

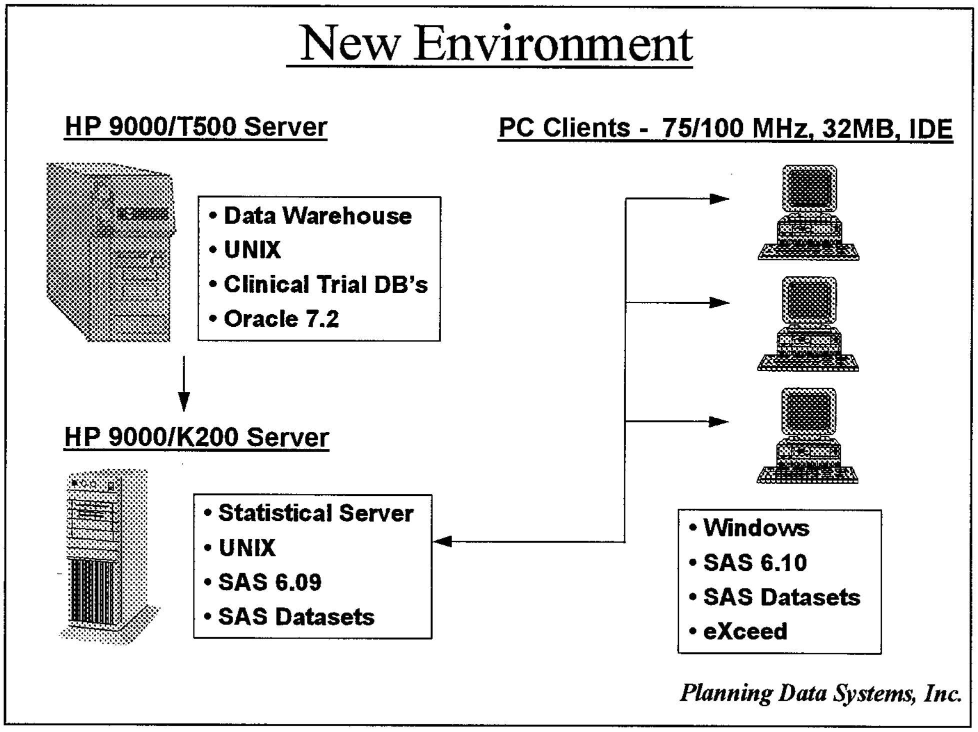
SAS Institute Inc. (1995), Client/Server Computing with the SAS System: Tips and Techniques, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1995), Building Your Client/Server Solution with the SAS System, Cary, NC: SAS Institute Inc.

For more information about this paper, contact the author:

Barry R. Cohen
Planning Data Systems, Inc.
PO Box 666
Ardmore, PA 19003
610-649-8701

Mr. Cohen is an independent information systems consultant, specializing in application development, object oriented programming, and client/server solutions. He has been using SAS software since 1980 in a wide variety of industry applications, including a focus on the pharmaceutical industry.



New Environment

HP 9000/T500 Server

- Data Warehouse
- UNIX
- Clinical Trial DB's
- Oracle 7.2

HP 9000/K200 Server

- Statistical Server
- UNIX
- SAS 6.09
- SAS Datasets

PC Clients - 75/100 MHz, 32MB, IDE

- Windows
- SAS 6.10
- SAS Datasets
- eXceed

Planning Data Systems, Inc.

Figure 1: The New Environment

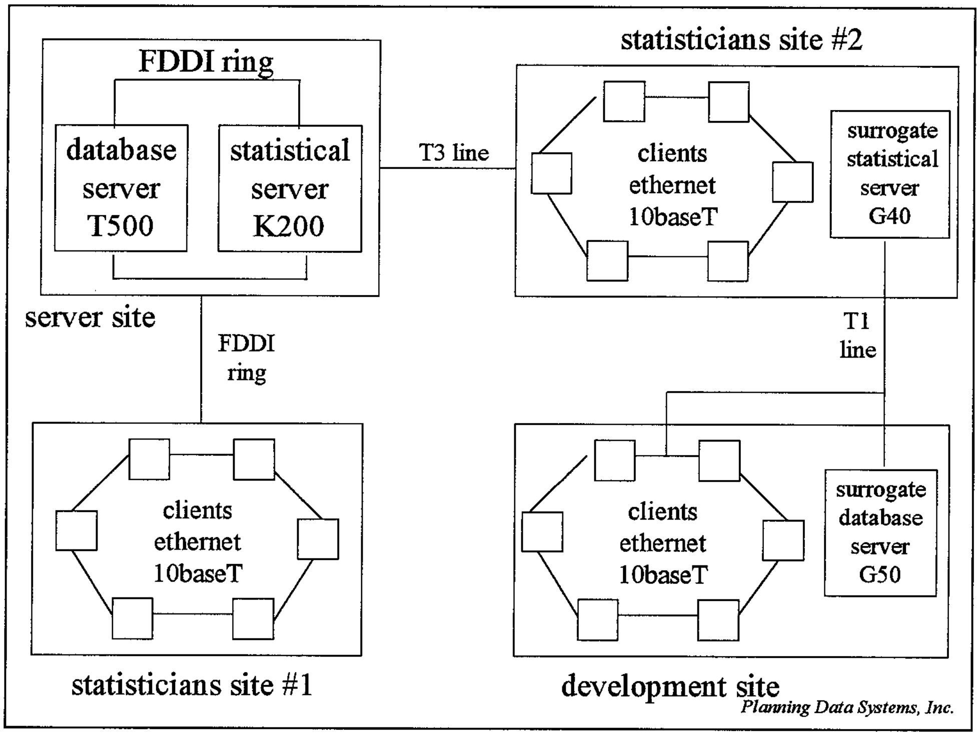
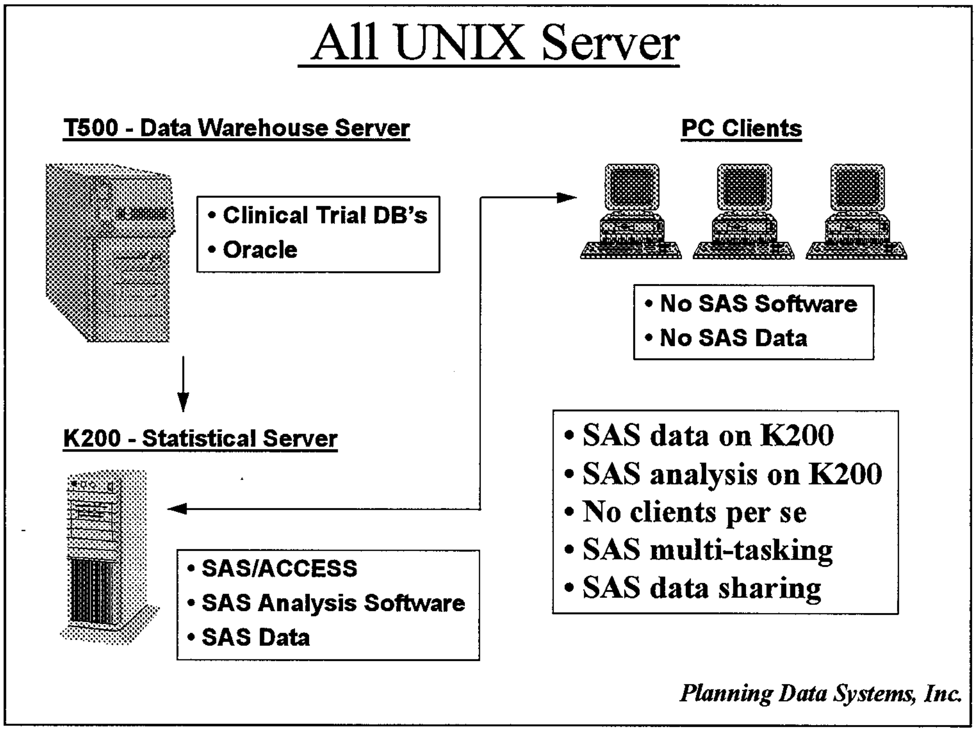


Figure 2: Network Connections



All UNIX Server

T500 - Data Warehouse Server

- Clinical Trial DB's
- Oracle

K200 - Statistical Server

- SAS/ACCESS
- SAS Analysis Software
- SAS Data

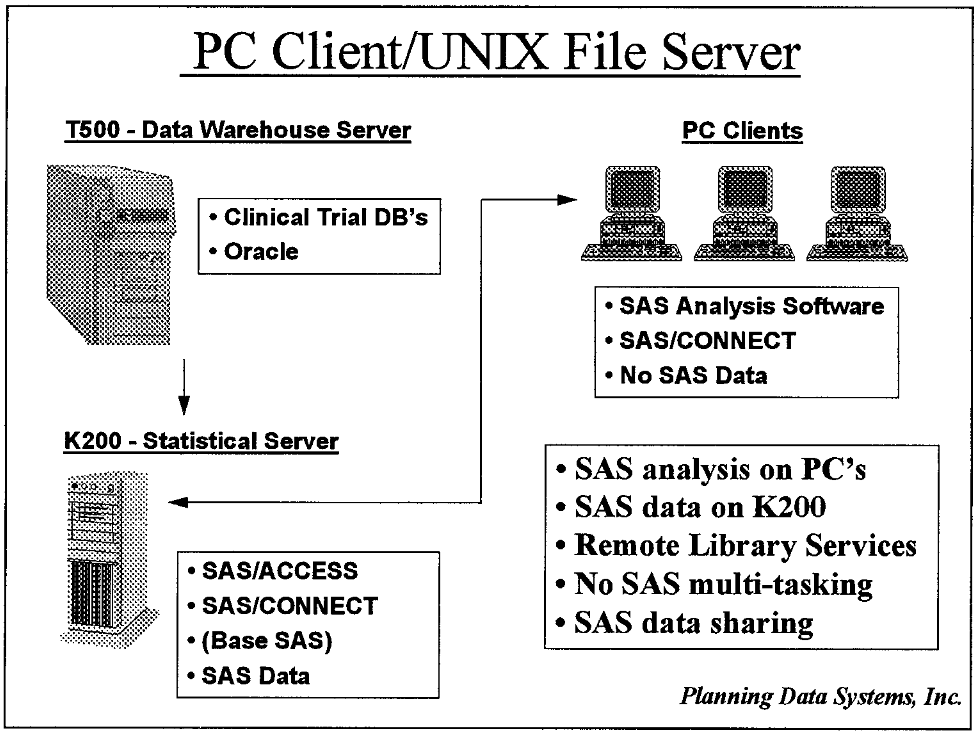
PC Clients

- No SAS Software
- No SAS Data

- SAS data on K200
- SAS analysis on K200
- No clients per se
- SAS multi-tasking
- SAS data sharing

Planning Data Systems, Inc.

Figure 3: All UNIX Server Configuration



PC Client/UNIX File Server

T500 - Data Warehouse Server

- Clinical Trial DB's
- Oracle

K200 - Statistical Server

- SAS/ACCESS
- SAS/CONNECT
- (Base SAS)
- SAS Data

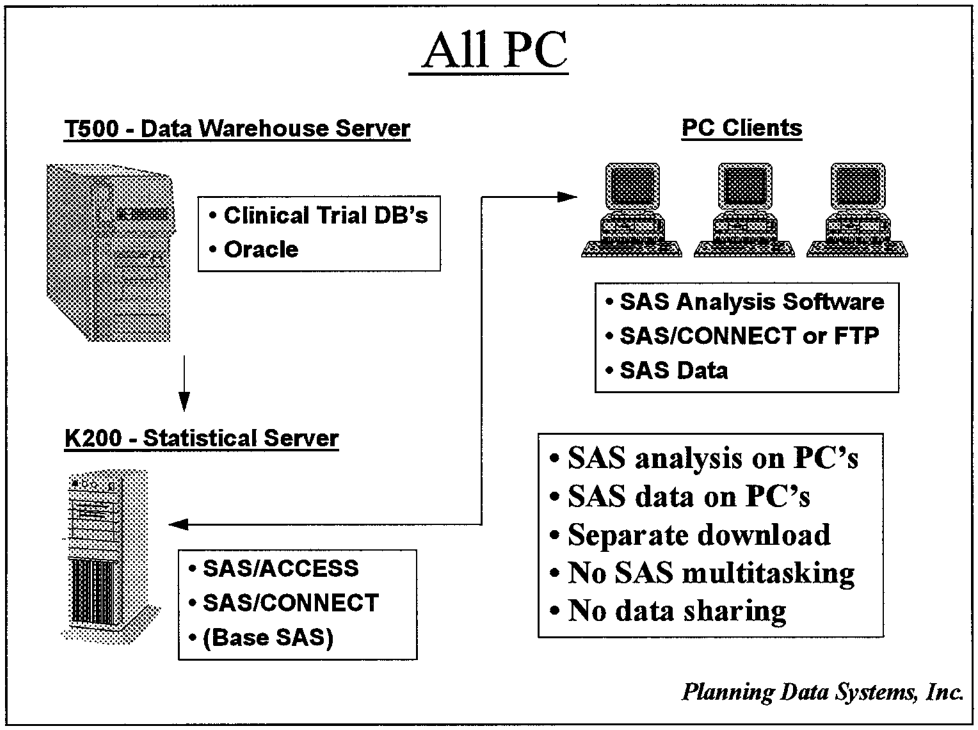
PC Clients

- SAS Analysis Software
- SAS/CONNECT
- No SAS Data

- SAS analysis on PC's
- SAS data on K200
- Remote Library Services
- No SAS multi-tasking
- SAS data sharing

Planning Data Systems, Inc.

Figure 4: PC Client/UNIX File Server Configuration



All PC

T500 - Data Warehouse Server

- Clinical Trial DB's
- Oracle

K200 - Statistical Server

- SAS/ACCESS
- SAS/CONNECT
- (Base SAS)

PC Clients

- SAS Analysis Software
- SAS/CONNECT or FTP
- SAS Data

- SAS analysis on PC's
- SAS data on PC's
- Separate download
- No SAS multitasking
- No data sharing

Planning Data Systems, Inc.

Figure 5: All PC Configuration