WILEY

WILEY PROFESSIONAL COMPUTING

1992 COMPUTER PRESS AWARD WINNER

THE ART OF COMPUTER SYSTEMS PERFORMANCE ANALYSIS

Techniques for Experimental Design, Measurement, Simulation, and Modeling

Raj Jain

CHAPTER 2

COMMON MISTAKES AND HOW TO AVOID THEM

Wise men learn by other men's mistakes, fools by their own.
—H. G. Wells

In order to motivate the use of proper methodology for performance evaluation, this chapter begins with a list of mistakes observed frequently in performance evaluation projects. This list then leads to the formulation of a systematic approach to performance evaluation. Various steps in correctly conducting a performance evaluation study and the order in which the steps should be carried out are presented.

2.1 COMMON MISTAKES IN PERFORMANCE EVALUATION

Unlike the games discussed in Section 1.2, most of the mistakes listed here are not intentional. Rather, they happen due to simple oversights, misconceptions, and lack of knowledge about performance evaluation techniques.

1. *No Goals*: Goals are an important part of all endeavors. Any endeavor without goals is bound to fail. Performance evaluation projects are no exception. The need for a goal may sound obvious, but many performance efforts are started without any clear goals. A performance analyst, for example, is routinely hired along with the design team. The analyst may then start modeling or simulating the design. When asked about the goals, the analyst's answer typically is that the model will help answer all design questions that may arise. A common claim is that the model will be flexible enough to be easily modified to solve different

problems. Experienced analysts know that there is no such thing as a general-purpose model. Each model must be developed with a particular goal in mind. The metrics, workloads, and methodology all depend upon the goal. The part of the system design that needs to be studied in the model varies from problem to problem. Therefore, before writing the first line of a simulation code or the first equation of an analytical model or before setting up a measurement experiment, it is important for the analyst to understand the system and identify the problem to be solved. This will help identify the correct metrics, workloads, and methodology.

Setting goals is not a trivial exercise. Since most performance problems are vague when first presented, understanding the problem sufficiently to write a set of goals is difficult. For example, a problem that was initially stated as one of finding a timeout algorithm for retransmissions on a network was later defined as a congestion control problem of finding out how the load on the network should be adjusted under packet loss. Once the problem is clear and the goals have been written down, finding the solution is often easier.

2. *Biased Goals*: Another common mistake is implicit or explicit bias in stating the goals. If, for example, the goal is "to show that OUR system is better than THEIRS," the problem becomes that of finding the metrics and workloads such that OUR system turns out better rather than that of finding the right metrics and workloads for comparing the two systems. One rule of professional etiquette for performance analysts is to be unbiased. *The performance analyst's role is like that of a jury*. Do not have any preconceived biases and base all conclusions on the results of the analysis rather than on pure beliefs.

3. *Unsystematic Approach*: Often analysts adopt an unsystematic approach whereby they select system parameters, factors, metrics, and workloads arbitrarily. This leads to inaccurate conclusions. The systematic approach to solving a performance problem is to identify a complete set of goals, system parameters, factors, metrics, and workloads. This is discussed in detail in Section 2.2.

4. *Analysis without Understanding the Problem*: Inexperienced analysts feel that nothing really has been achieved until a model has been constructed and some numerical results have been obtained. With experience, they learn that a large share of the analysis effort goes in to defining a problem. This share often takes up to 40% of the total effort. This supports the old saying: *A problem well stated is half solved*. Of the remaining 60%, a large share goes into designing alternatives, interpretation of the results, and presentation of conclusions. Development of the model itself is a small part of the problem-solving process. Just as cars and trains are a means of getting somewhere and not an end in themselves, models are a means of reaching conclusions and not

the final result. Analysts who are trained in modeling aspects of performance evaluation but not in problem definition or result presentation often find their models being ignored by the decision makers who are looking for guidance and not a model.

5. *Incorrect Performance Metrics*: A metric, as explained in Section 1.1, refers to the criterion used to quantify the performance of the system. Examples of commonly used performance metrics are throughput and response time. The choice of correct performance metrics depends upon the services provided by the system or subsystem being modeled. For example, the performance of Central Processing Units (CPUs) is compared on the basis of their throughput, which is often measured in terms of millions of instructions per second (MIPS). However, comparing the MIPS of two different CPU architectures, such as Reduced Instruction Set Computers (RISCs) and Complex Instruction Set Computers (CISCs), is meaningless since the instructions on the two computers are unequal. By manipulating the metrics, as shown in Chapter 11, it is possible to change the conclusions of a performance study. The considerations involved in selecting the right performance metrics are discussed in Section 3.2.

   A common mistake in selecting metrics is that analysts often choose those that can be easily computed or measured rather than the ones that are relevant. Metrics that are difficult to compute are ignored.

6. *Unrepresentative Workload*: The workload used to compare two systems should be representative of the actual usage of the systems in the field. For example, if the packets in networks are generally a mixture of two sizes—short and long—the workload to compare two networks should consist of short and long packet sizes.

   The choice of the workload has a significant impact on the results of a performance study. The wrong workload will lead to inaccurate conclusions. Workload selection is discussed in detail in Chapter 5. Benchmarking games that people play to show the superiority of their systems are discussed in Section 9.4.

7. *Wrong Evaluation Technique*: There are three evaluation techniques: measurement, simulation, and analytical modeling. Analysts often have a preference for one evaluation technique that they use for every performance evaluation problem. For example, those proficient in queueing theory will tend to change every performance problem to a queueing problem even if the system is too complex and is easily available for measurement. Those proficient in programming will tend to solve every problem by simulation. This marriage to a single technique leads to a model that they can best solve rather than to a model that can best solve the problem. The problem with these transformations is that they may introduce phenomena into the model that were not present in the original system or they may leave out some important phenomena that were in the original system.

An analyst should have a basic knowledge of all three techniques. There are a number of factors that should be considered in selecting the right technique. This topic is discussed further in Section 3.1.

8. *Overlooking Important Parameters*: It is a good idea to make a complete list of system and workload characteristics that affect the performance of the system. These characteristics are called parameters. For example, system parameters may include quantum size (for CPU allocation) or working set size (for memory allocation). Workload parameters may include the number of users, request arrival patterns, priority, and so on. The analyst can choose a set of values for each of these parameters; the final outcome of the study depends heavily upon those choices. Overlooking one or more important parameters may render the results useless.

9. *Ignoring Significant Factors*: Parameters that are varied in the study are called **factors**. For example, among the workload parameters listed above only the number of users may be chosen as a factor; other parameters may be fixed at their typical values. Not all parameters have an equal effect on the performance. It is important to identify those parameters, which, if varied, will make a significant impact on the performance. Unless there is reason to believe otherwise, these parameters should be used as factors in the performance study. For example, if packet arrival rate rather than packet size affects the response time of a network gateway, it would be better to use several different arrival rates in studying its performance.

   Factors that are under the control of the end user (or decision maker) and can be easily changed by the end user should be given preference over those that cannot be changed. Do not waste time comparing alternatives that the end user cannot adopt either because they involve actions that are unacceptable to the decision makers or because they are beyond their sphere of influence.

   It is important to understand the randomness of various system and workload parameters that affect the performance. Some of these parameters are better understood than others. For example, an analyst may know the distribution for page references in a computer system but have no idea of the distribution of disk references. In such a case, a common mistake would be to use the page reference distribution as a factor but ignore disk reference distribution even though the disk may be the bottleneck and may have more influence on performance than the page references. The choice of factors should be based on their relevance and not on the analyst's knowledge of the factors. Every attempt should be made to get realistic values of all relevant parameters and their distributions. For unknown parameters, a sensitivity analysis, which shows the effect of changing those parameters from their assumed values, should be done to quantify the impact of the uncertainty.

10. *Inappropriate Experimental Design*: Experimental design relates to the number of measurement or simulation experiments to be conducted and the parameter values used in each experiment. Proper selection of these values can lead to more information from the same number of experiments. Improper selection can result in a waste of the analyst's time and resources.

    In naive experimental design, each factor is changed one by one. As discussed in Chapter 16, this "simple design" may lead to wrong conclusions if the parameters interact such that the effect of one parameter depends upon the values of other parameters. Better alternatives are the use of the *full factorial experimental designs* and *fractional factorial designs* explained in Part IV.

11. *Inappropriate Level of Detail*: The level of detail used in modeling a system has a significant impact on the problem formulation. Avoid formulations that are either too narrow or too broad. For comparing alternatives that are slight variations of a common approach, a detailed model that incorporates the variations may be more useful than a high-level model. On the other hand, for comparing alternatives that are very different, simple high-level models may allow several alternatives to be analyzed rapidly and inexpensively. A common mistake is to take the detailed approach when a high-level model will do and vice versa. It is clear that the goals of a study have a significant impact on what is modeled and how it is analyzed.

12. *No Analysis*: One of the common problems with measurement projects is that they are often run by performance analysts who are good in measurement techniques but lack data analysis expertise. They collect enormous amounts of data but do not know how to analyze or interpret it. The result is a set of magnetic tapes (or disks) full of data without any summary. At best, the analyst may produce a thick report full of raw data and graphs without any explanation of how one can use the results. Therefore, it is better to have a team of performance analysts with measurement as well as analysis background.

13. *Erroneous Analysis*: There are a number of mistakes analysts commonly make in measurement, simulation, and analytical modeling, for example, taking the average of ratios and too short simulations. Lists of such mistakes are presented throughout this book during discussions on individual techniques.

14. *No Sensitivity Analysis*: Often analysts put too much emphasis on the results of their analysis, presenting it as fact rather than evidence. The fact that the results may be sensitive to the workload and system parameters is often overlooked. Without a sensitivity analysis, one cannot be sure if the conclusions would change if the analysis was done in a slightly different setting. Also, without a sensitivity analysis, it is difficult to access the relative importance of various parameters.

15. *Ignoring Errors in Input*: Often the parameters of interest cannot be measured. Instead, another variable that can be measured is used to estimate the parameter. For example, in one computer network device, the packets were stored in a linked list of buffers. Each buffer was 512 octets long. Given the number of buffers required to store packets, it was impossible to accurately predict the number of packets or the number of octets in the packets. Such situations introduce additional uncertainties in the input data. The analyst needs to adjust the level of confidence on the model output obtained from such data. Also, it may not be worthwhile to accurately model the packet sizes when the input can be off by as much as 512 octets. Another point illustrated by this example is the fact that input errors are not always equally distributed about the mean. In this case, the buffer space is always more than the actual number of octets transmitted on or received from the network. In other words, the input is biased.
16. *Improper Treatment of Outliers*: Values that are too high or too low compared to a majority of values in a set are called **outliers**. Outliers in the input or model output present a problem. If an outlier is not caused by a real system phenomenon, it should be ignored. Including it would produce an invalid model. On the other hand, if the outlier is a possible occurrence in a real system, it should be appropriately included in the model. Ignoring it would produce an invalid model. Deciding which outliers should be ignored and which should be included is part of the art of performance evaluation and requires careful understanding of the system being modeled.
17. *Assuming No Change in the Future*: It is often assumed that the future will be the same as the past. A model based on the workload and performance observed in the past is used to predict performance in the future. The future workload and system behavior is assumed to be the same as that already measured. The analyst and the decision makers should discuss this assumption and limit the amount of time into the future that predictions are made.
18. *Ignoring Variability*: It is common to analyze only the mean performance since determining variability is often difficult, if not impossible. If the variability is high, the mean alone may be misleading to the decision makers. For example, decisions based on the daily averages of computer demands may not be useful if the load demand has large hourly peaks, which adversely impact user performance.
19. *Too Complex Analysis*: Given two analyses leading to the same conclusion, one that is simpler and easier to explain is obviously preferable. Performance analysts should convey final conclusions in as simple a manner as possible. Some analysts start with complex models that cannot be solved or a measurement or simulation project with very ambitious goals that are never achieved. It is better to start with simple

models or experiments, get some results or insights, and then introduce the complications.

There is a significant difference in the types of models published in the literature and those used in the real world. The models published in the literature and, therefore, taught in schools are generally too complex. This is because trivial models, even when very illuminating, are not generally accepted for publication. For some reason, the ability to develop and solve a complex model is valued more highly in academic circles than the ability to draw conclusions from a simple model. However, in the industrial world, the decision makers are rarely interested in the modeling technique or its innovativeness. Their chief concern is the guidance that the model provides along with the time and cost to develop the model. The decision deadlines often lead to choosing simple models. Thus, a majority of day-to-day performance problems in the real world are solved by simple models. Complex models are rarely, if ever, used. Even if the time required to develop the model was not restricted, complex models are not easily understood by the decision makers, and therefore, the model results may be misbelieved. This causes frustrations for new graduates who are very well trained in complex modeling techniques but find few opportunities to use them in the real world.

20. *Improper Presentation of Results*: The eventual aim of every performance study is to help in decision making. An analysis that does not produce any useful results is a failure, as is the analysis with results that cannot be understood by the decision makers. The decision makers could be the designers of a system, the purchasers of a system, or the sponsors of a project. Conveying (or selling) the results of the analysis to decision makers is the responsibility of the analyst. This requires the prudent use of words, pictures, and graphs to explain the results and the analysis. *The right metric to measure the performance of an analyst is not the number of analyses performed but the number of analyses that helped the decision makers.*
21. *Ignoring Social Aspects*: Successful presentation of the analysis results requires two types of skills: social and substantive. Writing and speaking are social skills while modeling and data analysis are substantive skills. Most analysts have good substantive skills, but only those who have good social skills are successful in selling their results to the decision makers. Acceptance of the analysis results requires developing a trust between the decision makers and the analyst and presentation of the results to the decision makers in a manner understandable to them. If decision makers do not believe or understand the analysis, the analyst fails to make an impact on the final decision. Social skills are particularly important in presenting results that are counter to the decision makers' beliefs and values or that require a substantial change in the design.

Beginning analysts often fail to understand that social skills are often more important than substantive skills. High-quality analyses may be rejected simply because the analyst has not put enough effort and time into presenting the results. The decision makers are under time pressures and would like to get to the final results as soon as possible. They generally are not interested in the innovativeness of the approach or the approach itself. On the other hand, the analyst, having spent a considerable amount of time on the analysis, may be more interested in telling the decision makers about the innovativeness of the modeling approach than the final results. This disparity in viewpoint may lead to a report that is too long and fails to make an impact. The problem is compounded by the fact that the analyst also has to present the results to his/her peers who are analysts themselves and would like to know more about the approach than the final results. One solution, therefore, is to prepare two separate presentations (or reports) for the two audiences. The presentation to the decision makers should have minimal analysis jargon and emphasize the final results, while the presentation to other analysts should include all the details of the analysis techniques. Combining these two presentations into one could make it meaningless for both audiences.

Inexperienced analysts assume that the decision makers are like themselves and share the same beliefs, values, language, and jargon. This is often not true. The decision makers may be good at evaluating the results of the analysis but may not have a good understanding of the analysis itself. In their positions as decision makers, they have to weigh several factors that the analyst may not consider important, such as the political impact of the decision, the delay in the project schedule, or the availability of personnel to implement a particular decision. The analyst who makes an effort to understand the decision makers' concerns and incorporates these as much as possible into the presentation will have a better chance of "selling" the analysis than one who sees things only from his/her own point of view.

22. *Omitting Assumptions and Limitations*: Assumptions and limitations of the analysis are often omitted from the final report. This may lead the user to apply the analysis to another context where the assumptions will not be valid. Sometimes analysts list the assumptions at the beginning of the report but then forget the limitations at the end and make conclusions about environments to which the analysis does not apply.

The above discussion on common mistakes is summarized in Box 2.1, which presents a checklist of questions concerning performance analysis. All questions should be answered affirmatively. The list can also be used by the decision makers to review performance analyses presented to them.

Box 2.1 Checklist for Avoiding Common Mistakes in Performance Evaluation

1. Is the system correctly defined and the goals clearly stated?
2. Are the goals stated in an unbiased manner?
3. Have all the steps of the analysis followed systematically?
4. Is the problem clearly understood before analyzing it?
5. Are the performance metrics relevant for this problem?
6. Is the workload correct for this problem?
7. Is the evaluation technique appropriate?
8. Is the list of parameters that affect performance complete?
9. Have all parameters that affect performance been chosen as factors to be varied?
10. Is the experimental design efficient in terms of time and results?
11. Is the level of detail proper?
12. Is the measured data presented with analysis and interpretation?
13. Is the analysis statistically correct?
14. Has the sensitivity analysis been done?
15. Would errors in the input cause an insignificant change in the results?
16. Have the outliers in the input or output been treated properly?
17. Have the future changes in the system and workload been modeled?
18. Has the variance of input been taken into account?
19. Has the variance of the results been analyzed?
20. Is the analysis easy to explain?
21. Is the presentation style suitable for its audience?
22. Have the results been presented graphically as much as possible?
23. Are the assumptions and limitations of the analysis clearly documented?

2.2 A SYSTEMATIC APPROACH TO PERFORMANCE EVALUATION

Most performance problems are unique. The metrics, workload, and evaluation techniques used for one problem generally cannot be used for the next problem. Nevertheless, there are steps common to all performance evaluation projects that help you avoid the common mistakes listed in Section 2.1. These steps are as follows.

1. *State Goals and Define the System*: The first step in any performance evaluation project is to state the goals of the study and define what con-

stitutes the system by delineating system boundaries. Given the same set of hardware and software, the definition of the system may vary depending upon the goals of the study. Given two CPUs, for example, the goal may be to estimate their impact on the response time of interactive users. In this case, the system would consist of the timesharing system, and the conclusions of the study may depend significantly on components external to the CPU. On the other hand, if the two CPUs are basically similar except for their Arithmetic-Logic Units (ALUs) and the goal is to decide which ALU should be chosen, the CPUs may be considered the system's and only the components inside the CPU may be considered part of the system.

The choice of system boundaries affects the performance metrics as well as workloads used to compare the systems. Therefore, understanding the system boundaries is important. Although the key consideration in setting the system boundaries is the objective of the study, other considerations, such as administrative control of the sponsors of the study, may also need to be taken into account. If the sponsors do not have a control over some components, they may want to keep those components outside the system boundaries.

2. *List Services and Outcomes*: Each system provides a set of services. For example, a computer network allows its users to send packets to specified destinations. A database system responds to queries. A processor performs a number of different instructions. The next step in analyzing a system is to list these services. When a user requests any of these services, there are a number of possible outcomes. Some of these outcomes are desirable and some are not. For example, a database system may answer a query correctly, incorrectly (due to inconsistent updates), or not at all (due to deadlocks or some similar problems). A list of services and possible outcomes is useful later in selecting the right metrics and workloads.
3. *Select Metrics*: The next step is to select criteria to compare the performance. These criteria are called metrics. In general, the metrics are related to the speed, accuracy, and availability of services. The performance of a network, for example, is measured by the speed (throughput and delay), accuracy (error rate), and availability of the packets sent. The performance of a processor is measured by the speed of (time taken to execute) various instructions. The selection of the correct metrics is discussed in Section 3.2.
4. *List Parameters*: The next step in performance projects is to make a list of all the parameters that affect performance. The list can be divided into system parameters and workload parameters. System parameters include both hardware and software parameters, which generally do not vary among various installations of the system. Workload parameters are characteristics of users' requests, which vary from one installation to the next.

The list of parameters may not be complete. That is, after the first pass of the analysis, you may discover that there are additional parameters that affect the performance. You can then add these parameters to the list, but at all times keep the list as comprehensive as possible. This allows the analyst and decision makers to discuss the impact of various parameters and determine what data needs to be collected before or during the analysis.

5. *Select Factors to Study*: The list of parameters can be divided into two parts: those that will be varied during the evaluation and those that will not. The parameters to be varied are called **factors** and their values are called **levels**. In general, the list of factors, and their possible levels, is larger than what the available resources will allow. Otherwise, the list will keep growing until it becomes obvious that there are not enough resources to study the problem. It is better to start with a short list of factors and a small number of levels for each factor and to extend the list in the next phase of the project if the resources permit. For example, you may decide to have only two factors: quantum size and the number of users. For each of these two factors you may choose only two levels: small and large. The working set size and the type of workload may be fixed.

   The parameters that are expected to have a high impact on the performance should be preferably selected as factors. Like metrics, a common mistake in selecting the factors is that the parameters that are easy to vary and measure are used as factors while other more influential parameters are ignored simply because of the difficulty involved.

   In selecting factors, it is important to consider the economic, political, and technological constraints that exist as well as including the limitations imposed by the decision makers' control and the time available for the decision. This increases the chances of finding a solution that is acceptable and implementable.

6. *Select Evaluation Technique*: The three broad techniques for performance evaluation are analytical modeling, simulation, and measuring a real system. The selection of the right technique depends upon the time and resources available to solve the problem and the desired level of accuracy. The selection of evaluation techniques is discussed in Section 3.1.

7. *Select Workload*: The workload consists of a list of service requests to the system. For example, the workload for comparing several database systems may consist of a set of queries. Depending upon the evaluation technique chosen, the workload may be expressed in different forms. For analytical modeling, the workload is usually expressed as a probability of various requests. For simulation, one could use a trace of requests measured on a real system. For measurement, the workload may consist of user scripts to be executed on the systems. In all cases, it is essential that the workload be representative of the system usage

in real life. To produce representative workloads, one needs to measure and characterize the workload on existing systems. These and other issues related to workloads are discussed in Part II.

8. *Design Experiments*: Once you have a list of factors and their levels, you need to decide on a sequence of experiments that offer maximum information with minimal effort. In practice, it is useful to conduct an experiment in two phases. In the first phase, the number of factors may be large but the number of levels is small. The goal is to determine the relative effect of various factors. In most cases, this can be done with *fractional factorial experimental designs*, discussed in Part IV. In the second phase, the number of factors is reduced and the number of levels of those factors that have significant impact is increased.
9. *Analyze and Interpret Data*: It is important to recognize that the outcomes of measurements and simulations are random quantities in that the outcome would be different each time the experiment is repeated. In comparing two alternatives, it is necessary to take into account the variability of the results. Simply comparing the means can lead to inaccurate conclusions. The statistical techniques to compare two alternatives are described in Chapter 13.

   Interpreting the results of an analysis is a key part of the analyst's art. It must be understood that the analysis only produces results and not conclusions. The results provide the basis on which the analysts or decision makers can draw conclusions. When a number of analysts are given the same set of results, the conclusion drawn by each analyst may be different, as seen in Section 1.2.
10. *Present Results*: The final step of all performance projects is to communicate the results to other members of the decision-making team. It is important that the results be presented in a manner that is easily understood. This usually requires presenting the results in graphic form and without statistical jargon. The graphs should be appropriately scaled. The issue of correct graph plotting is discussed further in Chapter 10.

    Often at this point in the project the knowledge gained by the study may require the analyst to go back and reconsider some of the decisions made in the previous steps. For example, the analyst may want to redefine the system boundaries or include other factors and performance metrics that were not considered before. The complete project, therefore, consists of several cycles through the steps rather than a single sequential pass.

The steps for a performance evaluation study are summarized in Box 2.2 and illustrated in Case Study 2.1.

Case Study 2.1 Consider the problem of comparing remote pipes with remote procedure calls. In a procedure call, the calling program is blocked, control is passed to the called procedure along with a few parameters, and

Box 2.2 Steps for a Performance Evaluation Study

1. State the goals of the study and define the system boundaries.
2. List system services and possible outcomes.
3. Select performance metrics.
4. List system and workload parameters.
5. Select factors and their values.
6. Select evaluation techniques.
7. Select the workload.
8. Design the experiments.
9. Analyze and interpret the data.
10. Present the results. Start over, if necessary.

when the procedure is complete, the results as well as the control return to the calling program. A **remote procedure call** is an extension of this concept to a distributed computer system. A program on one computer system calls a procedure object on another system. The calling program waits until the procedure is complete and the result is returned. Remote pipes are also procedure like objects, but when called, the caller is not blocked. The execution of the pipe occurs concurrently with the continued execution of the caller. The results, if any, are later returned asynchronously.

The following project plan was written before starting the study.

1. *System Definition*: The goal of the case study is to compare the performance of applications using remote pipes to those of similar applications using remote procedure calls. The key component under study is the so-called channel. A **channel** can be either a procedure or a pipe. The system consists of two computers connected via a network as shown in Figure 2.1. The requests are sent via the channel from the client computer to the server computer. Only the subsets of the client

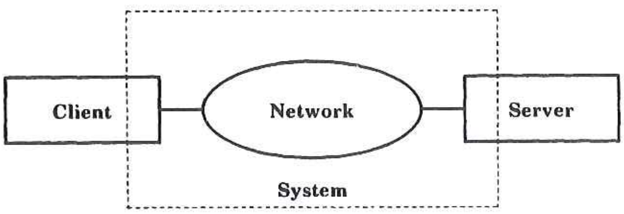

FIGURE 2.1 System definition for the study of remote procedure calls versus remote pipes.

and server computers that offer channel services is considered to be part of the system. The study will be conducted so that the effect of components outside the system is minimized.

2. *Services*: The services offered by the system are the two types of channel calls—remote procedure call and remote pipe. The resources used by the channel calls depend upon the number of parameters passed and the action required on those parameters. In this case study, data transfer is chosen as the application and the calls will be classified simply as small or large depending upon the amount of data to be transferred to the remote machine. In other words, the system offers only two services: small data transfer or large data transfer.
3. *Metrics*: Due to resource limitations, the errors and failures will not be studied. Thus, the study will be limited to correct operation only. For each service, the rate at which the service can be performed, the time taken for the service, and the resources consumed will be compared. The resources are the local computer (client), the remote computer (server), and the network link. This leads to the following performance metrics:
   (a) Elapsed time per call
   (b) Maximum call rate per unit of time, or equivalently, the time required to complete a block of n successive calls
   (c) Local CPU time per call
   (d) Remote CPU time per call
   (e) Number of bytes sent on the link per call
4. *Parameters:* The system parameters that affect the performance of a given application and data size are the following:
   (a) Speed of the local CPU
   (b) Speed of the remote CPU
   (c) Speed of the network
   (d) Operating system overhead for interfacing with the channels
   (e) Operating system overhead for interfacing with the networks
   (f) Reliability of the network affecting the number of retransmissions required

   The workload parameters that affect the performance are the following:
   (a) Time between successive calls
   (b) Number and sizes of the call parameters
   (c) Number and sizes of the results
   (d) Type of channel
   (e) Other loads on the local and remote CPUs
   (f) Other loads on the network

5. *Factors:* The key factors chosen for this study are the following:
   (a) Type of channel. Two types—remote pipes and remote procedure calls—will be compared.
   (b) Speed of the network. Two locations of the remote hosts will be used—short distance (in the campus) and long distance (across the country).
   (c) Sizes of the call parameters to be transferred. Two levels will be used—small and large.
   (d) Number n of consecutive calls. Eleven different values of n—$1, 2, 4, 8, 16, 32, \ldots, 512, 1024$—will be used.

   The factors have been selected based on resource availability and the interest of the sponsors. All other parameters will be fixed. Thus, the results will be valid only for the type of CPUs and operating systems used. The retransmissions due to network errors will be ignored (not included in the measurements). Experiments will be conducted when there is very little other load on the hosts and the network.
6. *Evaluation Technique:* Since prototypes of both types of channels have already been implemented, measurements will be used for evaluation. Analytical modeling will be used to justify the consistency of measured values for different parameters.
7. *Workload:* The workload will consist of a synthetic program generating the specified types of channel requests. This program will also monitor the resources consumed and log the measured results. Null channel requests with no actual work but with monitoring and logging activated will be used to determine the resources consumed in monitoring and logging.
8. *Experimental Design:* A full factorial experimental design with $2^3 \times 11 = 88$ experiments will be used for the initial study. The factorial experimental designs are explained in Chapter 16.
9. *Data Analysis:* Analysis of Variance (explained in Section 20.5) will be used to quantify the effects of the first three factors and regression (explained in Chapter 14) will be used to quantify the effects of the number n of successive calls.
10. *Data Presentation:* The final results will be plotted as a function of the block size n.

This case study was completed successfully and reported by Glasser (1987). □

EXERCISES

2.1 From published literature, select an article or a report that presents results of a performance evaluation study. Make a list of good and bad points of the study. What would you do different if you were asked to repeat the study.

2.2 Choose a system for performance study. Briefly describe the system and list

a. Services

b. Performance metrics

c. System parameters

d. Workload parameters

e. Factors and their ranges

f. Evaluation technique

g. Workload

Justify your choices.

Suggestion: Each student should select a different system such as a network, database, processor, and so on, and then present the solution to the class.