Network management software usability test design and implementation

by L. C. Percival
S. K. Johnson

The approach used at one of IBM's development sites for usability testing is somewhat different from methods used elsewhere. The approach was developed specifically for testing of software communications products as one aspect of the System Usability Process. The test design and implementation are described.

An important mission of the IBM Communications Programming Center in Research Triangle Park (RTP), North Carolina, is to develop communications software products that support computer networks. These products are involved in the performance of a variety of tasks necessary to maintain the integrity and performance of the network. These tasks are generally called network configuration definition, performance analysis, and problem determination and resolution.

Beginning in 1983, several major usability tests have been conducted in the RTP area. Users have been asked to perform specific examples of the tasks listed above with network management products such as the Network Communications Control Facility (NCCF), Network Logical Data Manager (NLDM), Network Problem Determination Application (NPDA), Virtual Telecommunications Access Method (VTAM), and VTAM Node Control Application (VNCA). Because the computer networks for which these products have been designed have become quite complex over a period of time, the products have also become complex. This complexity results partly from the attempt to support as wide a range of computer and communications hardware and software as possible, and partly from a lack of emphasis on the importance of usability at the time the products were designed.

The usability testing described in this paper is one aspect of the System Usability Process for Network Management Products.[1] With this process the RTP Communications Programming Center can design and develop products that make it relatively easy for product users to plan, build, operate, and support computer networks. The emphasis in network management usability testing has been on testing the entire system, i.e., testing several of the products as a group. This is particularly true for the set including NCCF, NPDA, NLDM, and VNCA, which are often used together to do network problem determination. In the tests, many usability problems have been found, documented, fixed, and retested to verify the solutions. The end result

of this test process is, of course, a more usable set of IBM network management products in the future. This paper describes some of the testing procedures used, and presents a brief summary of the results obtained with these tests. Differences between common practice in IBM product usability testing and the present procedures, which exist primarily because of the complexity of the network management function, are indicated to aid those planning to execute usability tests in the future.

Each major test is a joint effort involving several groups.

The approach

Usability testing in the RTP Programming Center, and elsewhere within IBM, involves the use of simulators, live code, and paper exercises as test instruments. In testing of network management products, unlike some other IBM products, rather specialized persons serve as subjects or test participants. The pros and cons of whether to use experienced or naive persons in system evaluation are presented in Reference 2. The primary reason for using experienced persons in testing the usability of network management products is that many aspects of network operation are quite complicated and technical. A person selected at random from the general population would stand little or no chance of performing some of the more complicated tasks correctly without considerable training. One could not expect that the average person would even be familiar with the names of the hardware generally found in computer networks, let alone its characteristics and idiosyncrasies. Therefore, current users of generally available versions of network management products are brought to RTP to evaluate future versions.

These users perform tasks embedded in realistic scenarios that have been developed in conjunction with persons from IBM Marketing who support the network products. Average performance across a relevant user class is then evaluated against usability criteria[3] developed for each scenario. Generally, when performance does not meet the established criteria, a usability problem is suspected as the cause. In such cases further analysis is conducted to determine the cause of the missed criteria and to document the existence of any usability problems discovered by a detailed analysis of the data gathered. The documented problems and often the recommended solutions are then submitted to the appropriate development group for resolution.

The people involved. In addition to the test subjects who serve as the users in evaluating the usability of products, various groups are involved in the testing.

Test sponsors. Each major test, like the one described later, is a joint effort involving several planning, development, evaluation, and assurance groups.[4] Because of the scale of the test, no single group has the hardware or human resources to conduct it alone. In this combined effort each group does part of the work, and since the groups depend on one another, successful execution of the test requires cooperation and coordination among all the groups.

Market Analysis and Support has traditionally had responsibility for overall coordination of the test. This department, part of the planning staff in the Programming Center, has been responsible for preparing nondisclosure requests to allow customer users to be brought in as test participants, securing any needed release forms for persons participating in the test, and ensuring a valid evaluation.

Human Factors provides expertise in the area of the user interface, as well as in test methodology and data analysis. Because of their background in applied research methods, members of this group ensure that the test data are collected under valid conditions by assisting in scenario development, reviewing the criteria, training test personnel, and helping design the test. They also lead the data analysis effort.

Development identifies the parts of a product where the end-user interface should be tested. Developers also write scenarios that embody the tasks that

customers will perform in production environments. In addition, they are responsible for implementing any simulations required for their product, determining the information to be gathered in questionnaires following scenarios, helping to analyze the data, and fixing problems found in their product.

In a major test, the staff consists of 15 to 20 people.

Assurance has the responsibility within IBM of ensuring that development and test areas correctly follow procedures and guidelines that allow a product to progress from planning to announcement, and also ensuring that the product conforms to quality standards. In the testing, this responsibility includes reviewing the test plan, aiding in data analysis, and aiding in problem follow-up and resolution.

Information Development is usually deeply involved in testing activities because it is the group that develops the on-line information and the manuals that are part of the product. For this testing, major responsibilities of the group are to help write scenarios for the use of the information and to fix any problems found in that information. Since members of this group are usually very interested in observing people use the information, several observers, or monitors, are usually provided by this group.

The *IBM Raleigh International Systems Center* brings IBM World Trade personnel to Raleigh, North Carolina, to gain practical experience on new releases of IBM communication products. These people help write valid test scenarios, install and support the latest development level of the product, and often provide the test environment.

Test staff roles. In a major test, the staff consists of 15 to 20 people and is composed of individuals from the groups sponsoring the test. The staff is somewhat larger than is usually necessary for usability evaluations for two primary reasons. One is that data are obtained from four participants simultaneously, so that the required number of monitors and orchestrators (see below), with the necessary back-up personnel, is large. The second is that the amount of support required, especially in a live code test, is large because of the technical requirements of testing these products. The monitors and orchestrators are trained by Human Factors, World Trade, and Development personnel. The duties of each of the staff are now described.

The *Overall Coordinator* has the responsibility of ensuring that the test runs according to plan. The coordinator contacts the customer representatives to see if there is an interest among current users in participating, makes travel arrangements for the participants, writes the test plan, and gains approvals for it from the development and assurance areas. The coordinator also secures any release forms from the participants, gathers a test staff from departments within the programming center, and then monitors the test as it progresses to make sure everything is running smoothly. He or she is responsible for dealing with any unusual situations that arise during the test, but otherwise is not generally involved in test execution. Once the test is complete, the coordinator must see to it that the usability problems are fixed by the development areas, and then write a final report on the results of the test. Typically, this position is filled by a person from the Market Analysis and Support Department.

Monitors watch and record what each participant is doing. There are one or two monitors per participant, depending on the type of test. Their primary responsibilities are to guide participants through the testing process, to observe them in the scenarios, and to log important test data. Monitors and orchestrators (described below) are solicited on a voluntary basis from the groups sponsoring the test. There have been no problems in obtaining monitors from product and information development, assurance, and other related areas because it is recognized that the experience gained in working with customers and observing them working with the products is very valuable.

Orchestrators provide coordination and generally control the execution of the test. Each scenario is

a different task performed with these products. The orchestrators must prepare the conditions for each scenario and then let the monitor know, via terminal messages, that the participant can begin the scenario. For example, if the scenario requires the participant to identify a network resource as inactive, the orchestrator must first cause this condition to occur, verify it, and then inform the monitor that the condition is in effect. The orchestrators must be aware of what each participant is doing at all times in order to conduct an effective test.

Technical Support Personnel participate in the monitor training sessions to teach the monitors and orchestrators how to use the products. During the test, they are on call in case problems arise with the products or simulators and to answer any questions about the products that the monitors cannot answer. They are usually members of the product development areas or IBM's Raleigh International Systems Center.

Methodology

Product users/test participants. With the help of the IBM marketing divisions, customer accounts that have current or potential communications software users are asked to participate in the usability test. They are informed that confidentiality and nondisclosure agreements must be signed regarding unannounced products. Other release forms relating to their willingness to participate in this type of testing and giving us permission to videotape the activities of participants during the test are also signed by the participants prior to the test.

A mix of system programmers, network operators, line technicians, and help-desk operators are solicited from each customer account, depending on the products being tested. The customers are informed of the type of person needed for the test, and they select which of their personnel will participate.[5] Four people are requested per account, with a total of 20 to 30 participants in each test.[6] Even though it is somewhat costly to bring these people to Raleigh for testing, the cost is more than justified in the interest of obtaining people with the appropriate background. An alternative to this approach, that of bringing in people from employment agencies, is not viable because of the amount of training such individuals would require even to be able to participate in certain aspects of the test. In addition, the network management development community takes much more notice of problems experienced by the appropriate user class than of problems experienced by persons with little or no network management background.

Techniques and tools. The particular tools used to test a product depend on how far along a product is in the development cycle and the resources available for tool development. Several techniques are routinely used. These include

- A paper and pencil mock-up of the user interface
- A simulation of the user interface which for all practical purposes looks and acts like the real thing, but which has no function behind the interface
- An early working prototype of a product that includes some function and the user interface
- Live code, which is simply a running version of the product in a live system

Several distinct types of data are captured to give a complete picture of the usability of a product.

Which of these tools is used depends very much on the purpose of the test, the resources available for test tool development, the point the product has reached in the development cycle, and the type of product being developed. The emphasis in network management product usability testing has been on simulation, early functional prototypes, and live code testing. The test described in this paper is a live code test, but the general methods employed in using any of these test tools are quite similar.

Data capture. Several distinct types of data are captured to give a complete picture of the usability of a product. Among these are performance data

such as the time taken to perform a task and the number of errors made along the way, comments about the implementation, the path taken to solve a problem, and the amount and type of assistance required to complete the task.

Test participants typically have one, and sometimes two, monitors working with them during the test. The monitor tells the participant what the next task is to be, observes how the participant performs the task, provides assistance wherever necessary, and records observations of the participant performing the task. In large-scale tests involving simultaneous data collection from several participants, monitors also communicate with a person orchestrating the overall test.

Monitor log. Monitors typically record their observations with an automated comment-logging program. This program has a number of predefined observational categories, so that many times the monitor simply presses a single key when a given action is taken. This key causes a time-stamped entry to be placed in the log that contains that observational category, the time, and any further comment the monitor wishes to record. An example of a category might be "Consult Operations Manual." After pressing the appropriate key for this category, the monitor may also enter the particular part of the manual, and whether the participant was able to find the information successfully. Much of the data from this log can be automatically analyzed and statistically summarized; text comments can be sorted and grouped in various ways for analysis.

Videotape. Another method of capturing data is on videotape. This method is excellent for capturing the test participants' reactions to material presented to them on their terminals and in manuals as they attempt to work through a scenario. It makes for a very dramatic presentation of user reaction to the product and helps to build a strong case for the existence of a given problem.

Videotape can also be used to capture the actions of the participants taken through the system, that is, to record their interactions with the system by videotaping the display on which they are working. There are, however, technical limitations that prevent the entire screen from being legible upon playback of the tape. In addition, analysis of the performance and comments of participants captured on videotape is very time-consuming. Therefore, it is not practical to use videotape for large-scale data collection in this type of testing.

Questionnaires. Questionnaires are used to obtain background information on the participants and opinions about the product and various aspects of the test situation. With regard to the background information, the participants are asked to provide some biographical information, such as job title, years of experience, and primary tasks. Such data allow us to assign each participant to one of the sixteen jobs found in the case study effort.[7]

Following each scenario, the participants are asked to fill out a questionnaire on several key items encountered during that scenario. In addition, they are asked some general questions such as whether this type of problem is encountered by persons of the participant's job type.

These questionnaires are typically placed in an on-line format so that the data can be captured and automatically analyzed without manual entry. Of course, paper versions are also provided should the participant prefer that format.

Trace of simulation. When a simulation of the user interface is developed solely for purposes of testing usability, trace facilities can be built into the simulator to capture the participant input along with the system output. This method is not a keystroke level of data capture like that used in other IBM locations,[8,9] but rather a command-level capture.[10] This trace can be time-stamped and recorded for later analysis. Items of particular interest that can be obtained from this trace are (1) the path taken in a scenario, (2) the screens which are called on numerous occasions or on which participants spend an inordinate amount of time, (3) the screens where help is most often requested, (4) the screens associated with command or other errors, and (5) the time spent in performing the task.

Monitor training

Before each test begins, Human Factors personnel head a two-week period of intensive monitor training. The first half-day of monitor training is spent reviewing testing ethics. After that there are two main items to be learned by the monitors: data logging and the scenarios.

During the test, monitors dynamically record observations and comments using a special Personal Computer program designed for this purpose. The material is time-stamped, stored in a data set, and analyzed at a later time. One part of monitor training is devoted to teaching the monitors to use this program and to record errors, comments, and assistances consistently among them.

The second part of monitor training teaches the monitors how a participant is expected to execute each scenario. Development or World Trade personnel demonstrate how the product could be used to complete the tasks required by the scenario; monitors then practice among themselves to perform the tasks and explore the product. This training becomes very valuable when the monitors must observe participants performing the tasks. Through practice and exploration they learn to quickly judge when a participant is going off the correct path. In addition they become proficient at logging, because while one monitor is performing the task, another monitor practices recording the activities with the data logging tool.

Pilot testing. Before the customer participants are brought in, a pilot test is performed with internal personnel. This test is used to iron out any difficulties with procedures and test materials, such as participant and monitor instructions and questionnaires, and to identify possible usability problem areas on which to focus. It is typically in the pilot test that the major problems are identified. The full-blown test that follows verifies that these problems are genuine and that they are experienced by customer product users under more controlled conditions. Pilot participants are usually solicited from the IBM RTP Information Systems group because their work closely matches the job characteristics of customers. Typically, the pilot test lasts about one week.

Test description

The tests have usually dealt with a relatively small but complex network that contains a mix of IBM hardware. Figure 1 presents the piece of a network at the Raleigh International Systems Center used in a recent test. It consisted of two large, mainframe "host" processors (HOST11 and HOST21), three communication controllers/NCPs (N249F4G, N139F4D, and N14BF3J), one IBM 3710 communication controller/protocol converter (P13036X), six local and six remote modems with associated lines, SNA cluster controllers (P13010C, P13064F, P13012C, and P14022K), a binary synchronous communication (BSC) controller (P13036C), an IBM 4700 controller (P14A1C2), and associated workstations or terminals.[11]

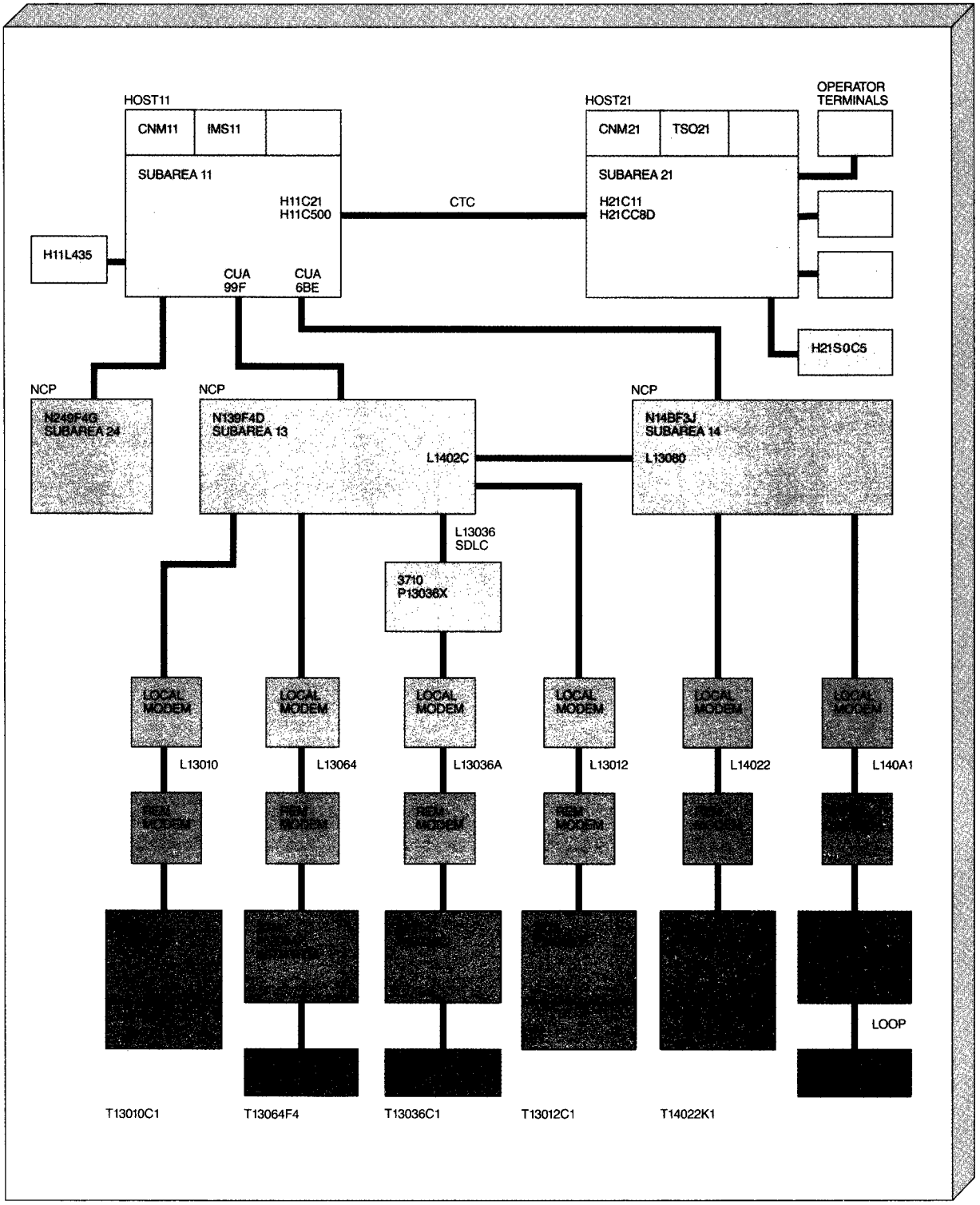
Test overview. The first morning of a two-day test is an introductory and warm-up session for the participants. The overall coordinator for the test gives a briefing on what will be happening during the test. A schedule of activities and a test procedure review are the main topics discussed. Then the participants are given a structured warm-up session in which they browse manuals, exercise certain product functions, and browse on-line help. Following this phase is the actual beginning of scenarios which typically involve installation or operation tasks.

The participants are asked to try to complete each scenario and "think aloud" while doing so, in a process called the Thinking Aloud Method.[12] Data logging procedures can account for some of the time that participants use in making oral statements by stopping the elapsed time clock when participants start making extended comments about anything. The monitors are recording all the participant activities and any system situation which might affect the validity of a scenario for a particular participant. When the scenario is completed, a questionnaire is filled out by the participant to obtain opinions on certain items within that scenario.

A debriefing session takes place at the end of the test. This session is an open discussion between the participants and the staff members to find out what the participants thought about the test and to record any additional comments they wish to make about the products.

Test scenarios. In general, the scenarios are written to make them resemble the way in which product users perform the same tasks in their normal jobs. This portrayal is done by working with development and IBM marketing support organizations to establish typical scenarios.[13-16] Each scenario includes a description of the way in which the product users would normally receive the input for the scenario, what the users normally would do in that scenario, and what they would do when finished with it. In a typical operational problem, a help-

Figure 1 Network used in live code usability testing at Raleigh International Systems Center



desk operator might receive a call from a terminal user complaining of poor response time. The operator would take down some information and begin to investigate the problem. When the problem was solved and fixed, or passed to technical support for fixing, or not solved and passed to a higher level of problem determination, the operator would call the terminal user back to say that the problem had been fixed, or was solved and was in the process of being fixed, or was being investigated further.

Significant effort is expended to ensure that the test scenarios are valid.

Significant effort is expended to ensure that the scenarios are valid. The scenarios used in testing have consistently received high marks from the test participants on validity for a variety of different customer environments.

There are three general classes of test problems that users are asked to solve. They include

1. *Performance Analysis*—Participants are asked to analyze some aspect of the performance of the network, such as utilization of a selected resource type. For example, the participant might be asked to determine to what extent each of the NCPs in Figure 1 is utilized so that additional terminals can be added to one of them.
2. *Problem Determination*—Participants are asked to solve and determine a fix for a problem, the symptoms of which are presented by a simulated user phone call or system alert. For example, the participant might be asked to determine why a terminal is not working. (See the sample scenario below.)
3. *Product Installation*—Participants are asked to perform some of the steps involved in the installation of the product for a specified network configuration.

Procedures. The problem determination scenarios that are run in a live code test such as those run at the Raleigh International Systems Center are something between a laboratory experiment and a field study. The site for the testing resembles a field setting more than a laboratory, but the ability to control a number of relevant conditions is more closely associated with a laboratory study. This setup is typical of what Parsons[17] has called a "Man-Machine System Experiment." Key areas of control necessary to run a valid usability test of network management software are (1) the ability to reliably cause the same problem to occur and present the same symptoms to each of the test participants, (2) the ability to isolate network problems so that a single participant works on a given piece of the network, preventing any interaction between the actions taken by the participants, and (3) a standard script for each scenario containing the several key pieces of information that are to be presented by the orchestrator to the participant for the scenario.

The monitors and participant generally sit in the same standard-size office with the monitors positioned to see over the shoulder of the participant. Although this is somewhat obtrusive, the participants are used to working with a group of people and have indicated that it is less stressful than working alone in a room with several video cameras trained on them and with a large mirrored window behind which they know not what is occurring. The participant has a terminal, as does one of the monitors. The participant uses his or her terminal to interact with the product. The monitor uses the terminal to communicate with the orchestrator when necessary. The remaining monitor has a Personal Computer used for logging observations.

For each scenario, a detailed description is prepared for the monitors. This description includes the starting and ending conditions, the actual cause of the problem, the expected path that the participant is likely to take to solve the problem, the panels or screens that will be encountered, and an accompanying description of the more technical aspects of these panels. From this description and their own experience with each scenario, the monitors can tell if the users get too far off track in trying to solve the problem and can try to redirect them.[18] Since the participant often tells the monitor why certain steps were taken, the monitor can

quickly determine whether the participant is moving in the correct direction. A list of assistance items ranging from general to specific is provided for the monitor to give to the participant. When the monitor gives assistance of this type, the assistance and the conditions necessitating it are recorded for later analysis.

Sample scenario. In this scenario, a problem is caused by noise inserted on a line between a pair of IBM modems depicted in Figure 1. This noise is produced by a Bradley Noise Generator and simulates a noisy telephone line (line L13012 in the figure). Parameters on the noise generator are adjusted to a set of predefined levels so as to induce the same level of poor line quality each time. This poor line quality results in noticeably poor response time on terminal T13012C1. After a few minutes of using the terminal to generate traffic on the line, the orchestrator calls the participant, simulating a user complaining about poor response time on that terminal and giving only the information that it is much worse than usual, along with the terminal identifier. The participant usually writes down the identifier and hangs up.

Any number of courses of action might be taken to attempt to solve this problem. One of the first things that might come to mind is to use NLDM to examine the response time for this terminal. However, looking at the network diagram quickly reveals that this will not be possible because the cluster controller to which this terminal is attached (P13012C) does not have the necessary Response Time Monitor (RTM) feature. NLDM will tell the participant that no response time data exist for the user's terminal. A different possibility that is often examined first is that of some type of physical problem in the network, in this case somewhere between communication controller/NCP N139F4D and cluster controller P13012C. To check out this possibility, the participant uses NPDA. By running the appropriate test between the NCP and the controller that are using the IBM modems, the participant is able to determine that the line quality is bad between the modems.

At this point the participant calls the terminal user (the orchestrator) back to let him know that the problem is with the line. Under real conditions the problem would then usually be passed to a person acting as an interface to the telephone company. Following the telephone call to the orchestrator, the participant is asked to fill out a short questionnaire on the scenario.

Usability problem identification

After the collection of data in the test, it is necessary to analyze it with the specific goals of (1) evaluating the usability of the product against the established criteria, (2) identifying and documenting problems, and (3) recommending an appropriate fix for any problems found. These steps are relatively standard in product usability evaluation, though the number of different groups involved may be unique to major tests of network management products in the RTP Programming Center.

Usability committee. To provide a fair analysis of the data, this committee comprises persons from a number of groups, including Product Development, Quality Assurance, Human Factors, and sometimes others. The product developers know the technical details about the products and, in general, the feasibility of proposed fixes. Assurance makes certain that the data are fairly evaluated and that product usability criteria are met. Human Factors usually coordinates the committee activities and provides leadership in the appropriate statistical analysis of test data.

Analysis. The analysis depends to some degree on the type of tool used in the test. Maximum data are provided by simulation with built-in trace facilities and the accompanying monitor logs and questionnaires. The other test instruments all provide less but somewhat equivalent data. In keeping with the live code test example, the present discussion therefore assumes that no trace file is available and that the analysis is based on the logs and questionnaires.

Statistical description of results. The statistical analysis has two chief goals: (1) Provide the necessary analysis for evaluation against the usability criteria and (2) Identify problem areas for further, more detailed analysis. Data such as elapsed time, number of assists, percent of participants successfully completing the tasks, and ratings on specified scales are quickly analyzed. Since there are usually a relatively small number of observations for any given variable, care is taken to provide the appropriate analysis and to resist providing one where it is not justified. Since the number of participants within any given job category is usually small,

with a total usually around 20, the data often cannot be broken down by job and still be statistically reliable. Instead, an average or percentage for a "mix" of network management jobs is often provided. If any user type deviated markedly from this average performance, a statement to this effect is made. This observation is especially important where a particular product feature is aimed at a specific job.

In the test plan the conditions of product success and failure are clearly specified.

Criteria evaluation. In the test plan developed prior to the test execution, specific criteria are stated for a number of measures and tasks. For example, to diagnose a problem of poor response time caused by noise on the line, criteria might include (1) 80 percent of participants completing a correct diagnosis of poor response time caused by noise on the line, (2) an average successful completion time of 10 minutes, and (3) an average of two occasions for external assistance. In testing of network management products, the criteria are based on empirical values from previous tests. The order of importance for criteria can be based on data collected in case studies such as those discussed by Gottschalk.[1]

In the test plan the conditions of product success and failure are clearly specified. These conditions are often separated into categories such as Exceeds or Meets Criterion and Moderately or Severely Misses Criterion to give a clearer picture of how good or bad the results were. For each test a matrix is drawn up which has as its rows the criteria, ordered from most to least important, and has as its columns the categories of success and failure. When the test has been run and the data analyzed, this matrix can be used to determine how the product fared in a number of representative tasks. Items taken into consideration include the width of the confidence interval around the observed value,[19] the category of success or failure, and the importance of each of the criteria.

The process of criteria evaluation provides a picture of the usability of the product in specific areas. It is a key factor in assuring the usability of the product being tested; it can also point to specific problems where the criteria are not met. For example, if criteria for completion rate and time have been met, but the assistance criterion has not, this suggests that problems exist in one or all of the following areas: panel wording, abbreviations, or on-line help associated with that task.

Comments and observations. For any type of test, one key source of data is the logs kept by monitors or observers. These logs contain not only items such as elapsed time and assists, but also observations made by the monitor as he or she looks over the shoulder of the participant, and comments made by the participant as he or she tries to perform the requested task. Participants are asked to provide comments on any item they think is especially good or bad, and to provide insights into how they are trying to perform the task by thinking aloud whenever possible. The monitor tries to take down the significant aspects of these thoughts in relation to what the participant is doing with the product and the documentation at the time.

Each person on the committee receives a copy of the log and is asked to go over it in detail, noting significant events and comments. Then the committee meets to produce a single log containing the observations from all members. This group log becomes the official test log and is used to identify or provide further insight into usability problems. It can also be used as a source of documentation for problems derived solely from participant comments.

Committee review. There are several general steps in this procedure:

1. The committee reviews the statistical analysis and the log to get a general feel for the problem areas as described above.
2. Any member of the committee may generate formal problems to be brought before the committee. Generating a problem involves more

detailed analysis of the available data and a formal statement of the problem, its cause if known, its severity, and a recommended fix.
3. The committee meets to discuss each problem and revise any of the formal aspects, especially the recommended fix and the severity of the problem. An informal vote is taken on each aspect of the documented problem.
4. For each problem accepted by the committee, an organization, usually Product Design or Development, is identified as the appropriate recipient of the problem. The goal of this step is to get the problem into the formal problem-tracking system for the product.
5. The last step sometimes involves a follow-on verification test and sometimes an individual commitment by the committee members. This commitment is the examination of the product specification and working interface to ensure that the fixes make their way into the product as specified.

Problem tracking and correction

When the data analysis is complete, problem reports are written and given to the appropriate development groups for resolution. In the past, a standardized form called a Usability Problem Report (UPR) has been used. The UPR describes the problem and its apparent cause, indicates the seriousness of the problem, and suggests a solution for it. Recently usability problems have been added to the other types of problems tracked by an on-line problem-reporting system used by the development and test areas in the Programming Center. This system allows the problem reporter to enter basically the same information as that on a UPR, but it is more closely tracked within development than the UPRs had been.

For each valid problem encountered, the development group is expected to commit to doing one of three things: (1) Correct the problem in the current release of the product; (2) Correct the problem in the next release of the product; (3) Correct the problem in the strategic replacement for the product.

The development group is expected to take the first of these alternatives when possible. The second is to be taken when the first is not possible and when the second can be taken without completely redoing the product. The third alternative is to be taken when neither the first nor the second is feasible.

Fix verification for problems to be fixed in a subsequent release or in a strategic replacement will be made via the usability test for the subsequent release or replacement product.

The test exit criterion requires (1) that all problems are satisfactorily answered by the appropriate development groups with the appropriate fix alternative indicated for each, and (2) that a committed plan of action is in place to resolve each problem.

Overview of results

Although the implementations of many network management functions have received favorable comments from test participants, usability testing almost always focuses on problems that exist in the implementation. Identified usability problems are assigned to one of three categories depending on their seriousness. By assigning each problem a severity level, Development is able to focus resources on the problems on a priority basis. The working definitions for the severity levels are as follows:

- Severity 2: Severe usability problem. This problem inhibits the ability of the intended product users to perform an appropriate task to such an extent that a significant number of these users are unable to perform the task at all.
- Severity 3: Major usability problem. This problem causes users to expend significantly more effort than they should reasonably be expected to in using the product to perform an assigned task, but is not so severe as to cause a significant percentage of users to fail to perform the task at all.
- Severity 4: Minor usability problem. This problem causes users some difficulty in performing an assigned task, but is more of an inconvenience than a major inhibitor with respect to satisfactory performance of the task.

In the on-line problem-reporting system now being used, there are no Severity 1 usability problems. This classification is reserved for problems with functional code.

A number of valid usability problems have been identified in the major tests completed to date.

The majority of these have been severities 3 and 4, but there have been a number of severity 2 also. Of the problems identified, about two thirds were fixed in the release being tested, with the rest to be fixed in later releases.

The types of problems have fallen into several categories: (1) those that are very specific to the product being tested, (2) those that exist because a number of products may be used to solve certain network problems, and (3) general problems in user interface design. Examples from the first category range from very specific items such as the wording or abbreviations used on certain panels to general items such as panel structure and flow and the need for greater direction and data interpretation in problem determination. In these circumstances, the specific problems are the easiest to fix and at the same time the least important. It is the general items, particularly the need for direction and data interpretation in network problem determination, that are the most difficult to solve. Examples from the second category include inconsistencies in areas such as command syntax, program function keys, and the use of color. It is quite important that these inconsistencies be eliminated if users are required to work with multiple products to solve a given problem. Examples from the third category include items such as the role of color in general panel structure and in the presentation of qualitative data, and the use of graphics to present information. Solutions for these general problems in user interface design are being sought both at RTP and other sites within IBM.

Conclusions

The usability testing conducted on the network management products at RTP and described in this paper is somewhat different from that conducted at many other IBM locations. Indeed, the major tests described here are quite different from much of the day-to-day evaluation done at RTP. The number of groups involved, the cost of obtaining experienced personnel from IBM customers, and the use of live networks are somewhat unique to usability testing of network management software at IBM RTP. Yet this willingness to commit the time, money, and personnel to these efforts will be responsible, along with efforts such as the case study described by Gottschalk,[1] for producing significant improvements to the user interface for IBM network management products.

Cited references and notes

1. K. D. Gottschalk, "The System Usability Process for Network Management Products," *IBM Systems Journal* **25**, No. 1, 83–91 (1986, this issue).
2. J. S. Kidd and K. M. Michels, "Staff development in systems research techniques," *Man-Machine Systems Experiments*, H. M. Parsons, Editor, Johns Hopkins Press, Baltimore (1972).
3. Typical criteria include time and amount of external assistance required to complete the task.
4. This type of joint effort in usability testing is somewhat unique both here and elsewhere within IBM. More common, small-scale evaluations are also conducted at RTP on an ongoing basis by Human Factors and Development.
5. Because of the case study effort discussed by Gottschalk in another paper in this issue[1] and reported fully by Beith, Moore, Pendley, and Percival,[7] the participants may be classified into one of the general jobs within the industry based on empirical findings.
6. Although this may seem like a large number of participants, it should be remembered that this number is made up of several different user classes. It is sometimes important to examine performance for more than one user class. Typically, five persons of a given user class will perform a given scenario.
7. B. H. Beith, G. C. Moore, W. L. Pendley, and L. C. Percival, *Users of IBM Communication Software: A Survey of Customer Jobs, Tasks, and Networks*, IBM Corporation, P.O. Box 12195, Research Triangle Park, NC 27709 (1986); available through the authors.
8. I. A. Clark, "Software simulation as a tool for usable product design," *IBM Systems Journal* **20**, No. 3, 272–293 (1981).
9. A. S. Neal and R. M. Simons, "Playback: A method for evaluating the usability of software and its documentation," *IBM Systems Journal* **23**, No. 1, 82–96 (1984).
10. The decision not to capture keystroke data was made because the command-level data were considered (1) to be sufficient for a meaningful analysis and (2) a great deal less expensive to capture (given development time) than the keystroke data.
11. The network pictured is actually a subset of a larger one. Test activities are restricted to this network so that users of the larger network will not be unnecessarily disturbed.
12. C. Lewis, *The Thinking Aloud Method in Interface Evaluation*, Research Report RC-9265, IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598 (1982).
13. Some of this information is documented in technical papers written at the Raleigh International Systems Center. For examples, see References 14, 15, and 16.
14. I. McGregor, G. Matchett, and H. P. Dueholm, *Communication Network Management/Using the CNM Tools*, Raleigh International Systems Center Technical Paper, GG24-1561, IBM Corporation (1982); available through IBM branch offices.
15. H. J. Liberty, Jr., and J. A. Gabor, *SNA Problem Determination Guide/ACF R3 Volume 1*, Raleigh International Systems Center Technical Paper, GG24-1514-1, IBM Corporation (1984); available through IBM branch offices.
16. H. J. Liberty, Jr., and J. A. Gabor, *SNA Problem Determination Guide/ACF R3 Volume 2*, Raleigh International Systems Center Technical Paper, GG24-1523-1, IBM Corporation (1984); available through IBM branch offices.

17. H. M. Parsons, *Man-Machine System Experiments*, Johns Hopkins Press, Baltimore (1972).
18. Although a maximum time for each scenario is provided for the monitor, the decision about when and where to intervene is a judgment that the monitor must make. Since the type and circumstances necessitating the assistance are logged, the analysis team can make a decision about whether the assistance given invalidates the data for any given scenario.
19. W. L. Hays, *Statistics*, Holt, Rinehart, and Winston, New York (1981).

Lynn C. Percival *IBM Communication Products Division, P.O. Box 12195, Research Triangle Park, North Carolina 27709.* Dr. Percival received the doctorate in human perception and performance from the University of Louisville in 1982. He joined IBM at Raleigh in 1983 after a National Research Council Postdoctoral Fellowship at the Naval Aerospace Medical Research Lab in Pensacola, Florida. He has worked for several years in the design and evaluation of the user interface for network management and other software systems. For the last two years he has been involved in all phases of a large-scale study of users of IBM communication products. He has received excellence awards for this study and for his involvement in usability testing of network management products.

Susan K. Johnson *IBM Communication Products Division, P.O. Box 12195, Research Triangle Park, North Carolina 27709.* Ms. Johnson joined IBM in 1983 as a programmer involved in system testing of the IBM 3710 Network Controller. For the past year she has been involved in usability testing of the network management products. She has recently become a member of the Office Communications Services' field support group in Raleigh. Ms. Johnson received a B.S. in computer science from the University of Florida in 1982.

Reprint Order No. G321-5264.