THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 61, No. 6, July-August 1982
Printed in U.S.A.

Automated Repair Service Bureau:

The Front-End System

By S. G. CHAPPELL, F. H. HENIG, and D. S. WATSON

(Manuscript received June 17, 1981)

The primary role of the Loop Maintenance Operations System front-end computer is to help the Repair Service Bureau personnel track and repair troubles reported on telephone services by our customers. Each customer trouble report is entered into the system and its status is updated at each step toward completing the repair. Management reports are generated that warn of overload conditions and potential degradation of repair service.

I. INTRODUCTION

The components of the Automated Repair Service Bureau (ARSB) described throughout this volume serve four major functions. These are:

(*i*) maintaining a customer line record data base so that repair personnel have up-to-date information about the facilities being repaired,

(*ii*) recording and tracking troubles reported on telephone equipment from the time the trouble is reported until the time it is cleared and closed out,

(*iii*) testing and analyzing the condition of customer loops, and

(*iv*) analyzing closed trouble report data to aid in managing the repair process.

The first and last of the above functions are handled by the Loop Maintenance Operations System (LMOS) host,[1,2,3] where the power of a large main frame computer and the availability of large amounts of disk storage can be used to advantage. The third function—automated loop testing—is performed by the Mechanized Loop Test (MLT) system.[4]

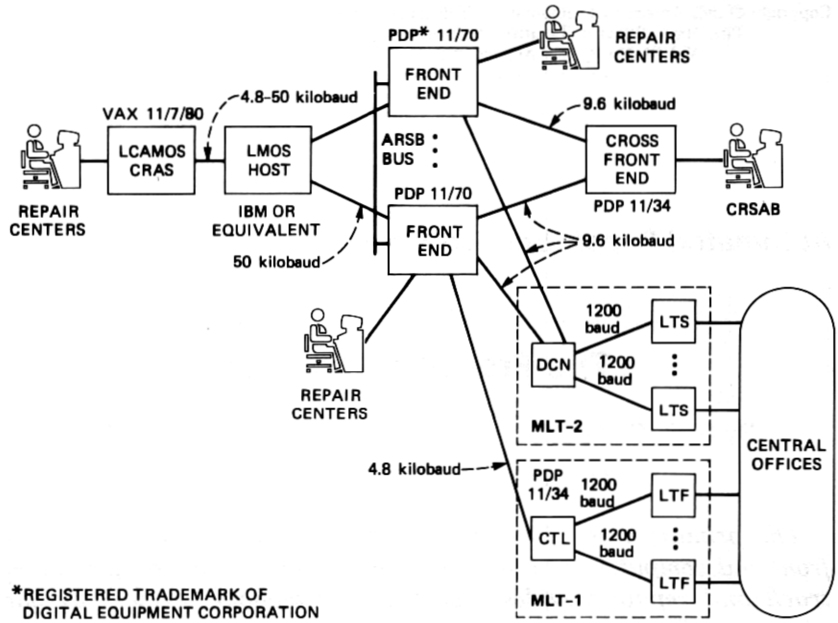

Fig. 1—Automated Repair Service Bureau—an example.

The second function is the topic of this paper and is the role of the LMOS front end (FE), a transaction-oriented tracking system designed to record troubles and maintain information on their status until the customer is satisfied that the problem has been corrected.

In addition, the LMOS FE is a communications handler. It serves as the primary user interface to the ARSB, providing access to both the host and MLT, in addition to the FE itself.

This paper describes the major capabilities of the FE, its role in the distributed ARSB, and its continuing evolution.

II. HARDWARE OVERVIEW

A typical LMOS system configuration is shown in Fig. 1. Figure 1 also depicts LMOS interfaces to other systems included in the ARSB and described elsewhere in this issue.[4,5] The LMOS system consists of a large IBM or IBM-compatible host (370 or 303X class) connected to as many as ten PDP* 11/70 LMOS FEs by 50-kilobaud data links. The interface between the two types of computers is well defined: the FE looks like a terminal controller to the host.

Access to FEs is provided via synchronous display terminals and

* Registered trademark of Digital Equipment Corporation.

printers (e.g., Teletype* 40/4 keyboard displays and printers). Each FE can support up to 512 such devices on up to forty-eight 4.8- or 9.6-kilobaud data links. These terminals and printers provide access to the system from the Repair Service Bureaus (RSBs), the Centralized Repair Service Answering Bureau (CRSAB), and various staff and data systems organizations. Typically, the terminals in the repair bureaus are connected directly to the FE, while those in the CRSAB, requiring access to multiple FEs are connected to a cross FE (XFE) which acts as a context switch to the FEs it serves.[6]

To achieve high availability, both the FE and the XFE systems are configured with backup systems. The FE systems are configured with a backup PDP 11/70 for every two FEs, and the Western Electric Company has developed a switch to allow the communications lines to be switched quickly from either FE system to the backup.

The FEs are also connected via data links to MLT. Up to 16 MLT controllers (DEC PDP 11/34s) can be handled by a single LMOS FE.

With LMOS-2 (the second generation of LMOS), a high-speed bus has been added to the system architecture. This 300-foot, 3.2-megabaud bus connects up to 12 FE systems (including backups) and provides the hardware base for the inter-FE communication described below.

III. THE FRONT-END TRANSACTIONS

Although the FE software includes some 50-odd transactions, the workhorses of the system are the four trouble processing transactions and the management report transactions. They comprise approximately 85 percent of the user transactions entered into the FE.

A typical trouble processing sequence is shown in Fig. 2, with the masks simplified somewhat for illustrative purposes. The full repair process is described in more detail in Ref. 7.

The Trouble Entry (TE) transaction is normally entered by a Repair Service Attendant (RSA) in the CRSAB when the customer calls to report a trouble. The attendant enters the customer's telephone number and the transaction returns a Trouble Report (TR) mask partially filled in with information from the customer's line record, information on any outstanding troubles associated with the telephone number, and the time when the repair bureau is able to have the trouble fixed. The TE transaction also initiates an MLT test on the customer's line.

When the partially filled-in TR mask is displayed, the attendant enters the trouble description provided by the customer. The attendant then negotiates a time with the customer (called the commitment time) when the trouble will be repaired and enters this information on

* Registered trademark of Teletype Corporation.

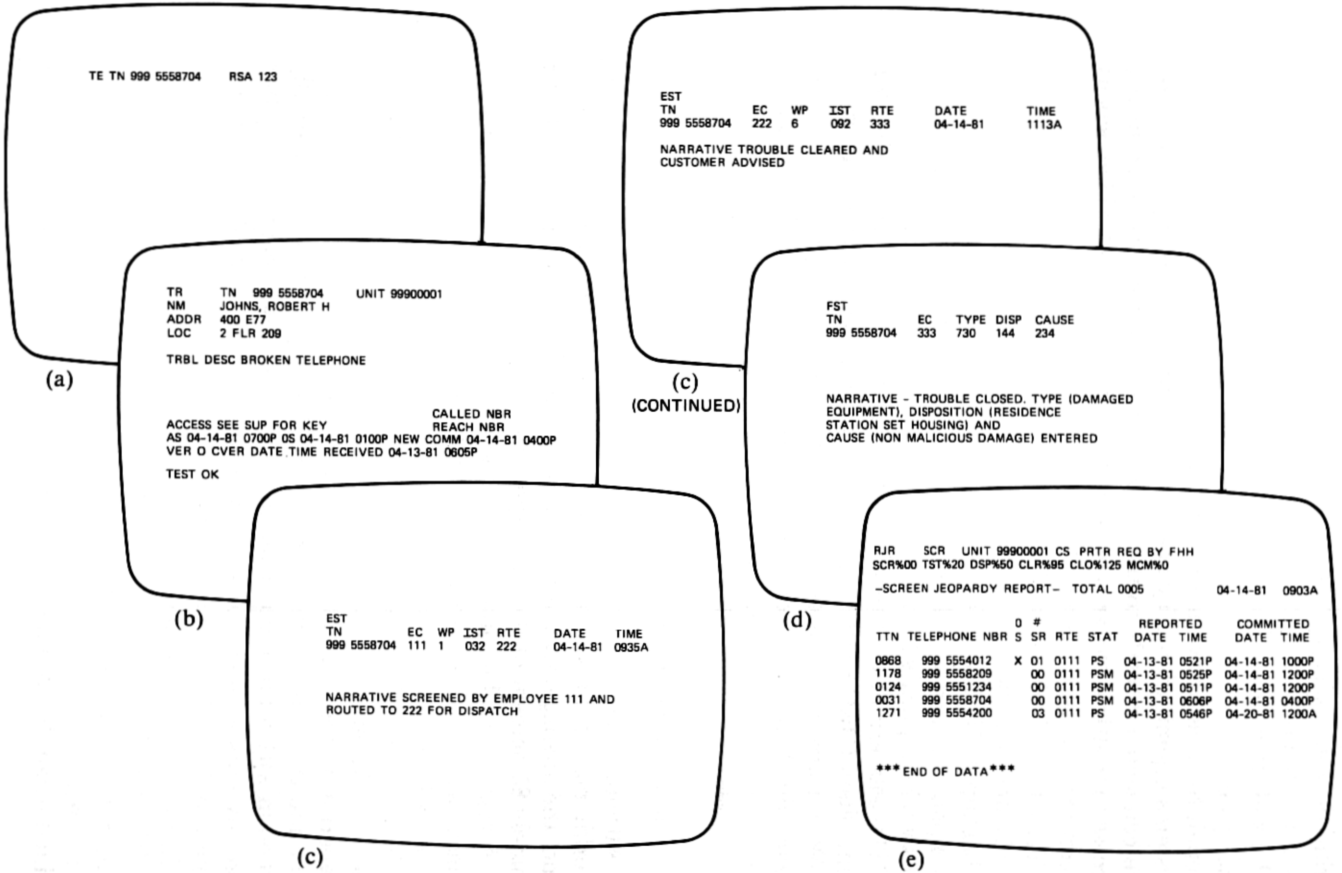

Fig. 2—Typical trouble processing sequence. (a) Completed trouble entry (TE) mask. (b) Completed trouble report (TR) mask. (c) Completed enter status (EST) masks. (d) Completed final status (FST) mask. (e) Completed jeopardy mask.

the mask. The TR transaction records the trouble in the FE trouble data base and generates a Basic Output Report (BOR). This output report contains the trouble description, MLT test results, and information from the customer's line record. The report is printed at the repair bureau assigned to fix the problem.

At the repair bureau, repair personnel screen the trouble, re-test it under special circumstances, dispatch someone to fix it, and once it has been cleared, notify the customer and close it. As each of these steps is taken, repair personnel enter current status information into the trouble data base using the Enter Status (EST) transaction. The status information includes the work performed on the trouble, who performed the work, and to whom the trouble is being routed next.

When the trouble is cleared and the customer advised, a final status is entered using the Final Status (FST) transaction. The FST records information on the cause and disposition of the trouble, and marks the closed trouble ready for transfer to the LMOS host (for later analysis[3]) and for deletion from the FE open trouble data base.

This status information allows the repair bureau management to track the trouble as it is being repaired. The Request Jeopardy Report (RJR) transaction prints out all troubles for which the bureau is in danger of missing the commitment time negotiated with the customer.

These five FE transactions—TE, TR, EST, FST, RJR—comprise the basic trouble processing sequence. Other FE transactions perform additional functions, allowing management and bureau personnel to get various reports on the number and status of outstanding troubles, to enter company and bureau-related information (e.g., the hours each bureau is open) and to administer the FE system.

IV. FRONT-END COMMUNICATIONS

In addition to its trouble tracking functions, the FE serves as a communications handler for the entire ARSB system (Fig. 1). Front-end communications software handles the interfaces to the synchronous terminals and printers on the FE and the links between the FE and the host, and the FE and the MLT.

From the users' standpoint, three types of access are provided:

(*i*) Terminals in the CRSAB and the RSB access the FE itself, either directly or through a XFE. These terminals are used to enter and track troubles, as described above.

(*ii*) Terminals connected to the FE also have access to the MLT test systems (PDP 11/34's) connected to the FE. Front-end applications and communications software enables repair bureau personnel to use FE transactions to initiate tests on a loop or a series of loops and to display or print the results.

The FE software also includes transactions to administer the MLT system and a download facility so that MLT controller software can be downloaded to the 11/34s from the FE.

(*iii*) Terminals attached to the FE have a switch-through interface to the LMOS host. Transactions not recognized by the communications software on the FE as FE transactions are automatically passed on to the LMOS host where they are treated as normal Information Management System (IMS) transactions. The terminal output from these transactions is routed back through the FE to the user's terminal. This interface allows a single LMOS terminal to access both the FE and host systems. Terminals connected directly to the host are normally provided for data base update groups, since these groups require access only to the host.

V. DESIGN CONSIDERATIONS

The principal design requirements for the FE system are high availability, data integrity, and performance.

5.1 Availability

The requirement for high availability stems from the critical role of the FE; it serves the trouble taking and tracking function central to repair bureau operations and is the gateway to the other ARSB components. This availability is provided by using backup hardware and by reducing the interdependency of the various system components. There is one spare FE for every two active FEs, plus the associated hardware necessary to manually switch from the active to the spare FE. This process normally takes 5 to 10 minutes during which time the FE is unavailable. The rapid recovery time was very important to our early success since it allowed us to crash relatively often without enraging our terminal users. We found several short outages to be much more acceptable to our users than one long outage. Measured availability in the field is normally over 99.5 percent.

The system is designed (see below) so that the repair bureaus and centralized answering bureaus can continue to operate efficiently even though the host is down. Thus, the host is not duplicated.

5.2 Data integrity

The requirement for data integrity is an operational requirement: customer troubles must not be lost, either while they are open or after they have been closed out but not yet moved to the host. This means that data base integrity must be provided when the system crashes. When the crash does not involve damage to the physical disk, this data integrity is provided by FE software that backs out any partially completed transaction. When a data base is damaged (e.g., from a head

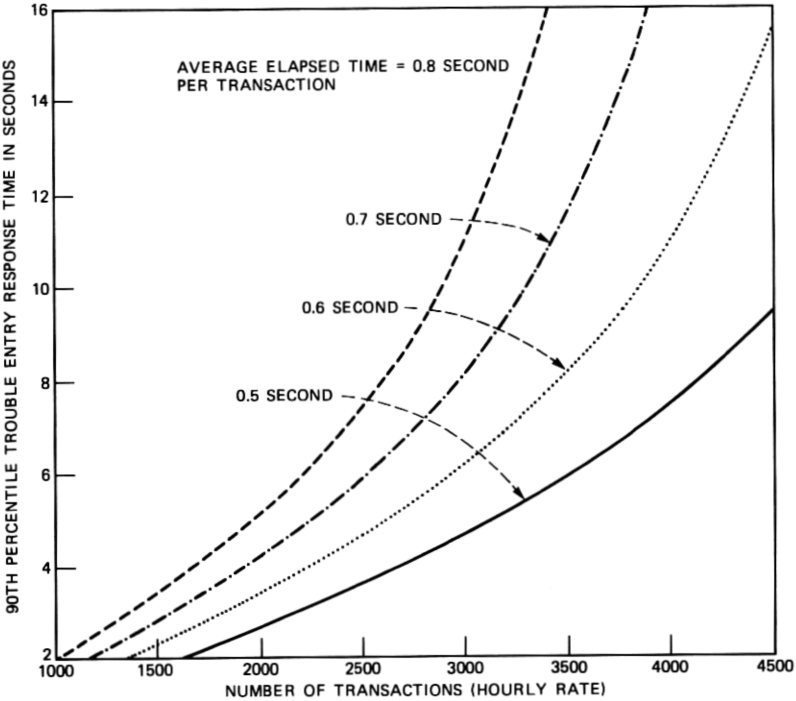

Fig. 3—Loop maintenance operations system FE average transaction response times.

crash), recovery is provided by using journal log tapes to update the latest (typically the previous night's) backup copy of the data base.

Data integrity must also be preserved when data are being transferred between the host and the FE. Data integrity is achieved here using standard acknowledgment techniques. The sending system looks for acknowledgment that its transmission has been received and correctly processed. If it does not receive that acknowledgment, the transmission is re-initiated. In addition, the receiving system recognizes duplicate transmissions and takes appropriate action to protect against system failures during final acknowledgment processing.

5.3 Performance

The curves in Fig. 3 depict FE performance as a function of the number of transactions handled per hour and the average elapsed time per transaction. The elapsed time for a transaction is the time the transaction spent in the FE from the time the transaction task is initiated until the time it is terminated. The "average elapsed time per transaction" is a measure of the average service time per transaction, and varies primarily with the mix of transactions running on the system (some transactions consume significantly more resources than others) and the amount of background work running.

The transaction mix varies from FE to FE because of local operational practices, the nature of the customer base (e.g., whether it is primarily business or residential) and the availability of other ARSB components (e.g., MLT). Transaction mixes will also vary on a single FE in the course of a day, normally with a higher percentage of TE and TR transactions appearing in the morning and a higher percentage of statusing transactions appearing later in the day as troubles are cleared and closed out. System capacity is determined on the basis of the transaction mix at the peak trouble reporting hour.

Front-end system performance in Fig. 3 is shown in terms of the 90th percentile response to the TE transaction since that is the most critical response time measurement. After initiating a TE transaction, the attendant in the answering bureau must wait for a TR transaction mask to be returned—with the customer on the line—before taking information on the reported trouble or negotiating a commitment time. The actual response time the attendant sees is the front end TE response time, plus approximately two seconds for XFE and transmission time.

The load on an FE can vary greatly as a function of the day of the week (Monday morning is traditionally the busiest trouble reporting period) and weather conditions. Most companies size their systems for a "normal busy hour," that is, the load they expect to see on a rainy Monday morning. As the system becomes more heavily loaded, the operating companies use system tuning parameters and administrative procedures to maintain response times to the answering bureau attendants at the cost of less critical functions. Catastrophic conditions (e.g., hurricanes) can, however, throw the system into an overload condition. In this case, a portion of the troubles will be taken manually (i.e., written down on paper) for later entry into the system.

VI. THE DISTRIBUTED ARCHITECTURE

The interface between the LMOS host and FE is a classic example of a technically conservative distributed system. The distribution of line record and trouble data on the FE and the host illustrate the design principles applied in the system:

(*i*) Where possible, data and functions are partitioned, so that they are required on only one system.

(*iv*) Where partitioning is not feasible, duplication of the data or function is provided to de-couple the system components and optimize performance and availability.

There are three major functional links between the host and the FE: processing closed customer troubles, updating customer line record, and generating the BOR for the repair bureau.

6.1 Processing closed troubles

The trouble data base on the FE is completely partitioned from the trouble information contained on the host. The FE knows only about open troubles; it has no knowledge of a trouble once it has been closed. The host has no knowledge of open troubles, but maintains a trouble history data base containing 40 days of closed trouble information.

The trouble processing functions are similarly partitioned. The FE transactions deal only with taking and tracking open troubles; the host transactions and the Trouble Report Evaluation Analysis Tool (TREAT) display and analyze historical trouble data.

Troubles marked for closed trouble processing by the Final Status (FST) transactions are batched for transmission to the host. The sending system (in this case the FE) waits for acknowledgment that the batch of troubles has been successfully received and processed by the host. If such acknowledgment is not received, the FE will reinitiate the transmission. This acknowledgment process is required because it is critical that closed troubles are transmitted to the host for measurement and reporting purposes. The transmission process is expensive in terms of system resources, and system administration facilities are provided so that these functions can be turned off during prime shift busy hours.

6.2 Customer line record updates

In contrast, the customer line record information on the FE is a duplicated subset of the line record information on the host. The data is duplicated to minimize the interdependence of the FE and the host for both performance and availability reasons. The data on the FE "miniline record" data base is approximately 10 percent of the data stored in the host's line record. It is the subset that is critical to the repair process. (One of the more predictable evolutionary phenomena of the system has been an increase in the data which is seen as "critical.")

Updating the line record data bases is done using a strict master-slave relationship, with the LMOS host being the master. The process begins when the host data base is updated by the Automatic Line Record Update[2] programs processing service orders, or by data base personnel issuing host transactions to change the line record.

Nightly, host line records that have changed are batched for transmission to the FE. The FE initiates the "change miniline record" process in which the host transmits a group of new line records, awaits acknowledgment from the FE, then sends the next group, until all the updates are complete. The FE line records can only be updated as a result of the host's updates or from off-line bulk load programs. Theoretically, this should keep the data bases synchronized once all the host change requests have been processed, although the two data bases may be out of step for a considerable period of time until the FE "catches up." However, the data bases do get out of synchronization and cross-audit programs are provided to detect discrepancies and reload the FE "miniline record." More importantly, none of the software of the host or the FE depends on the two data bases being synchronized.

6.3 *The output report*

Output report processing is an example of an area where a *function* has been duplicated on the host and the FE to avoid system interdependence. Normally, when a trouble is entered into the system, the FE sends the trouble description and the test results to the host. The host takes this information, adds the full line record information and an abbreviated history of the troubles taken against this telephone number over the last 40 days, and formats the BOR, which it sends back to the FE to be printed at a repair bureau.

If the host is unavailable, an output report must still be sent to the bureau, since this piece of paper informs the bureau that they have a trouble to be worked. This output report is a "Mini Output Report" (MOR), generated by the LMOS FE. The MOR has only a subset of the line record information and it does not have any trouble history. It does, however, have most of the information critical to repairing the trouble.

VII. SOFTWARE EVOLUTION

The original LMOS FE software used an operating system called Bell Operating System (BOS) which was developed for, and tailored to, the LMOS application. The system was written in assembler code and Digital Equipment Corporation's MACRO-11* language and included a file system with logging and recovery procedures, plus a sophisticated communications handler which allowed us to support synchronous terminals and interfaces to the host, the XFE, and the MLT.

In 1978, a decision was made to redesign and reimplement the FE software. This new system (LMOS-2), which is currently being tested at Michigan Bell Telephone Company, uses the *UNIX*† program operating system.

The applications software is written in C (a high-level language) and a superset of C called Transaction Specification Language (TSL).[8] The motivation for redesigning the software was threefold:

(*i*) The success of the LMOS system generated many requests for additional functions. Change requests started coming at the rate of one

* Registered trademark of Digital Equipment Corporation.
† Trademark of Bell Laboratories.

a week. Being responsive to such requests with minimal ripple effects was clearly a growing requirement, and "change tolerant" software became a new objective. Most of the requests required changing the FE data bases or the FE transaction masks. This motivated the development of a data base management system, a generalized mask handler, and an internal data structure designed to protect the transactions themselves from knowledge of the physical layout of data on the screen or in the data base.[8]

(*ii*) The original LMOS system was designed assuming that repair bureaus were geographically based. Thus, an FE could be expected to service a number of repair bureaus within a geographical area, and, conversely, a given repair bureau was expected to need access to only one FE. In most cases, when a trouble had to be referred from one repair bureau to another, the two repair bureaus were both based on the same FE, and software was provided so that both could obtain up-to-date information on the status of the trouble. In those rare cases where a trouble needed to be handled by a repair bureau not based on the same FE, trouble referral and any status information had to be handled manually.

The addition of a high-speed bus (Fig. 1) to the ARSB architecture allowed us to transcend some of the geographical constraints in the original LMOS system by providing multi-FE transactions. Thus, in LMOS-2, a trouble can be referred to a repair service on any FE on the bus, and its status will be known to both the original and the currently responsible bureau.

The multi-FE features are still evolving, but have been useful in meeting the requirements of a changing Bell System organization. Shortly after LMOS-2 development started, the Bell System reorganized along market segments. Instead of having a repair bureau serving a geographical area, new business, residence, and network repair bureaus were planned. The more recent reorganization of the Bell System into regulated and unregulated companies has imposed yet another set of requirements on LMOS. These changes will also make use of the new ARSB architecture.

The evolving requirements resulting from the changing organizational environment have reinforced our commitment to producing "change tolerant" software.

(*iii*) Software technology had developed to the point where the idea of building a transaction system from a number of re-usable software tools or components appeared feasible. This tool-oriented approach is described in Ref. 8.

VIII. SUMMARY AND CONCLUSION

The LMOS FE computer is the "trouble tracking" component of the

ARSB. Customer reported troubles are entered into the system by attendants in a CRSAB, automatically tested using MLT, and routed to the appropriate RSB. At the repair bureau, status information is updated at each step toward the completion of the repair, and management reports are generated to warn of overload conditions and potential missed commitments.

The first LMOS FE was installed at Southwestern Bell Telephone Company in June, 1975. As of year-end 1981, approximately 230 FEs were deployed in 18 operating telephone companies covering approximately 65 million customer lines.

REFERENCES

1. R. L. Martin, "Automated Repair Service Bureau: The System Architecture," B.S.T.J., this issue.
2. C. M. Franklin and J. F. Vogler, "Automated Repair Service Bureau: Data Base System," B.S.T.J., this issue.
3. S. P. Rhodes and L. S. Dickert, "Automated Repair Service Bureau: The Trouble Report Evaluation and Analysis Tool," B.S.T.J., this issue.
4. O. B. Dale, T. W. Robinson, and E. J. Theriot, "Automated Repair Service Bureau: Mechanized Loop Testing Design," B.S.T.J., this issue.
5. P. S. Boggs and J. R. Mashey, "Automated Repair Service Bureau: Cable Repair Administrative System," B.S.T.J., this issue.
6. J. P. Holtman, "Automated Repair Service Bureau: The Context-Sensitive Switch," B.S.T.J., this issue.
7. M. W. Bowker et al., "Automated Repair Service Bureau: Evolution," B.S.T.J., this issue.
8. R. F. Bergeron and M. J. Rochkind , "Automated Repair Service Bureau: Software Tools and Components," B.S.T.J., this issue.