EDPACS
THE EDP AUDIT, CONTROL, AND SECURITY NEWSLETTER

OCTOBER 1992 VOL. XX, NO. 4

THE CONTINUOUS PROCESS AUDIT SYSTEM: KNOWLEDGE ACQUISITION AND REPRESENTATION

FERN HALPER
JOHN SNIVELY
MIKLOS VASARHELYI

The Continuous Process Audit System (CPAS) was developed by AT&T Bell Laboratories for the AT&T internal audit organization. The purpose of CPAS is to provide its users with an integrated diagnostic view of large financial systems. This monitoring methodology can be applied to a larger family of applications in the general area of system monitoring and management.[1]

CPAS works by gathering and collating diagnostic data produced from different parts of a system and presenting key checkpoints and analytics in a workstation environment. The purpose of the analytics is to ensure the financial integrity of the system and call the information systems auditor's attention to any anomalies within it. The same data can be used to answer other business questions, and the methodology can be extended to integrate other systems and be used by managers as well as information systems auditors.

This article describes the issues involved in acquiring and implementing (i.e., in representing) the knowledge needed to monitor large complex systems. The types of knowledge that must be acquired for a monitoring system are discussed as well as how to gather it.

THE MOTIVATION FOR CONTINUOUS MONITORING

Traditional methodologies offer very limited assurance to the auditor and cannot recognize problems in large integrated computer systems in a timely manner. Consequently, there is an exponential increase in the risk associated with auditing these systems. At the same time, conventional risk management tools have proven to be highly inadequate. These conditions have demonstrated the need for continuous process monitoring to reduce the audit risk associated with examinations of large complex systems. The nature of the methodology used to accomplish this type of monitoring is important.

IN THIS ISSUE

Executive Editor
THE EDP AUDITORS ASSOCIATION

Editor
BELDEN MENKUS, CISA

Associate Editor
MICHAEL P. CANGEMI, CPA, CISA

EDP AUDITORS ASSOCIATION, INC. FOUNDED 1969

AUERBACH PUBLICATIONS
Warren Gorham Lamont
Division of Research
Institute of America Inc.

Typically, the audit and management of large computer systems is fragmented among people. A number of individuals may possess detailed knowledge about how the various modules of the system work, but most likely there is no centralized understanding of how the system as a whole is functioning. Effective audit and management of a system should integrate information from the different system modules to help the auditor gain that understanding. Manual monitoring and auditing around the computer are no longer sufficient approaches as systems become larger and more complex and the potential for revenue loss increases.

AN OVERVIEW OF THE CPAS METHODOLOGY

THE PURPOSE OF CPAS IS TO PROVIDE ITS USERS WITH AN INTEGRATED DIAGNOSTIC VIEW OF LARGE FINANCIAL SYSTEMS.

Conceptually, as depicted in Exhibit 1, the monitoring system consists of three levels: data provisioning, knowledge, and presentation. The data provisioning level provides the raw data for the analysis. Data can be obtained in three different ways:

- ☐ The auditor or manager can extract data directly from application data bases through the use of a fourth-generation language, such as FOCUS.
- ☐ Reports can be prepared and delivered specifically for the monitoring application.
- ☐ Data can be extracted from operational reports that are produced by the system. This approach essentially is a secondary means for obtaining data.

The data that has been obtained is stored in a data repository, such as a data base, or stored in raw form. This data is of a dynamic time-specific nature because it contains information on the actual status of the application being monitored.

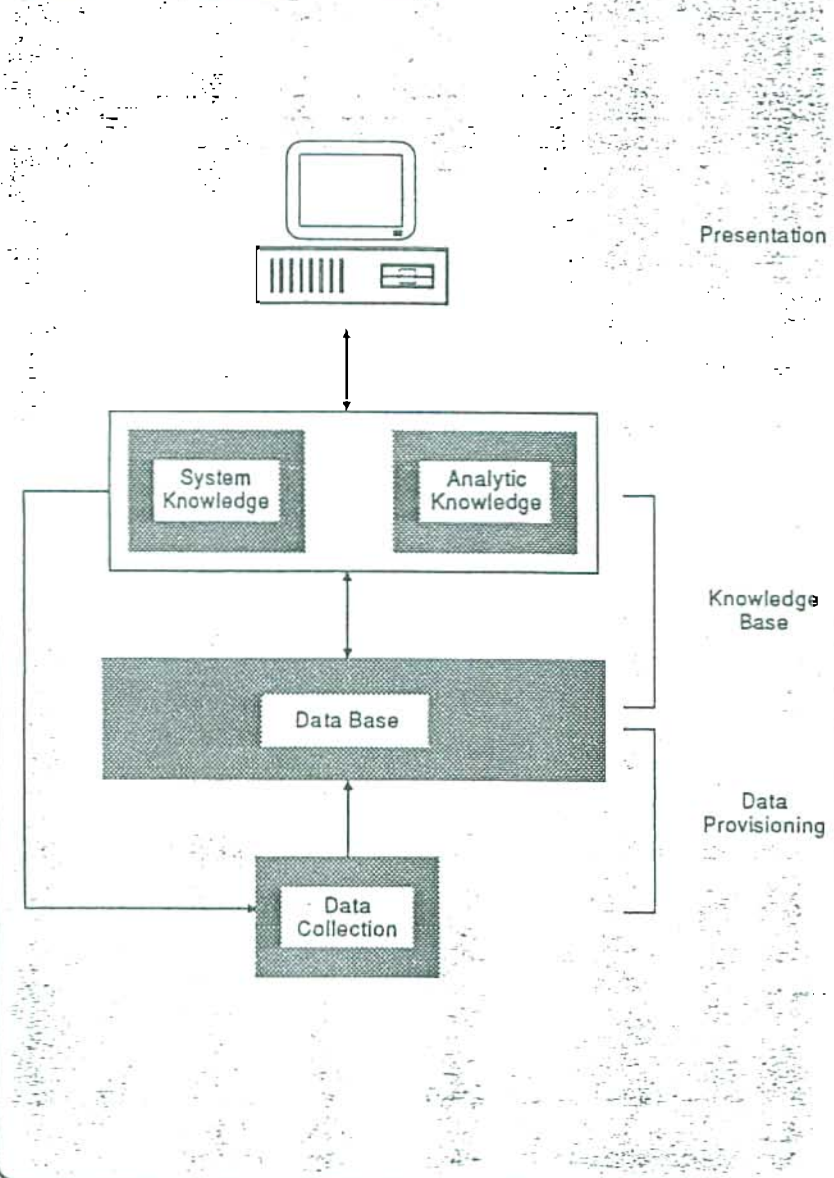
Another form of information of a more permanent character—but not of a static nature—is generated and stored in the knowledge base. This base includes information about the structure of the system being monitored and the analytic definitions. Data is analyzed using various software, and the output is sent to the presentation level of the system.

Currently in CPAS, operational reports are used as the data source. Data from these reports is extracted using pattern-matching programs placed in a data base, and the system is analyzed using the data from this data base.

The analysis is structured by flowcharts representing the different modules of the system. The flowcharts are designed to represent a functional view of the system. They convey knowl-

If you have information of interest to EDPACS, contact Eric Birenbaum, Senior Editor, Auerbach Publications, Warren Gorham Lamont, a division of Research Institute of America Inc., One Penn Plaza, New York NY 10119. EDPACS (ISSN 0736-6981) is published monthly by Auerbach Publications, Warren Gorham Lamont, a division of Research Institute of America Inc., 210 South St., Boston MA 02111-2797, (617) 423-2020. The subscription rate is $135/year in the US. Prices elsewhere vary. Printed in USA. Second-class postage is paid at Boston MA. Postmaster: Send address change to EDPACS, Auerbach Publications, 210 South St., Boston MA 02111-2797.

Exhibit 1. *Conceptual View of Methodology*



edge, provide a conceptual structure, and serve as a display vehicle for some of the analysis of the system being monitored. A high-level view of the system is available at level 0 of CPAS. (So-called system health indicators are an example of what might be presented here.) Detailed information about each system module is available at lower levels.

Information is presented by CPAS primarily as metrics and analytics. Metrics are defined as direct measurements of a system. They can represent simple input or output from the system. These metrics can be compared with the applicable system standards, and if a standard is exceeded, an alarm appears on the screen.

Analytics are defined as a depiction of some relationship among the metrics. The analysis can be a snapshot view of the

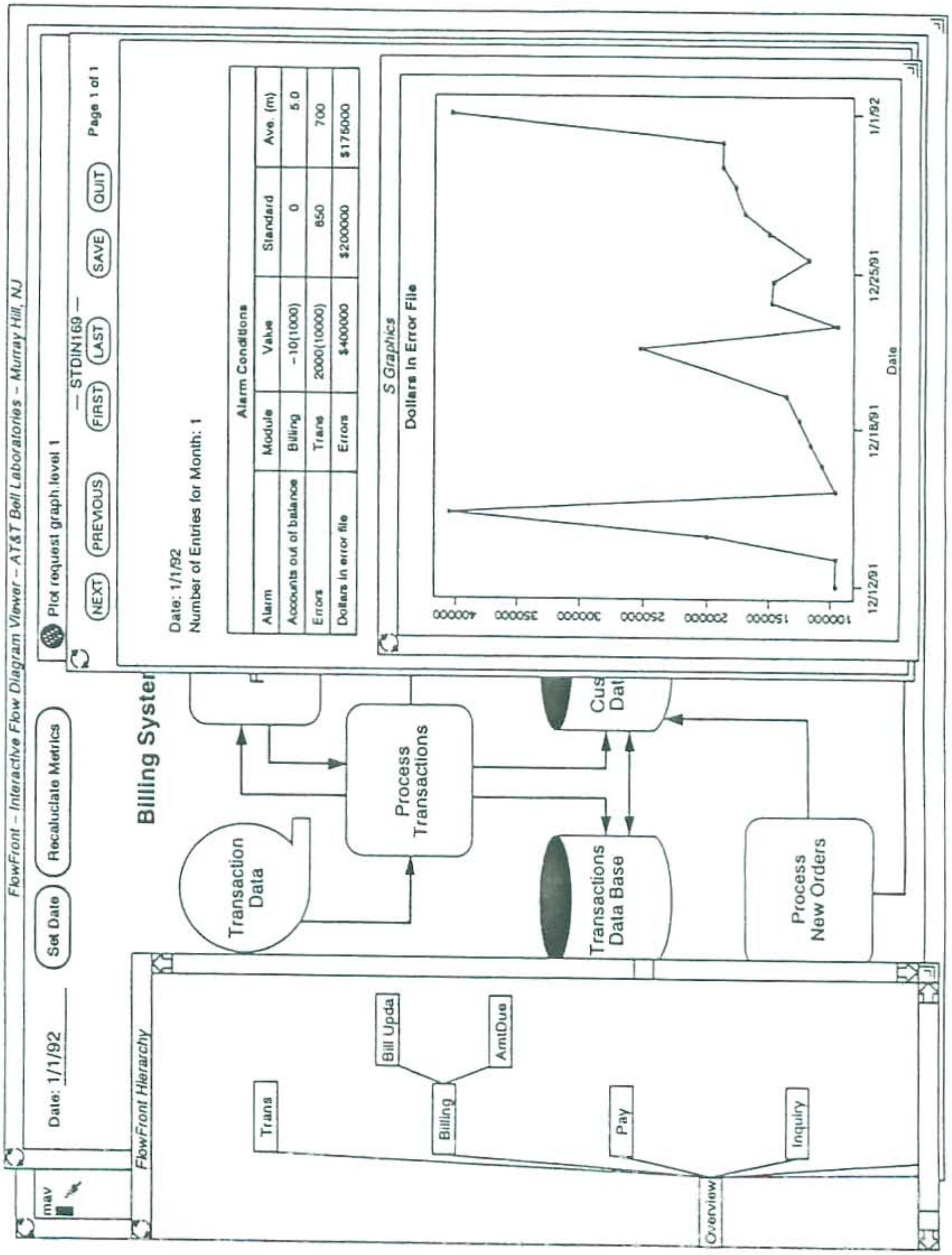


Exhibit 2. *View of the Billing System*

system for the day, or the analysis can be performed using time series data. Exhibit 2 illustrates how such an analysis might be presented. This screen display was prepared using the CPAS toolkit and has the look and feel of any CPAS application. This depiction represents a level 0 view of a billing system. This system might bill electrical use, telephone calls, or equipment sold. In this system, transaction data might relate to such things as electricity use recorded in kilowatts or records. This data is received by the organization's information processing center. The data is passed through to different modules of the system.

The hierarchy window on the left in Exhibit 2 indicates what part of the billing system is represented by the flowchart. The overview node in the hierarchy is being depicted. The date bar indicates the date that the analysis uses as the base date. (In this illustration, it is 1/1/92.) Other bars can be added to this display to break down the analysis further. For example, a bar labeled *location* might represent different billing locations, such as the New York City boroughs of Manhattan or Queens. A high-level report, accessed through a menu for this module, shows any alarm conditions that exist for this base date. These alarm conditions should be user defined.

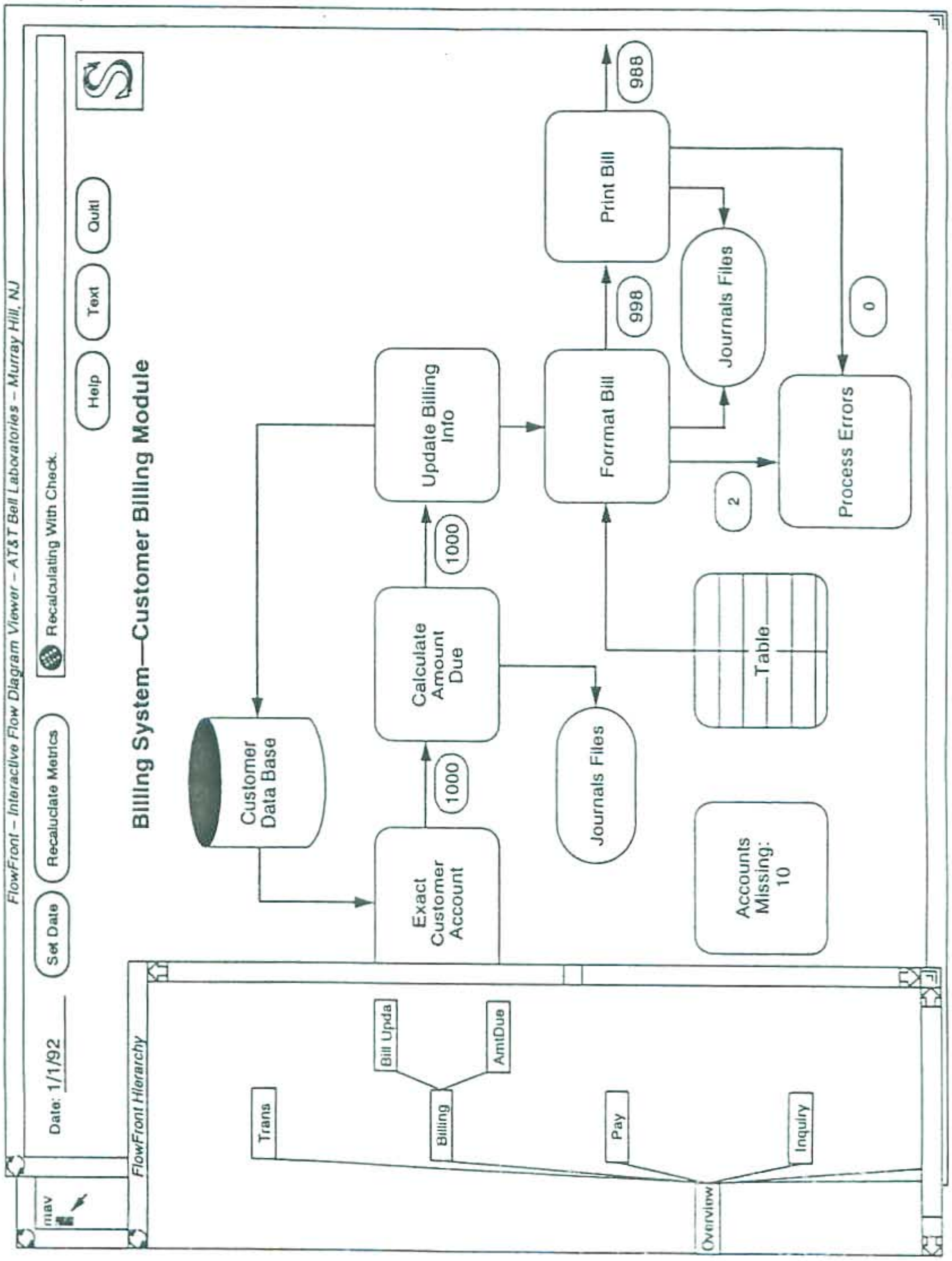
CPAS WORKS BY GATHERING AND COLLATING DIAGNOSTIC DATA PRODUCED FROM DIFFERENT PARTS OF A SYSTEM AND PRESENTING KEY CHECKPOINTS AND ANALYTICS IN A WORKSTATION ENVIRONMENT.

On this date, the information systems auditor's test for completeness of input indicates that there is an out-of-balance condition in the billing module. In addition, an error threshold has been exceeded in the transaction processing module, and this may have caused the error file to grow much too large—in this instance, it amounts to $400,000. A time series plot shows that the error threshold has been exceeded on several occasions during the past month and that generally the errors seem to be removed from the file. The information systems auditor, however, probably would want to understand why this has occurred.

An auditor or manager may want to look at the customer billing module in more detail to learn more about the out-of-balance condition. A CPAS user who logs on only intermittently probably would select the billing node in the hierarchy window to check how many times this condition has occurred since the last log-on. Doing this would cause the branch to the selected node to be highlighted, and a new flowchart representing the customer billing module would appear on the workstation monitor. This condition is illustrated in Exhibit 3.

In this depiction, the metrics, indicated as boxes next to the flowchart, show the flow of accounts through the customer billing module. The alarm, which is found on the lower left of Exhibit 3, indicates that there were 10 accounts lost between the format bill module and the print bill module on this date. The user could use the information available in the system to find out more about such an out-of-balance condition. In addition, the CPAS user would want to know how many times this has happened during the past month or longer period. The user also could enter other modules to find out more about the other alarm conditions. The base date can also be changed to study what has happened on previous days.

Exhibit 3. Customer Billing Module



KNOWLEDGE ACQUISITION

CPAS has been developed to provide those who use it with an integrated diagnostic view of a system in order to determine the financial integrity of that system. This requires that the user have an understanding of how to analyze the different system modules and how these modules are integrated. Often, as already has been suggested, knowledge about very large systems is fragmented among different sources. For example, a person may have detailed knowledge about one module of a system but not how that module links to other modules. Or a person may have a great deal of experience managing certain types of billing systems but not necessarily the billing system under investigation.

The goal of the knowledge acquisition process is to accumulate and integrate this diffuse knowledge. To do this, two kinds of knowledge are needed—knowledge about the system itself and knowledge about how to analyze the information coming out of the system.

Knowledge about the system includes a functional view of the system as well as operational information about the system. This operational information is needed so that the monitoring system can integrate information from different parts of the application system for the same period (i.e., the same run).

Knowledge about how to analyze the system includes how to interpret the data day by day, which is viewed in the form of a snapshot of the system's activities at a particular point in time. This knowledge also encompasses knowing how to determine if there are any long-term problems with the system.[2]

The concept of continuous auditing was relatively new at the time that work on what became CPAS began. This concept also was not understood totally by all of those who might be expected to use such a system. Thus, it was decided to prototype CPAS as an enhanced decision support system. The plan was to add expertise into the system as an iterative process. This knowledge acquisition process is summarized in Exhibit 4. The methodology evolved as work progressed on developing the monitoring system.

The four steps of the methodology are described in more detail in the next sections with illustrations of the operational procedures used. These may vary substantially depending on either the context or the scope of the system being monitored.

THE MONITORING SYSTEM CONSISTS OF THREE LEVELS—A DATA PROVISIONING LEVEL, A KNOWLEDGE LEVEL, AND A PRESENTATION LEVEL.

Exhibit 4. *Knowledge Acquisition Process*

Step	Purpose
Unstructured Interview	Understand Problem Domain
Structured Open Ended Interview	Understand Problem Structure Determine Expert Versus Novice
Task Analysis	Use Reports to Map Basic Snapshot Analysis
CPAS Used	Build Expertise into System by Using CPAS to Map More Sophisticated Analysis

THE USE OF UNSTRUCTURED INTERVIEWS

As a first step, the conventional unstructured interview was used to begin to understand the subject domain. Informal meetings were conducted with various information systems auditors to specify the target objectives of the monitoring system and to determine who would provide the necessary expertise. It was necessary to understand what kind of knowledge was needed and where within the organization that knowledge resided.

STRUCTURED OPEN-ENDED INTERVIEW AND ARCHIVAL ANALYSIS

Once this foundation was in place, structured open-ended interviewing techniques were used to determine how the subject was organized. The auditors that were interviewed provided a high-level view of the system, which included some overall analytics as well as a breakdown of the system modules. Documentation was located and obtained for each module of the system, and contacts with the experts were developed. Meetings with each of the system experts and reading of the available documentation followed. Once knowledge about the structure of the system had been extracted, meetings were held with the individuals responsible for auditing these different modules.

Each session started with a flowchart of the system that the auditor was responsible for. The auditor was asked to describe this system and how he or she would undertake an audit of it.

Typical questions asked of the systems experts, who were either information systems auditors or information systems developers, included:

- ☐ What are the main system modules?
- ☐ What is the purpose of each module?
- ☐ How do transactions flow through each module?
- ☐ How can the transactions be altered?
- ☐ Where can transactions be lost?
- ☐ Where are the suspense files?
- ☐ How long can transactions stay in suspense?
- ☐ How are transactions passed from one module to another?
- ☐ What is the relationship between the modules?
- ☐ What does the interface between the modules look like?
- ☐ What are the controls over each module and between the modules?

Typical questions asked about system operation included:

- ☐ How is a run day defined for batch systems?
- ☐ How are files managed?
- ☐ How are reports managed?
- ☐ How often are they produced?
- ☐ How are reruns treated?
- ☐ How long are reports stored on the system?
- ☐ How can information uniquely be identified?
- ☐ Is there a common denominator in the system?

Typical questions asked about audit and management analysis included:

- ☐ What are the goals of the analysis?

- ☐ What controls are being tested?
- ☐ Which pieces of data are used from the reports, and how is that data used?
- ☐ How is the data interpreted daily and over time?
- ☐ What decisions are made from the data, and how are they made?
- ☐ What are the standards against which data is measured?

Some of these sessions were tape recorded, and the information given was categorized as an audit objective, an audit focus, an audit rule, information about the system, or a potential system problem. Concurrently with these preliminary meetings, the relevant audit work papers and reports were examined to identify:

- ☐ The current audit steps.
- ☐ The data items being examined.
- ☐ The specific rules concerning the required audit evidence.
- ☐ Any actual procedures for data gathering, search, and analysis.

Because the CPAS monitoring system was by definition data driven, it was necessary to devise a method to identify which data was used and how the data was used. Operational reports were used by the information systems auditor as the primary data source. This was done because direct data base access was not possible, and the reports provided both detail and summary information. After a careful study of the system and the goals of the analysis, it was determined that an algebraic representation could be used to model the system and specify the analysis.

TASK ANALYSIS

Next, the auditors and managers were helped to specify the analysis using the operational reports. First, the goals of this analysis were identified again. Then a walk through the analysis was performed with the user. Each piece of information that the user thought was important was given a variable name. These variables were used to specify the analysis. The variable representation was useful because it could tie directly into a data base representation. (These analysis specification sessions could vary in length depending on the size of the module and the expertise of the auditor or manager interviewed.) At the end of a session, a preliminary specification on how to analyze a module was produced.

THE USE OF CPAS

This preliminary specification was then handed to a programmer to be implemented. Once the analysis was realized in CPAS, the user provided feedback, and the analysis was changed accordingly. The job of integrating the analysis for each module ultimately fell to a knowledge engineer. This knowledge engineer also needed to know information about the system that the experts did not need in conducting their manual analysis of it. For example, managers or auditors who use data from different reports to assess the completeness of input can deal with such operations as system reruns visually. The knowledge engineer needed to know certain operational aspects of the system in order to try to deal with this in the system. This implies, incidentally, that the task of analyzing the whole system is greater than simply analyzing the parts individually.

KNOWLEDGE REPRESENTATION

In the initial implementation of CPAS, SQL was used to extract the needed data from the data base. For example, individual metrics were calculated in separate SQL queries. More complex analytics were calculated using SQL embedded in C. Because the SQL queries did not resemble the algebraic specifications, it made the analysis more difficult to validate and maintain. It was concluded that the process could be improved if the implementation language more closely resembled the algebraic language used by the knowledge engineers. Ideally, this language could be used directly by the knowledge engineers in place of the specification language as a knowledge acquisition tool. This would cut down on implementation time and make the analysis easier to maintain and validate.

For example, the individual who would use CPAS might be interested in electrical use in kilowatts and records that is accepted to be billed from three different places in a Northeast region (e.g., Manhattan, Queens, and Brooklyn). Data related to this analysis might be found on three different daily reports, one each for Manhattan, Queens, and Brooklyn. Included in the report is information about the date, region, location, the use errored out, and the use accepted to be billed. The data can be extracted from the reports and put in a data base table. The data base table adds structure to the analysis that the individual reports did not have.

Exhibit 5 represents an output table from a relational data base that shows a schematic view of use in both records and kilowatts from three different sources—Manhattan, Queens, and Brooklyn—for a hypothetical utility company. Certain parameters, such as date and region, which were supplied to the query, do not appear in the table. The data base contains information about errors (e.g., err1, err2) and the number of records and kilowatts finally accepted to be billed. This data may be used to determine the error rates in the Northeast region for a particular day.

Exhibit 5. *Output of Relational Data Base*

units	src	err1	err2	acpt
rec	NYC	0	2	300
rec	QUEENS	2	3	350
rec	BKLYN	3	0	350
kwats	NYC	0	50	1000
kwats	QUEENS	8	88	732
kwats	BKLYN	25	0	1500

date=4/1/92 region=northeast

In this table, units and src are indexing information. The other three column labels—err1, err2, and acpt—are values associated with these names. The software uses the label information in output tables to produce lists of name-value pairs:

```
rec[NYC,err1] = 0
rec[NYC,err2] = 2
rec[NYC,acpt] = 298
.
.
.
kwats[BKLYN,acpt] = 1500
```

These are name-value pairs that represent the exact information found in Exhibit 5. To use the representation, the user needs to know where the key columns stop and the data begins (err1, err2, acpt) and how the brackets are used. These name-value associations are produced automatically by the software, for later use in algebraic equations. Once the name-value pairs are generated, the array language will operate on them. In the array language, index variables, such as NYsrc and Errors, are defined. Their values implicitly are iterated when they are used in expressions. In addition, the language uses a summation convention, whereby index variables appearing on the right side of an equation and not on the left are summed over before assignment takes place. Other features of the language include the ability to deal with missing data.

To illustrate how this works, assume that it is necessary to calculate the error rate for units = rec for error types 1 and 2 for the entire Northeast region. It is possible to sum all of the accepted records and calculate all errors and error rates using the data shown in Exhibit 6. In addition, percent.error1 and percent.error2 can be compared with a standard. The output from this model also can be used as input to more expert rules.

THE EXPERIENCE OF USING CPAS

The knowledge base for a CPAS-like application needs to contain functional and operational information about the system itself as well as how to analyze the system. It is convenient to think of the financial system in terms of an algebraic model and to

Exhibit 6. *Input and Error Calculation Modules*

```
[NYsrc] = {NYC, QUEENS, BKLYN}
tot.acpt[NY] = rec[NYscr, acpt] #implicit summation
tot.input[NY] = rec[NYsrc,err1] + rec[NYsrc,err2] + rec[NYsrc,acpt]
```
Input Calculation Module

```
[Errors] = {err1,err2}
tot.rec[Errors] = rec[NYsrc,Errors]
percent.error1 = (tot.rec[err1]/tot.input[NY])*100
percent.error2 = (tot.rec[err2]/tot.input[NY])*100
```

The line *tot.acpt[NY] = rec[NYsrc,acpt]* is equivalent to

```
tot.acpt[NY] = rec[NYC,acpt] + rec[Queens,acpt] + rec[BKLYN,acpt]
```

and the line *tot.rec[Errors] = rec[NYsrc,Errors]* is equivalent to these two assignments:

```
tot.rec[err1] = rec[NYC,err1] + rec[QUEENS,err1] + rec[BKLYN,err1]
tot.rec[err2] = rec[NYC,err2] + rec[QUEENS,err2] + rec[BKLYN,err2]
```
Error Calculation Module

define parts of the analysis in terms of algebraic variables. Although the knowledge engineers used algebraic equations in writing the CPAS specifications, there was definite reluctance on the part of some people to think about a system in these terms.

People with greater competence and skill were more comfortable, as CPAS users, with the concept. In fact, these people actually did have a model of the system in their heads and found it natural to use the variable notation. Novice information systems auditors seemed to focus only on system flowcharts and metrics when asked how they analyzed the system. This may be because they really did not understand the concept of a continuous audit.

Typically, auditors will work with only a few days of systems operation data when they are performing an audit. The continuous process audit concept was new and not appealing to some. Consequently, the initial analysis focused on a snapshot analysis of the system. This involved looking for completeness of input.

An auditor or manager who used CPAS and saw the trending information that it made available would start to ask for additional analytics. Therefore, the process can be seen as iterative, in which lower levels of knowledge and use lead to progressively sophisticated and demanding use. Consequently, it can be said that the goal was to use CPAS itself as well as to help build audit and management expert knowledge. This can lead to a higher hierarchy of control and introspection in the actions of management and audit.

METHODOLOGICAL ISSUES

Several issues related to the development and application of the CPAS methodology should be noted. Unlike much of the artificial intelligence and expert systems work that has been done in this area, this effort did not resort to formalized knowledge representation languages and classification approaches. Existing computer tools, such as PostScript, were used to achieve similar results.

The nature of the available human knowledge in this case substantially affected the approach used in developing CPAS. This knowledge, for both organizational and historical reasons, was extremely dispersed. This situation forced the knowledge engineer to be a sort of discoverer and integrator.

The objectives established from the start of this effort and the reality of the eventual CPAS implementation were substantially different. Behavioral obstacles to a major change in auditor behavior led to the creation of a lower-level, less rule-based implementation.

It is natural to expect that the development of knowledge-based tools like CPAS will force the development of metamanagement knowledge that encompasses skill in the manipulation of these tools. ■

Fern Halper is a member of the AT&T Bell Laboratories, Murray Hill NJ, technical staff. Before joining AT&T in 1986, she was a member of the

Colgate University geology faculty. Halper received a PhD in oceanography from Texas A&M University.

John Snively is a member of the AT&T Bell Laboratories, Murray Hill NJ, technical staff. Before joining AT&T in 1986, he was a member of the University of Utah mathematics faculty and the Brigham Young University civil engineering faculty. Snively received a PhD in mathematics from Texas A&M University.

Miklos Vasarhelyi is the KPMG Peat Marwick Professor of Accounting at the Rutgers University Graduate School of Management and a consultant supervisor at Bell Laboratories, Murray Hill NJ. Before assuming his current positions, Vasarhelyi was associate professor and director of Columbia University's Accounting Research Center. He received a PhD in management from the University of California at Los Angeles.

Notes

1 The PROMETHIUS effort at AT&T is the management counterpart to the audit application. It uses the same technology and platform as CPAS. It is envisaged that when it is developed fully, PROMETHIUS will support the management and monitoring of large financial applications at the operational, managerial, and executive levels.

2 To date, snapshot analytics have been incorporated only into the knowledge base. The knowledge needed to analyze the trending data can be acquired by using the trending information provided by CPAS.

SUPPORT TOOL REVIEWS

End-User and Departmental Computing, by William Sampias, James Kincaid, and Albert Marcella, Jr. Institute of Internal Auditors (Customer Service Center, Post Office Box 14099, Orlando FL 32889-0003), 1992; videotape, discussion leader's guide, and participant's workbook; various paginations. Order Item 9000–7. Price: $350, including shipping; send payment with order in US funds. Discount for IIA members. Add $40 for orders from outside of North America. For checks drawn on non-US banks, add $30.

This product is described by the Institute of Internal Auditors as a video-assisted seminar. It is the first in a series of educational materials to be developed from the institute's *Systems Auditability and Control Report.* Its content is drawn from the report's Module 7 on end-user and departmental computing. Four hours of CPD credit are available to those who complete this course. It begins with an overview of this type of computing. The remainder of the course deals with relevant risks, controls, and audit issues. Both of the documents that accompany the videotape include a glossary of relevant terms, a bibliography, and a reprint of the applicable institute standards. The discussion leader's guide also includes two analytical exercises to be worked by the seminar students.

The videotape's content is closely related to that of the printed course materials. The presentation in the videotape, however, is sufficiently self-contained to be suitable for use in introducing senior executives and managers to the risks and control concerns related to this type of computing. ■

Confidential Information Sources: Public And Private, Second Edition, by John Carroll. Butterworth Heinemann (80 Montvale Avenue, Stoneham MA 02180), 1991, 386 pp.;