THE UNIVERSITY OF CALGARY

COMPUTER ASSISTED OPERATIONS,
A PRACTICAL APPROACH TO NETWORKING FIELD
COMPUTERS IN A LIMITED COMMUNICATIONS ENVIRONMENT

by

Geoffrey P. Hiltz

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF

Master of Engineering

DEPARTMENT OF ELECTRICAL ENGINEERING

CALGARY, ALBERTA

August, 1990

ISBN 0-315-61960-0

THE UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled, "Computer Assisted Operations, A Practical Approach to Networking Field Computers in a Limited Communications Environment", submitted by Geoffrey P. Hiltz in partial fulfillment of the requirements for the degree of Master of Engineering.

Supervisor, Dr. S.T. Nichols
Department of Electrical Engineering

Dr. John Kendall
Faculty of Science

Dr. W.C. Chan
Department of Electrical Engineering

Dr. G.S. Hope
Department of Electrical Engineering

Date August 28, 1990

ABSTRACT

This thesis describes Computer Assisted Operations (CAO), a practical method for sharing computer information in oil field operations. CAO is a framework where computer applications, running on the PC microcomputer platform, can communicate information for on site field computer processing. CAO provides transaction processing across many distributed and disconnected databases with overlapping data sets, operating asynchronously. It does this in an environment where communications between processors is rare and relatively random.

Field computing systems have historically been intended to capture data for central computing facilities. CAO stresses decision support on field distributed computers, above raw data collection for central machines.

Timely decision support is not possible without practical access to day to day information. This is complicated by the high cost of on line communications to field staff. The CAO network minimizes communication costs by exploiting communication opportunities when they occur, usually once a day.

Conventional networks control record updates by restricting access to one station at a time. This is impossible in the CAO scenario, resulting in the need to combine edits and to reconcile conflicting data (data collisions). Communication events can also result in variable delays in the sequencing of processing each transaction, which may result in transactions being processed out of sequence (timing exceptions). CAO handles these complications in a manner practical to the oil field environment.

An Amoco Canada pilot CAO installation began operation on February 10, 1989 in the Pembina District near Drayton Valley, Alberta. It ceased operation in December of 1989 after a successful test. The pilot field facilities do not have telephone communication, so hand held computers were used as the medium through which the network transfers data. The pilot proved the methods developed in this thesis.

ACKNOWLEDGMENTS

I thank Garth Leavitt, who joined the project as a contract analyst but later became an employee of Amoco Canada. Garth's CAO involvement began after the initial system design, but he became a full partner during the long months of detailed design, development, installation, and testing of the system. I would also like to thank Erwin Spitzer and Long Nguyen of Minerva Technology Inc., for their early participation in system development. Special thanks also to the management and staff of the Pembina District for their enthusiastic support, participation, and patience throughout the pilot.

Special thanks are reserved for Dr. S. T. Nicholls, my research supervisor at the University of Calgary, for his support, suggestions, and assistance throughout my studies and this thesis project.

I wish also to thank the many professionals of the Production Systems Group, and Information Services Development and Support - Production, who have offered expert advice from time to time.

I am indebted to the Amoco Corporation, and affiliates Amoco Production Company and Amoco Canada Petroleum Company Ltd., for permission to use this project as a basis for thesis and to disclose the details contained herein.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1 INTRODUCTION

> "Each problem has hidden in it an opportunity so powerful that it literally dwarfs the problem. The greatest success stories were created by people who recognized a problem and turned it into an opportunity."
>
> Joseph Sugarman

The fall in oil prices of 1986 motivated most oil and gas production companies to adapt business policies to promote an increase in productivity in a harsh business environment. Section 2.4 provides background information for the business motivations, but it can be stated here that field computing was seen to have significant benefits to the corporate organization. The benefits were seen to come in three major ways:

1) more informed decision making at source,

2) increased staff productivity, and

3) more efficient and accurate data gathering.

Throughout 1986 and 1987, the author became involved in several initiatives to exploit computing technology in the field to meet these business objectives. Low cost portable PCs and hand held computers made it possible to provide processing power in the field. However, the author recognized a common problem to all the proposed initiatives, namely the absence of electronic information flow back to the field. Without dual direction data movement, the decision support initiatives would be less efficient due to limited data access and constraints to application capabilities.

The author proposed new methods believed to be a practical solution to these shortcomings, and coined the term Computer Assisted Operations (CAO). A CAO pilot was proposed to Amoco Canada Management to test these methods, which was approved in early 1988; development began in March of that year.

The CAO proposal was to use stand alone PCs and hand held computers for local processing. Each processor (called a "station") would be capable of exploiting any communication session to pro-actively communicate newly acquired or modified data to the CAO network, and to receive, from the network, data generated elsewhere that is within the bounds of the local data set. Each station relies on other stations in the network to pro-actively report new and modified data, and on the network to deliver it.

The role of the CAO network described in the preceding paragraph is the reverse of conventional network or data sharing schemes, where convenient communication is assumed and data is only provided on demand. Thus the CAO network should not be considered in terms normally associated with computer networks.

The CAO network is not an on-line system; continuous communications are too expensive given the connectivity logistics of most field operations. Instead, CAO exploits occasional communication opportunities to transfer data and reconcile databases. It is specifically designed as a computer networking alternative for installations where continuous computer communication (normally a prerequisite of conventional networking) is not economical.

In practice, communications are typically established once per day, providing for daily updating of databases on the network. For most situations, daily information flow is timely and represents a substantial improvement over existing manual or hard-copy systems.

1.1 Research Issues Addressed

The thesis proposes and tests new methods to maintain practical synchronization of databases across many disconnected and physically remote processors, in an environment where communications between processors is rare and relatively random.

Specific items addressed in this thesis include:

- automatic routing of information to individual stations,
- coordinating and distributing routine database functions, such as edits and deletions, across the distributed and disconnected databases,
- handling multiple sources of single data items,
- detecting and handling data conflicts,
- detecting and handling network transactions received out of sequence,

- balancing user "ownership" and control of data with the joint needs of other users on the network, and
- importing of data from "special" sources for handling by the CAO network.

1.2 Thesis Scope

The thesis defines the CAO system concept, and then describes a pilot installation where the methods were built and tested. Chapter 2 provides background on CAO, including its basic definition, business role, justification, and history within the Amoco organization. Chapters 3 through 5 describe the CAO hierarchy, networking, and transaction processing. The CAO methods of data control and acquisition are discussed in Chapters 6 and 7. Chapter 8 details the various hardware devices and explains the choice of the MS-DOS (Microsoft Disk Operating System) platform for CAO implementation. Chapters 9 and 10 present the CAO pilot and its results. Chapter 11 gives the conclusions, and Chapter 12 discusses the future of CAO.

1.3 Thesis Objectives

The following objectives form the substantive matter of this thesis project:

1) To develop the computer techniques and design the CAO network to function in the manner and environment described.

2) To build a prototype to prove the proposed methods are technically feasible.

3) To pilot test the prototype to determine if the methods are practical in actual field use.

4) To determine and quantify business benefits that are expected to accrue from CAO data transfer techniques.

5) To determine the impact on staff and their response to the CAO system.

1.4 Team Development and Author's Contribution

The thesis project was undertaken as a business initiative by the author's employer, Amoco Canada Petroleum Company Ltd. Amoco's standard business practice and the scope of this project required a team to develop and test the CAO technology. It is therefore important to identify the author's contribution.

The author conceived of the CAO data movement requirement and general techniques from individual efforts performed in 1986. The author then became involved with hand held data capture and was responsible for product identification and assessment of the hand held technology available in 1987, including systems engineering and hardware selection for projects requiring hand held computers.

In the latter part of 1987 the author independently developed the basic system design which was to become the subject matter of the thesis. The CAO hierarchy, networking, transaction processing, unique file types, edit handling, collision handling, and timing exception handling, were all worked out during this period. Convinced that the material had business (as well as technical) merit, the author developed the business case and began the process of obtaining project approval, funding, and resources. The project was approved in January of 1988 with the author assigned the role of project manager and engineer.

The approval of this project was coincidental to the pre merger effort between Dome and Amoco. As a result there were no available company programmers to staff the project. Consulting and contract staff (identified in the Acknowledgment) were therefore retained.

The author was responsible for leading the project design, project budgeting and scheduling, all hardware selection and ordering, and vendor development coordination for the unique CAO requirements. The author performed most of the system testing, arranged the installations and worked with Mr. Leavitt (see Acknowledgment) during the installations, trained the field staff, and was the first support contact for field staff. The author was responsible for assessing the field trials, although independent review and feedback by the District was encouraged. Most of the unique terminology of the CAO system was the author's creation. Virtually all of the programming, however, was performed by Mr. Leavitt and Mr. Nguyen.

CAO has met with some interest in the industry. The author presented a paper to the 40th Annual Technical Meeting of The Petroleum Society of CIM (Canadian Institute of Mining and Metallurgy)[1] at Banff, May 29, 1989. The Society of Petroleum Engineers (SPE) became aware of the paper and requested it be presented at the SPE Petroleum Computer Conference in San Antonio, June 28, 1989. (Copyright to the paper was retained by CIM, so the paper was not published in the SPE proceedings; unbound copies were distributed at the conference by the SPE.) The Foothills Chapter of the Canadian Society of Industrial Engineers also requested a presentation on CAO at the Quarterly Meeting in Calgary, September 20, 1989. The paper will be published in the Journal of Canadian Petroleum Technology by The Petroleum Society of CIM.

2 COMPUTER ASSISTED OPERATIONS

> "Think about it: almost every advance in art, cooking, medicine, agriculture, engineering, marketing, politics, education, and design has occurred when someone challenged the rules and tried another approach."
> Roger von Oech

Computer Assisted Operations grew out of a more general area of computing activity known within Amoco as "Field Computing". This chapter introduces CAO within this context. The traditional role of field computing, the early role of CAO, the need for data transfer in the field, and the business need for field operations computing are discussed in Sections 2.1 through 2.4. Section 2.5 discusses the large data handling burden faced by Amoco. Section 2.6 defines the group of employees specifically targeted as CAO clients. Finally, Section 2.7 provides a definition of the CAO concept.

2.1 Traditional Role of Field Computing

Over the last two decades, the term "field computing" has generally implied one of the following three types of computing:

1) **Data acquisition** programs that generally involve key punching data into central computer facilities for use by administrative, financial, and engineering staff.

2) **Supervisory, Control, And Data Acquisition (SCADA)** systems that use telemetry to monitor and control oil field measurement and process apparatus. Many SCADA systems are intended to acquire data more efficiently than the manual process. Other functions, like automatic control, analysis, allocations, and alarming, may also be provided (Giles and Wells, [2]).

3) **Technical** programs for field engineers and specialized job classes. These programs were typically available to persons having central computer access, since the required data was often available only from a "large system" database.

Field computing systems of the first two types have historically captured data for central systems. Little emphasis has been placed on information access by field staff, particularly at locations remote from an area office. Field access is generally limited to printed reports or occasional remote terminal connection.

The technical computing systems have generally been too specialized, or too expensive, to aid the general field operations staff. This is changing with the deployment of PCs and programs which are simpler to use and understand. Access to the data required for analysis can still be a problem for remote systems.

2.2 The Early Role of CAO

Although highly beneficial, the traditional role of field computing generally did not serve the needs of field operations staff. Before the introduction of the PC, technical constraints made remote or mobile field computing too expensive to be practical. Even with the PC there was a general lack of awareness to the benefits of field computing in the oil industry.

The severe oil industry economic climate of 1986 forced many companies to find new ways of cutting costs and generating revenues. Field computing was significantly impacted in this time period; accelerating the introduction of computing technology to field staff was seen as a very cost effective means for profitability.

It was clear the role of field computing would also have to change. Data acquisition, telemetry control, and specialized technical computing are still necessary, but supplying computer technology directly to field staff for decision support became a priority. To emphasize the direct application of computers to the general benefit of operations staff, the author coined the term "Computer Assisted Operations".

Throughout 1986 and 1987, a CAO application to assist field staff in acquiring and analyzing routine production and operating information was developed. The application helped operators assess the day to day performance of their wells, and to balance the production, inventory, and disposition of the oil, water, and gas they produce. To meet corporate requirements, production data is electronically transferred to central office computing systems for month end accounting. This first application is called "Field Data Entry" (FDE) and was documented by G. R. Gill in 1987.[3]

After the Amoco - Dome merger of 1988, a new field system was designed to replace FDE, so that the production monitoring and accounting systems of the two companies could be combined. This new program was named Production Operator's Work station And Reporting system (POWAR). The change over to POWAR occurred in 1989, about the same time as the CAO system was being tested using FDE (see Chapter 10 for discussion on pilot timing). POWAR was designed to fit with the CAO networking technology, but none of this capability was developed in POWAR for initial implementation since the original distribution of POWAR was to occur before the pilot evaluation was completed. The intent is to take the CAO networking technology and build it into POWAR now that the technology has been proven.

2.3 The Need For Data Transfer Within CAO

Although highly successful, FDE was a PC application designed for keyboard entry of information (Johnson, [4]). It was not designed for data transfer between FDE PCs in the field. Nor was it designed for data transfer between field and office based systems

(except for month end transfer to the production accounting system, similar to the data acquisition role of field computing discussed in Section 2.1.)

Keyboard entry is inefficient since there is much information which gets entered into more than one computer. Often a computer generated printed report is used by an operator to gather numbers to key into his PC. This can lead to transcription errors and month end reconciliation problems when duplicate data items disagree, or data is completely absent.

More serious than data errors is the data bottleneck which restricts the accessibility of information to remote PCs. This constrains system development and potential since many types of analysis require access to more data than is practical to reenter by hand. Electronic data transfer would eliminate these problems, while providing more timely and easy access to data. Electronic transfer of additional data types will lead to more and better field decision support applications.

The Field Operations Systems group, a subgroup of the Production Systems Group of Amoco Canada, was established in 1988, in part through the initiatives of the author, to manage, develop, and champion field computing endeavors such as CAO. Information access is a central theme to the purpose of the group, which jointly defined its vision as follows:

> "The Field Operations Systems group sees Amoco Canada becoming an organization where there are no impediments to operations staff routinely making all decisions that are most efficiently handled at the field level. Operations staff will have the tools to access and analyze the information they require, from their regular work place, even when that place is not an office. Boundaries between information systems, if they exist at all, will be completely transparent to our staff. Individuals will be able to easily modify the systems to meet their changing needs, producing fewer, better reports for decision making. Data handling will be reduced by manage-by-exception design, and the wide spread use of expert systems will further improve the quality of decisions."

The CAO pilot (see Chapter 9) modified FDE for data transfer using the techniques described in this thesis.

2.4 Business Need For Field Operations Computing

The impact of computing on profitability was recognized by others in the company as well. After down-sizing and restructuring in 1986, the company turned its attention to computing opportunities in 1987. A high level study called the Computer Applications Technology Committee (CATC) was commissioned for Amoco Production Company and all its subsidiaries, including Amoco Canada. CATC concluded the following in 1987:

> 1) "All aspects of our business are considered candidates for computerized improvement"
>
> 2) "Management strongly believes that increased usage in professional and operating functions will result in higher quality answers in shorter time periods by fewer staff"

The Amoco Production Company Strategic Systems Plan for 1988 (confidential to Amoco) was more specific in its conclusions:

> 1) "(The) highest (computer) opportunities exist in Natural Gas and Field Production due to a lack of spending to develop decision support capabilities for these areas in the past."
>
> 2) "Production support for the field (gather data and analyze operations) will improve decision making by over $1.0 Billion annually."

The $1.0 Billion figure stated above is in US dollars and includes all of Amoco Production Company, of which Amoco Canada is a subsidiary.

The study and strategic plan set a clear direction for the development and deployment of field computing systems within Amoco Production Company.

The CATC initiated a study called the Computer Applications for Production and Profitability (CAPP) which identified specific computing opportunities in various disciplines, including field automation (SCADA), but failed to address directly the issue of field operations computing. The CAPP study was also undermined by changes in the business environment and continued company restructuring which changed many of the assumptions on which CAPP was based. In the first half of 1989, a second study was undertaken to reevaluate the CAPP study in a review dubbed ReCAPP.

ReCAPP was also intended to correct a deficiency in the CAPP study which became apparent after its release. Although the original CAPP study had identified field computing as the leading area of opportunity, it did not include field representation in its team make up. Accordingly, the ReCAPP study was structured differently to ensure field input to the process. Thus for the first time, the full scope of field operations computing was studied within Amoco Production Company.

The ReCAPP Field Operations Team Report (confidential to Amoco), released in May of 1989, drew several important conclusions:

1) "Computing does have a significant role to play in the management of field operations."

2) "Field operations are manpower constrained, not opportunity constrained."

3) "Need to implement the combination of easy access to data and better field-oriented analysis tools along with training/support to achieve maximum benefits."

4) "Examples of how computers can add value to field operations (include) programs to improve inventory control, controlling operating costs better and optimizing equipment run-time and performance. By saving time, field foremen can spend more time in the field to improve safety and control quality. In addition, all field employees will have more time and tools to analyze problems and find better solutions."

5) "Most field people are eager to learn about computers and how they can be used to improve job performance."

6) "The opportunity for field personnel to apply the latest computing technology to enhance their job performance is a positive step in making Amoco a better place to work, leads to overall job enrichment and improves morale."

The author participated in the field operations portion of the ReCAPP study, where CAO received recognition as one of several initiatives with excellent field computing potential.

Clearly field computing is a dominant factor in the way Amoco will manage its production facilities in the new business environment. There is a wide variety of applications with the potential to improve the profitability of Amoco.

CAO is the Amoco Canada approach to the basic corporate strategies outlined in the identified reports. CAO is tailored to the unique Canadian aspects of our business, including the cost structure in communication systems, regulatory and reporting requirements, and local operating philosophies.

2.5 Data Handling Burden

Any discussion on the benefits of field computing would be incomplete without a discussion of the large quantities of data which must be captured and processed at the field.

Increased attention to accounting practices, production and measurement tolerances, along with stricter regulatory requirements, is increasing the information gathering burden on oil and gas production companies. The complexity of computation for production facilities monitoring and product balancing is also increasing. Computer based systems are now routinely required to meet the challenge.

Amoco Canada has substantial motivation to streamline the data processing due to the large amount of data regularly handled. Amoco operates 14,782 oil and gas production wells; water injection, source, and disposal wells; and gas or nitrogen injection wells. Of this total, 11,470 require regular monthly reporting. Besides the large number of wells, Amoco Canada operates approximately 700 multi-well batteries, 1,800 single-well batteries, 72 gas plants and 91 gas gathering systems. Each of these entities require reporting beyond the 11,470 wells connected to them. (All numbers reflect operating status at November, 1989.)

To put this into perspective, Amoco produces 1,900 S-Reports each month for reporting to the Alberta Government Energy Resources Conservation Board (ERCB).

(S-Reports are a group of standard production and handling reports defined by the ERCB.) This is approximately 4,000 pages of information that report net monthly production and disposition of oil, water, gas, and their constituent components (if separated). Copies of these reports also go to working interest partners of each facility. The multiplier effect results in greater than 21,000 S-Report copies, or greater than 42,000 pages, generated and distributed each month by Amoco. (Most reports are sent electronically to the interested parties; the "pages" referred to are generally the electronic equivalent.)

Besides the above, approximately 600 EM-Reports (similar to the Alberta S-Reports) are submitted to the Saskatchewan authorities for monthly reporting, along with an unknown number of copies to working interest partners.

The S-Reports and EM-Reports are calculated from a very large number of individual transactions. No study has been undertaken to accurately quantify the total number of transactions handled each month, but a conservative estimate can be reached by considering the amount of data produced on one large, conventional oil production facility. One area with conventional oil production records approximately 4,600 transactions each month for 600 producing oil wells and related facilities. Applying the average ratio of 7.6 transactions per producing oil well, we can estimate the total number of transactions captured at the field as approximately 87,000 (11,470 wells requiring regular monthly reporting multiplied by 7.6 transactions per month per producing well).

It must be remembered that these transactions include only the necessary information to meet minimum accounting needs for production reporting and revenue handling. It does not provide for the larger amount of data necessary for operations and technical staff to monitor the wells performance and generate operating and engineering strategies.

Finally, there are other areas of our business that place data acquisition burdens on our field staff which are not included in the above figures. For example, special technical tests on wells and facilities, maintenance and construction costs, construction scheduling and reporting, materials warehousing and tracking, shift scheduling, time sheet reporting, vehicle reporting, safety milestone tracking and reporting, meter calibrations recording and witnessing, invoice verification and payments, budgeting and expense statements, cathodic protection surveys, site specific inventories, meter seal records, fluid analysis, and other areas.

2.6 CAO Client Base

The term "field staff" can mean any of a wide range of job functions. In CAO, the targeted staff have historically gone without direct computer support. These staff form the CAO client base, also known as the CAO user community.

The CAO client base is identified as the operational staff of the Production Department; and includes as primary clients the production, maintenance, diagnostic, and construction staff responsible for the day to day operation of oil and gas production facilities and related equipment. Operations supervisory staff are included as primary customers.

A secondary client base includes the administrative and engineering staff who directly interface with field operations personnel and therefore use the CAO applications developed for the primary client base.

2.7 CAO Defined

Computer Assisted Operations is a concept which sees a computing environment for users where every defined information need is available to the user, in a format tailored to his job function. CAO seeks a computing environment for operations staff where the information processing tools *relevant to their jobs* is available on a computer at their job site. These tools require consistent user interface and appear to work together for the good of the operator. Data which is acquired by the operator, or generated elsewhere, appears through the interface of his computing tools in a consistent manner, regardless of where or how the data was collected. The operator is not be burdened with the details of the system which originated the data, nor is he burdened by the need to pass on information he has acquired or modified. Those functions are handled automatically for him by the CAO framework.

The greatest challenge comes from making the diversity of computing sources and data repositories appear transparent to the user. The CAO framework developed in this thesis is the method to provide this transparent movement of data. CAO seeks a consistent user environment, but it does not necessarily seek the integration of programs; CAO does seek the integration of data.

The CAO framework is a practical approach to networking field computing systems for multi-directional information flow. CAO is neither an application nor a system; it is a **framework** which provides the multi-directional **flow** of information necessary to sustain effective **decision support** in the work place of field employees.

CAO is the computing architecture where applications running on separate field processors can communicate and share data in a practical integrated manner. The CAO framework provides transaction processing across many distributed and disconnected databases with overlapping data sets, operating asynchronously. CAO operates as a distributed database, in a computing environment where communications between processors is rare and relatively random.

The CAO framework achieves practical synchronization of field databases through combination of its three basic elements: hierarchy, transaction processing, and networking. When data items enter the network, a copy of the data is automatically sent to all systems in the network where it is needed. Edited or deleted data items are also communicated within the network.

CAO does not require continuous communications. CAO requires only that occasional communications connections are possible, usually once each day. This reduces the cost of connectivity, thereby enhancing the economic benefits of field computing.

Computing applications suitable for CAO include most forms of field computing, present and anticipated, where information must be shared or communicated. Applications which have been identified for possible CAO implementation include production accounting, production strategy, well management, well work over design, work order systems, field messaging, cost monitoring, dynamometer testing, hourly staff time sheet reporting, shift scheduling, vehicle reports, safety milestone awards monitoring, preventative and predictive maintenance scheduling, construction scheduling and materials tracking, and plant log sheets.

CAO is not suitable for applications requiring instantaneous transmission of information; for example, operating alarms. Such needs can only be met using systems that have communications to pass alarms within acceptable time limits. However, the CAO concept does make provision for sharing routine information with telemetry systems on an opportunity basis.

2.8 CAO Business Objectives

The business objectives of the CAO initiative are:

1) To build an integrated framework to consolidate diverse operations applications, computers, and physical locations into a common network structure;

2) To use low cost communications alternatives to achieve the framework identified above;

3) To stimulate the development and deployment of decision support tools to operations staff;

4) To eliminate manual handling, redundant entry, and re-work of production and technical data;

5) To improve the accessibility and timeliness of operations data to all users;

6) To consolidate field data so that operational, technical, and financial analysis is performed on consistent information; and

7) To improve the quality of information gathered, shared, and processed.

3 CAO HIERARCHY

> "Order is heav'n's first law."
> Alexander Pope

To meet the needs of present and anticipated data transfer requirements, a CAO hierarchy was proposed that supports the traditional field computing roles identified in Section 2.1, and the requirements of the CAO concept outlined in Section 2.7.

The CAO hierarchy is shown in Figure 1. Each block in the diagram represents a particular type of CAO "station". The station types are conceptual in nature, identified by the role they play in the hierarchy. There are four primary station types which are arranged in distinct layers of subordination. They are the Production Data Resource (PDR), CAO host, work station, and Remote Entry Unit (REU). The population of stations in each layer generally increases as the layers descend, forming a pyramid.

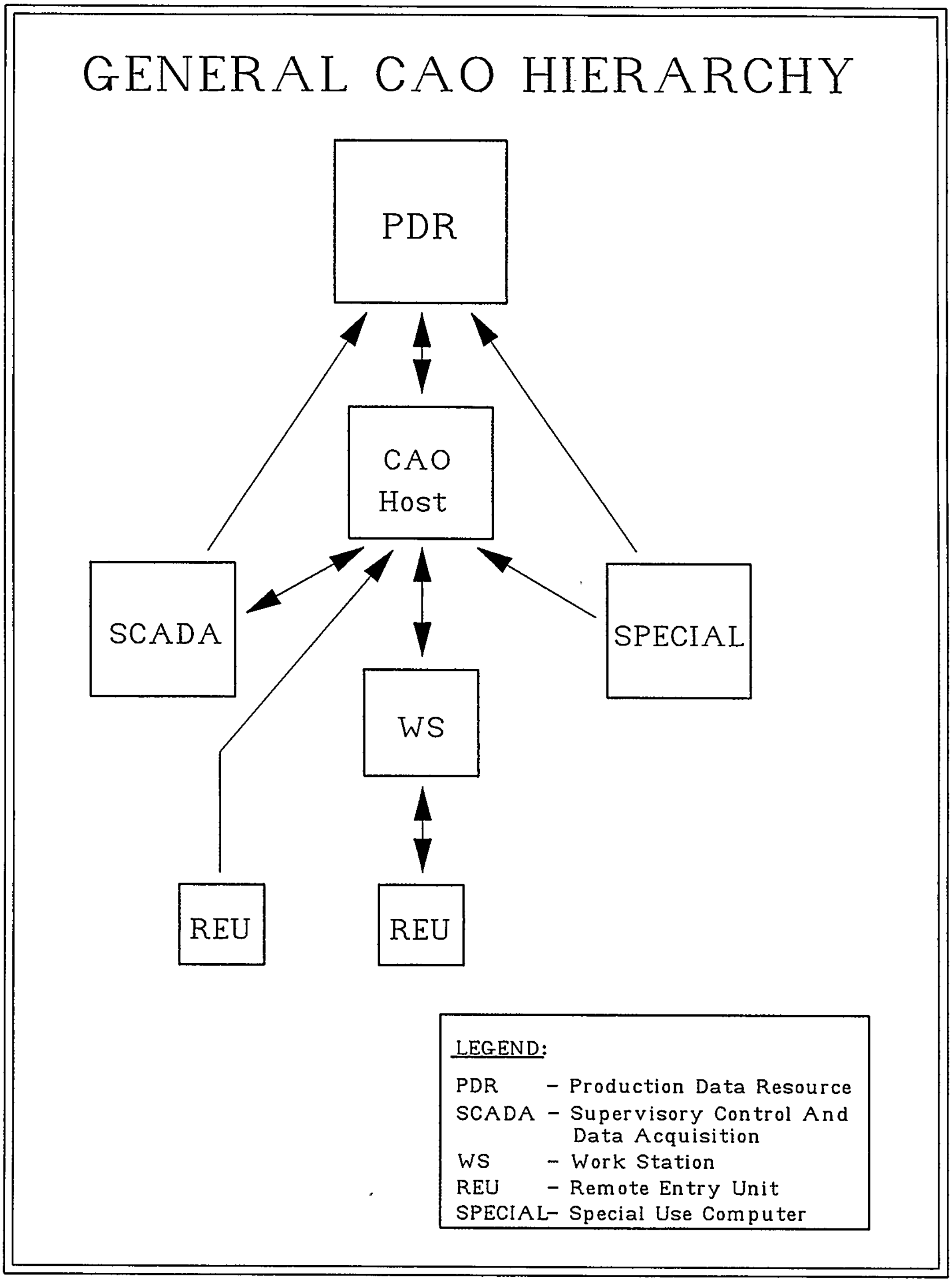

Figure 1: General CAO Hierarchy

Two other station types are shown, the Supervisory Control And Data Acquisition (SCADA), and the Special station. These are not truly CAO stations since it is not intended that they be built to rigorous CAO standards. But they are included in the hierarchy to highlight the desirability of transferring information between these station types and the CAO network.

Each of the primary stations is required to coordinate the flow of information between related stations beneath it in the hierarchy, and to act as an information gateway to the rest of the network. In the CAO hierarchy, information generally does not flow directly between stations in the same level. Instead, information flow is coordinated by the immediately overseeing station. Each station in the network retains, in its databases, a copy of the information required by the local users. Excellent response times are therefore achieved locally while avoiding high communication costs.

It is not necessary that all data items flow everywhere throughout the hierarchy. Information that is required elsewhere is passed on through the network, but data items which are highly localized are retained in the local station and are not forwarded. Information is not passed unless doing so provides value to other stations in the hierarchy, as determined by the application design team.

Discussion of the CAO hierarchy is not meant to imply a large installed user base in complete conformity to the discussion following. The hierarchy is intended to provide structure consistent with the data transfer and local user operating needs. However, all CAO hardware has been selected to be consistent with this hierarchy. Further, the existing field applications targeted for CAO data transfer capabilities are developed consistent with the hierarchy.

3.1 Production Data Resource (PDR)

The highest level in the hierarchy, the PDR, is a collection of databases and applications on the main frame computing system. The PDR acts as a central repository for all data which must be shared throughout the corporation. The PDR provides analysis, reporting, and processing of information at the central office level. The PDR also carries out the transaction processing of data items between the main frame systems and the CAO hosts.

The PDR has been defined in concept to recognize the need for data transfer at the main frame level. It has not been fully defined and is not constructed at this point in the CAO development, since demand for CAO capability at the main frame will not appear until CAO functionality is substantially developed at the lower levels. Accordingly, we have concentrated our resources on developing the lower levels of the hierarchy.

3.2 CAO Host

Each field area office on the CAO network has at least one CAO host. The CAO host provides centralized storage of area data and coordinates the information flow between the field located processors and the PDR. This is a large task since almost every data item generated in the field area is processed by the CAO host.

Although all stations beneath the CAO host in the hierarchy can receive and process information, new information is considered preliminary until it has been passed to the host, processed, and sent to the relevant CAO stations. Processing of the information by the host is handled in batch form, usually once a day.

The CAO host retains data from all stations beneath it in the hierarchy. It is therefore capable of providing the same information to area office staff as is available to field staff. The host is further capable of consolidating information for the operating area from its centralized database.

See Section 8.3 for a discussion of CAO host hardware requirements.

3.3 CAO Work Station

The next lowest level in the CAO hierarchy is the CAO work station. The CAO work station is a field located processor which provides decision support to operations staff. Usually, but not always, the CAO work station is located at a central battery or compressor station for convenient access by more than one operator.

CAO work stations are the work horse for field application processing They provide applications and applicable data to a small number of target users.

CAO work station hardware requirements are given in Section 8.4.

3.4 Remote Entry Unit (REU)

The REU constitutes the lowest level of the CAO hierarchy. The REU is primarily a simple data entry device which field staff can take any where their work activity requires. Some limited processing is also provided by the REU in certain applications. REUs must be highly portable and are typically hand held or lap top computers.

In its role as a data input device there are no restrictions placed on the relationship of REUs to work stations; any work station accepts information from any REU. The work station passes the data through the network to the CAO host for handling and, if required locally, retains a copy in its database.

See Section 8.5 for a discussion of the REU chosen for the CAO pilot.

3.5 Special Systems

Special systems are non CAO programs or computer systems with which it is desirable to share information.

Generally these are specialized accounting or technical systems which are purchased from vendors and hence do not conform to CAO standards.

Despite the specialized nature of these types of systems, it is typical that data sets overlap. CAO seeks to eliminate redundant entry of data by providing bulk transfer of information to and from these systems. See Section 5.5.

Examples of special systems include such things as gas chart integration programs (where paper charts of recorded process variables are converted to quantities of material for measurement purposes), inventory programs, field diagnostic programs, specialized function computers such as lab analysis, scale house acquisition, and accounting programs.

3.6 Supervisory, Control, And Data Acquisition Systems

A special relationship exists between SCADA and the CAO hierarchy. SCADA is a major player in field computing systems, and is Amoco's preferred method of data acquisition wherever economical. Most SCADA systems are complex networks controlled by a master computer. CAO incorporates master SCADA computers into the network similarly to the special systems described in Section 3.5.

The field telemetry portion of SCADA systems are not included in the CAO framework, but are indirectly accessible through the SCADA master computer. SCADA systems on the CAO network send and receive data to and from the CAO host. Using the CAO host as a gateway to the CAO network simplifies the network interface minimizing impact on the already complex SCADA system.

Multi-directional data transfer between SCADA and the rest of the CAO network closes the information loop between field and office systems. Two way data transfer has a variety of applications where office systems can be used to perform decision analysis. The resulting decisions are carried out under automatic SCADA control.

4 CAO NETWORK

> "The significant problems we face cannot be solved at the same level of thinking we were at when we created them."
>
> Albert Einstein

The CAO network is unlike most conventional networks. As discussed in Chapter 1 the CAO network assumes communications is generally unavailable, thereby forcing network nodes to pro-actively route data to the relevant stations. Information is transferred on the network at the transaction level. Keys imbedded in the transaction are used to identify what the information is. Other keys in the transaction are used by the network to relate the transaction to established entities, and thereby determine which station needs a copy of the transaction. Oil field facilities and job functions are well defined and therefore lend themselves to the structured CAO network.

Although the CAO network is unlike conventional networks, the interrelation between CAO stations or nodes satisfies the ANSI definition of a network (ANSI/IEEE standard 100-1984, [5]).

Each CAO station is uniquely distinguished on the network. Every application on the CAO host has access to relational tables that associate all possible data elements with one or more CAO stations. For example, a master well file lists the identity of the stations which oversee each well. The applications use the relational tables to address information to the stations. Chapter 6 discusses the control files which provide the relational tables for transaction routing.

The CAO network function handles the details of routing data to the required stations through the physical channels available. The system concept applies to several common communication media, from diskettes to satellites. Selection of a particular communication medium is based on economics and local business needs.

Communication sessions can be grouped into two types:

1) **Host-controlled:** host directly controls the other station in a master-slave relationship, and

2) **Remote-controlled:** host sits in a "listen" mode waiting to service incoming requests initiated and controlled by the other station.

The communication category is transparent to CAO applications and is handled as part of the network operation. Often the communication is automatic for both categories. In a minority of cases, users are required to manually control the communication sequence.

In the host-controlled category, network communications are sequenced within the batch process. The host sequentially polls each station for its new or modified information, processes the data, and sends the data out to each waiting slave station thereby completing the process.

In the remote-controlled category, the remote initiates and controls all aspects of the communication session. The connecting station is obligated to deposit at the host all new data it has acquired and request any new data files addressed to it. The host acts as a simple file server during each session. When the host batch processes data, it looks to its own disk for files reported from remote stations, and deposits outgoing files (called "transit" files) on its disk for later retrieval by the remote stations.

It would be ideal to have the remote station interact directly with the databases residing on the CAO host. However, this is not cost effective in the single tasking MS-DOS environment.

4.1 Station Identification

Every station in the CAO network is assigned a unique four character Station Identity Code (SIC) to distinguish it from all other stations on the network. The SIC is used primarily for addressing information across the network. It is also used to tag each new data item so that the originating source and the most recent modifying station are identified. This assists in operational trouble shooting, data audits, and data maintenance.

The SIC is actually a four character alpha-numeric code. Within CAO, alpha character keys are restricted to upper case only. Thus with 26 alpha characters and 10 numeric characters, the theoretical maximum number of stations which can populate the network is 36^4 or 1,679,616.

There is no restriction placed on the SIC assigned to any station, other than the requirement for uniqueness. For administrative and convenience purposes, however, certain conventions have been established for assigning the SIC numbers. Referring to Figure 2 "Station Identity Code Convention", there are three basic components in the SIC structure.

1) **District Code:** Amoco Production Company assigns a two digit code to each of its Districts in all its subsidiary companies. The code uniquely identifies a District throughout Amoco's world wide operation. Amoco Canada has six Districts and the Calgary Region office. It happens that the least significant digit of the codes assigned to each Amoco Canada District and Calgary is unique, meaning only the least significant digit is required to uniquely identify a District or Calgary within Canada. The first character of the SIC is assigned the least significant digit of the district code.

   The District Codes are as follows:

   57 - Brooks

   58 - Grande Prairie

   60 - Calgary

   75 - Lloydminster

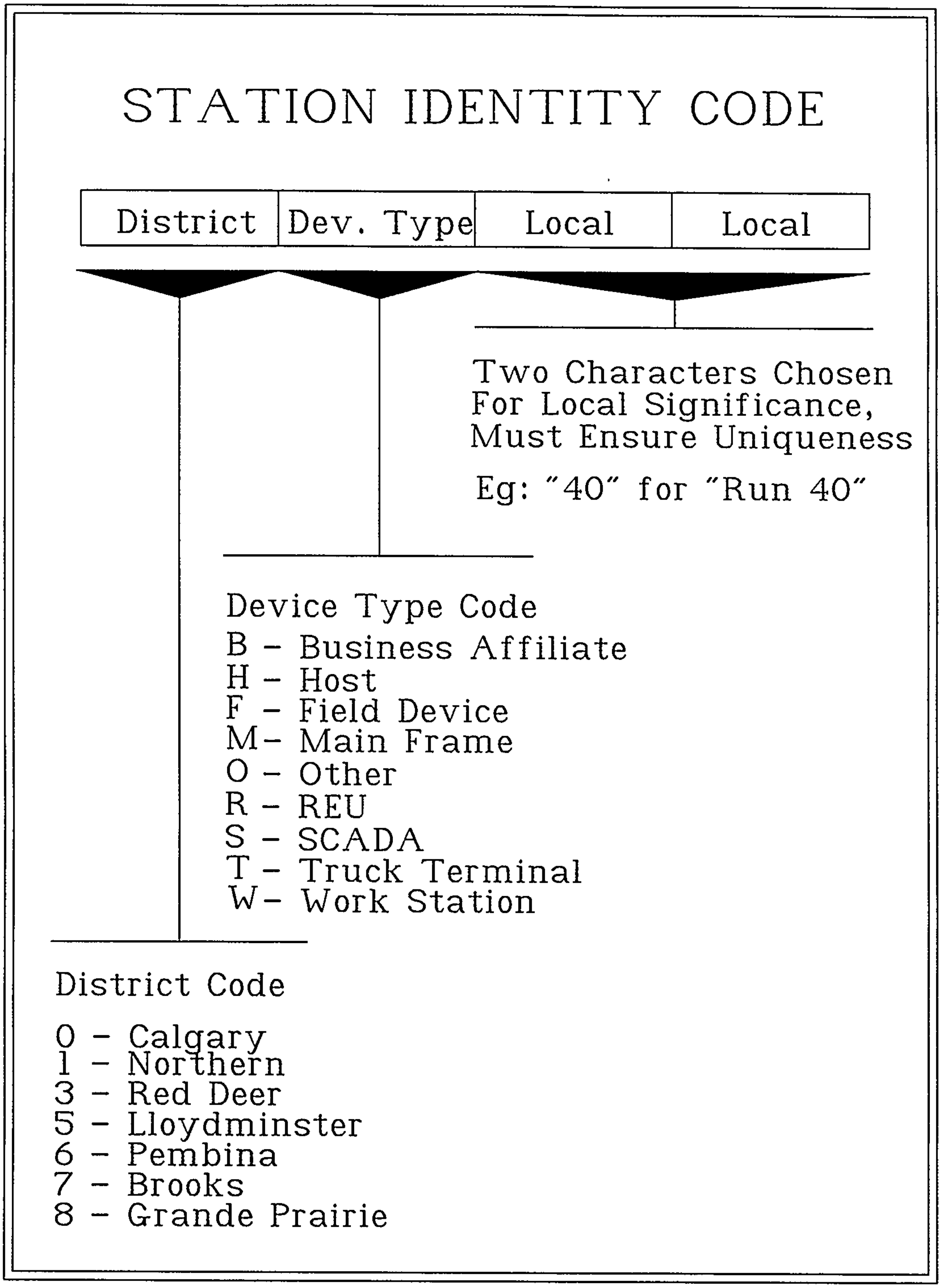

Figure 2: Station Identity Code Convention

91 - Northern Operations (formerly Edmonton District)

93 - Red Deer

96 - Pembina

Using the least significant digit of the District code as the the first character in the SIC allows decentralization of SIC administration to each District. SICs can be assigned locally without fear of conflict with any SIC assigned in other Districts.

The Calgary District Code is a misnomer, since Calgary is the Region office, and not an operating District. A District code is assigned and is used to represent Calgary head office identification without any ambiguity with the operating Districts. CAO uses the District code to identify CAO stations representing Region functions.

2) **Device Type:** One character in the code is reserved to identify the type of device which originated or modified the data. Assigned codes are as follows:

H - Host

M - Main Frame

O - Other

R - Remote Entry Unit

S - SCADA System

T - Truck Terminal

W - Work Station

Other codes reserved for future use include:

B - Business Affiliate

F - Field Device

3) **Local:** Two characters are reserved for local assignment and can have any local significance.

Examples: The SIC for an REU on operating run 37 in the Pembina District may be "6R37". The SIC for a SCADA computer in the Nipisi field of Northern Operations may be "1SNI".

Once assigned to a station, the SIC is considered to belong to the operational "run" or field functional assignment. It cannot arbitrarily be reassigned to another station or operating function without upsetting the data routing of the CAO network. The SIC is assigned to the work station of that run. Thus if the hardware is replaced, the SIC must be entered into the new hardware and altered in the hardware removed from the run.

The SIC is instrumental in the network routing control of all transactions. See Section 6.8 for a discussion on network routing control.

Application level programming is not burdened by the physical details of the CAO network. Applications are able to pass individual data items to specific stations by depositing the data into special files addressed for the desired station. These special files are called "transit" files and, once data has been deposited in them, are not disturbed by any other process until they arrive at the addressed station. To address a file, an application simply imbeds the desired station identity code in the file name. Additional characters in the file name identify the application and file type.

4.2 The REU As Communication Medium

The CAO network is designed to transfer all information electronically and can use any common communication medium such as diskette, telephone, radio, and coax. In many work station locations, conventional communication facilities are not available. CAO provides communication through REUs as a unique option for data transfer. Using the REU as the data link can be cost effective if operators are already provided with REUs for data entry.

Hand held computers are available in the market place which are highly portable and have good memory capacity (8KB to 4MB). Hand held computers suitable for use as REUs in CAO can be individually selected by address for communication through a multiplexor connected to a PC. These features, combined with the daily activities of field operators at some locations, permit the use of REUs for network communication in CAO.

Each morning operators gather at the field area office for a morning review with their field foreman and to pick up any new information relating to their area. They then drive to their operating "runs" to review facilities, gather data, and perform minor maintenance. At least once each day they access their field located work station to complete daily computing activities. Finally, they return to the field area office to report the days information and to have a daily wrap up.

CAO uses the REU as a temporary data repository for transfer of information between the CAO host and CAO work stations. The operators carry the REU between the host and the work station as part of their normal routine. Connection of the REU to either station is a simple plug connection and the data transfer sequence is fully automated.

For addressing purposes, the CAO host relates one specific REU to one specific CAO work station. Each REU that is used as a communication go-between with a specific work station is called the "designated REU" of the work station.

Transit files addressed for a specific work station are automatically placed by the host in its designated REU. All work stations automatically check the station identity code of each REU when it connects to determine if it is its designated REU. If so, the work station reads any transit files addressed with its SIC and deposits in the REU all new or modified data for host processing. New or modified data being communicated to the host is contained in a special class of file called a "network" file. See Section 5.1.

Not every REU is related to a work station since there are typically more operators than there are field PCs. REUs which are not related to any specific work station are termed "free REUs".

At the field area office, the REUs are connected to the CAO host through a multiplexer, allowing the host full communication control over the REUs (consistent with the host-controlled communication discussed in Chapter 4).

In late 1987 and early 1988, most of the hand held computer products on the market handled communications in the remote-controlled mode. This is due to the large customer base the hand held vendors have in the areas of route accounting, warehousing, and marketing, where employees often use the telephone or other communications techniques to call a waiting host and initiate communications at random times in the day.

In Amoco's case, it is more cost effective to use host-controlled communications. Operations staff generally don't have communications available until they return to the office at the end of their shift. It is a waste of time to have each one of them connect to the PC and wait while the communication session is completed (as would be necessary in remote-controlled communications). In host-controlled communications, they simply place their hand held unit in a multiplexer and leave. All the communications are then handled by the PC in a master mode during a batch process. When the operators arrive the following morning their REUs are ready and waiting for them.

Daily data gathered in each REU, along with information from the work stations, is automatically read and processed by the CAO host. After processing, the host deposits in the designated REUs the transit files addressed to the associated work stations.

A work station can communicate its network information to the host, through the REU, in one of two ways:

1) The work station can simply deposit its data in a special file in the REU's memory for retrieval by the host. The REU application does not recognize nor use those special files, thus preserving data security and integrity.

2) The work station can transfer ownership of the data to the REU by combining its data with the REU data.

This allows review and edit of the work station supplied data on the REU by the REU application. This method of communicating information is a transfer of ownership of data, and not the networking of an addressed file.

4.3 Handling Communicated Files

Addressed data files on the network are subject to certain rules which assure the files arrive at their intended destination. Simply put, once a file has been addressed it is never deleted until it is received by the station to which it was addressed and has been properly processed.

Other stations may create these files, or add to them by appending transactions to the bottom of the file (see Chapter 5), but they cannot modify the existing contents, delete records, nor delete the file. The file must be forwarded at the earliest convenient communications opportunity.

Communications between REUs and work stations (or hosts) are not "networked" in the conventional sense within CAO and are treated differently than addressed files. See Section 7.2.

4.4 Network Timing Concerns

In the CAO scenario, communications are undertaken at times of opportunity and at the convenience of the operations staff. The network therefore has very little control over the timing of receipt and delivery of transactions at each station. It is very possible for a transaction effecting two work stations to be received by one, edited, returned to the host, and received again, before the second work station even receives the initial transaction.

In most cases, the time delays between communications sessions effect only the timeliness of data received and processed. However, the asynchronous nature of the network can lead to processing conflicts when multiple edits are performed on different stations, with differing transit times, so the edits arrive at a station out of order. This is called a "timing exception", which is the attempt to process a transaction against a data set which contains a matching transaction dated more recent than the incoming transaction.

The strict enforcement of the CAO hierarchy partially alleviates this concern by creating a focal point of synchronization in the network; all communications in the network are controlled and reconciled by an entity of higher level in the hierarchy.

The hierarchy alone does not guarantee resolution of the timing problems. Three other processes are combined with the hierarchy action to moderate the impact of timing exceptions:

1) Special rules for sending and receiving transit files (see Sections 5.8 and 5.9).

2) Special processing logic during the transaction processing (see Section 5.4).

3) Reporting of timing exceptions (see Section 5.6).

4.5 Data Compression and Mail Files

Communication between processors is constrained by several factors. For REU communications, the constraining factor is the available memory of the device. Modem transfer via telephone is constrained by long distance charges and staff time delays. It is desirable to minimize both of these factors. Modem transfer by any means, telephone or radio, is susceptible to line impairments. Remote sites, where conditions are typically substandard, are especially susceptible.

Minimizing memory requirements, transmit times, and staff delays all share a common cure, namely minimization of file size. Reducing file size also improves the likelihood of success over communication channels where impairments are transitory.

Amoco has selected a commercial PC file compression program for general corporate use. Testing of CAO transit files suggests file compression of between 50% and 70% is possible using the selected program. Unfortunately, delays in obtaining a corporate-wide license prevented use of the program in the CAO pilot. Licensing is now assured and file compression will be used in later development.

One of the advantages of file compression programs is that several different files addressed for the same network station can be bundled together into one compressed file, then unbundled at the receiving station for processing of the constituent files. Thus the network need only route one file to each station. The file containing the compressed bundle of files for communication is called a "mail" file.

4.6 Logging Communication Error Events

It is intended that most of the communications events occur in a batch or unattended mode. This is particularly true of the CAO host. The CAO network logs all communication events to assist operators and systems support personnel. See Section 9.12 for an example of a communication error report generated from the log kept in the CAO pilot system.

5 TRANSACTION PROCESSING

CAO models the human and paper networks existing in conventional data handling scenarios. Each day the CAO work stations and REUs report new or modified operations data to the CAO host. The CAO host also mails to each work station any new or changed information relevant to that work station.

At least once each day, the CAO host batch processes the information to perform the "transaction processing". "Transaction processing" in CAO means that data records are dealt with one at a time, that each event is the smallest definable for a specific business purpose, and that each event is worked through the system to the appropriate user stations and applications. However, the term is used somewhat differently in CAO than in conventional transaction processing, such as airline reservation systems. In conventional transaction processing systems, users generally deal directly with a central database. In CAO, users deal with a local database which later communicates all new or changed data to the network. Central database processing does not occur until after data arrives at the CAO host, typically a few hours after entry.

Another term for the CAO style transaction processing is "reconciliation" processing. The term "reconciliation" applies since the host batch processing and the network ensure that (in the absence of newly initiated data edits) all applicable stations on the network will contain identical images of the same data records.

The CAO framework is designed on the principle that application data can enter the system at any CAO station. If a station does not recognize a data entity, it passes the record up the hierarchy until it reaches a level where enough system scope is collected to recognize the data. It is then processed like any other data item and distributed throughout the CAO network, presumably down the hierarchy, until it reaches all necessary repositories. In this way data can be entered anywhere in the system and still be processed correctly. Data entered into a station which has no local relevancy is called "external" data in CAO.

5.1 Electronic Data Transfer (EDT)

Electronic Data Transfer (EDT) is the communication of data in the broadest sense within CAO and associated special systems. EDT is the "data transfer" function referred to in Section 2.3. EDT within the CAO framework results from transaction processing, networking, SDI, and SDE, operating throughout the CAO hierarchy. EDT is significant by comparison to the manually intensive methods that is required in the absence of CAO data transfer techniques in the field.

EDT is generally reserved for the independent data acquired or modified throughout the network. There is generally little need to provide EDT capabilities for static or derived data since static data rarely changes and derived data is redundant. However, certain types of derived data may be communicated if they are calculated in special systems due to policy or complexity of the derivation calculations. See Section 5.5 for a discussion of data transfer from special systems, and Section 9.4 for an example of derived data treated as independent data.

A transfer is required when data is initially entered or is modified. Each individual station is responsible for tracking data which must be communicated. This is accomplished by the following classes of database files:

1) **Main** files are the database files retained in a station which are directly accessed by local applications for normal processing.

2) **Network** files are generated by each station on the CAO network which stores all new or modified data which must be communicated up the hierarchy for processing.

3) **Preliminary** files are generated in an REU which contain newly captured data. Preliminary files are inputs to both the host batch processing and work station data acquisition. Preliminary files may be used by the REU as its main processing file.

4) **Collection** files used by the host for processing of the incoming data from preliminary and network files, SDI transfers, and host keyboard edits. The host processes each incoming data event against the collection files.

5) **Transit** files include information resulting from the hosts transaction processing which is addressed

to a specific work station. Data in the transit file is used to add to or update the main file at the destination work station.

6) **Transaction** files include those files where a list of all transactions processed at a station are maintained in chronological order. See Section 5.6.

The main and network files are paired for each application database file containing independent dynamic data which must be communicated on the network. The main files store all data required for local processing. The network file contains only information which has been entered or modified since the last communication session with the network. To communicate all new and modified data to the network, an individual station simply passes its network file to the supervising station up the hierarchy and then deletes the contents of its network file locally; the cycle then repeats. Preliminary and network files are treated as equivalent inputs to the transaction processing at the host.

When a record is initially entered by a user, an identical copy is made in the main and network file. In this way the newly created data is passed on to the network for general distribution. See Chapter 7 for a detailed discussion of data acquisition.

5.2 Data Edits and Deletions

Transaction processing allows for data edits and deletions to be communicated across the network like regular data items. An edit transaction is distinguishable from a new data transaction by the presence of blank data fields. Only the modified data fields contain values in an edit record, while all non-modified fields are blank. Edits coincidentally made on different fields of the same record at different stations will be combined in the host. After processing, all occurrences of the record on the network will reflect the edits of all stations.

When a record in the main file is modified, those fields that have changed are copied into the network file; unchanged fields remain blank. If a corresponding record does not exist in the network file a new one is created and the changes are copied into it. In most cases when a modification is made, some of the fields in the network record remain blank (with the exception of the key fields). The data fields in the network files are character, rather than numeric, so that zeroes can be distinguished from blanks.

The CAO convention is to treat all blank fields as "null" fields. In other words, a station receiving an edit record will overwrite the fields in the existing record with the contents of the matching data fields from the edit record which are non blank. The data fields in the original record are not altered for those matching data fields in the edit record which contain blanks.

Figure 3 shows a simplified example of an edit to the "oil" field of a well test record.

A data deletion is signalled by the setting of a "delete flag" in the record to be deleted. The CAO convention is that a deletion notice always takes precedence over any other processing concern. Thus a deletion immediately causes the record to be deleted in the main file, with a copy of the deleted record made to the transaction file.

DATA EDIT - WELL TEST EXAMPLE

		Volumes		
		Oil	Water	Gas
WELL 1	89.08.01	10	4	0.23

Original Record in Main File

WELL 1	89.08.01	11		

Operator Revision

WELL 1	89.08.01	11		

Edit Transaction Communicated to Network

WELL 1	89.08.01	11	4	0.23

Resulting Record in Station Databases

Figure 3: Data Edit - Well Test Example

Should a matching record exist in the collection, network, or preliminary files, the record delete flag is immediately set but the transaction is not deleted from the files. Although this results in a redundant issue of the delete notice to the network, this is deemed appropriate since it provides an extra measure of confidence that any older versions of the record that may have been in transit (due to time delays on the network) will in fact be deleted when the second delete notice arrives.

If a station receives a delete notice for which it cannot find any matching record (for example, the record had already been deleted), it simply ignores the incoming delete transaction.

Any occurrences of the record in a transaction file, including the delete notice transaction, are not deleted since the transaction file is a chronological history of processing and not an application data set. "Deleted" records are retained in the transaction file until eliminated by a time wiper function.

5.2.1 Combined Edits

The CAO framework allows for more than one station to edit its copy of a record within the same system cycle or working day. See Sections 5.7 and 5.8 for further discussion on "system cycle". Figure 4 depicts the case where an existing record is edited by two different locations and is handled in the same transaction processing batch.

In the first case, an operator recognizes that the water volume should be corrected to 5 m^3 from 4 m^3. The station generates an edit record with all data fields blank except the water value field which contains the new desired value. At another station, identified as the source "NGAS" (Natural Gas Accounting System, discussed in Section 9.4), the gas volume value is modified from 0.23 10^3m^3 to 0.32 10^3m^3.

COMBINED EDITS - WELL TEST EXAMPLE

		Volumes		
		Oil	Water	Gas
WELL 1	89.08.01	11	4	0.23

Original Record in Main File

WELL 1	89.08.01		5	

Operator Revision

WELL 1	89.08.01			0.32

NGAS Revision

WELL 1	89.08.01		5	0.32

Edit Transaction Communicated to Network

WELL 1	89.08.01	11	5	0.32

Resulting Record in Station Databases

Figure 4: Combined Edits - Well Test Example

The host then receives both these edit transactions in a single batch processing. There is no conflict between these two edits, they are simply "interpreted" by the host as two separate edits taken against the same data record. Thus the transactions are combined with the non blank fields overwriting the blank fields to create a edit transaction with two fields to be updated. The combined transaction is then distributed by the host to the network.

This concept is extended to any number of combined edits. The host simply accumulates the edited fields by overwriting the blank values with the non blank values for each field to be edited. A conflict arises only if two edit transactions attempt to modify the same field to disagreeing values. Conflicts are discussed in Section 5.3.

Combining of edits can also occur when a work station processes an REU preliminary file against its network file. This processing scenario is similar to the host method just discussed.

Timing delays in the network could result in the attempt to combine data edits originating in different dates. This is a legitimate process and is discussed further in Section 5.4.

5.3 Data Collisions

CAO transaction processing must contend with the same data entering the system at more than one entry point. Duplicate data arriving at the host for batch processing are detected and compared. If all data fields agree a single copy is accepted and no extraordinary action is taken. If at least one data field in the arriving transactions disagree, a "data collision" is said to have occurred.

The detection of a collision during transaction processing is accomplished by screening all incoming data with records already existing in the collection file. The record keys, such as well identifier and date (see Chapter 6), are compared with all the records in the collection file. Should any records match in these keys, the host then compares the contents of the data fields to see if they are in agreement. If any differ, a collision is signalled and the collision processing sequence is initiated for that data field.

Specifically, a data collision is the attempt to automatically process multiple transactions of identical keys and differing non-blank data fields in the same system cycle. This could occur at a host, as described above, or at a work station. Section 5.9 discusses the work station scenario in greater detail.

Section 5.4 discusses the timing aspects of data collisions in the context of timing exceptions. Sections 5.7 and 5.8 provide more detailed discussion on the concept of system cycle.

Data collisions are a problem to be reconciled during the automatic processing of data transactions. For example, a record may be partially processed so that a copy of the transaction exists in both the collection file and main file. If an attempt is made to reprocess the same record under automatic control the full collision detection process is invoked, complete with collision handling algorithm if necessary.

Human judgement is always considered superior to the methods of automatic processing. For example, suppose a user edits a record which has previously been partially processed so that a copy exists in both the collection and main files. The scenario is identical to the previous paragraph except that the editing is being done under human control. The user entered edits are accepted, overwriting existing data, without collision detection and handling processing. A later network transaction to modify the record would, however, invoke full collision detection and handling processing against the existing record.

The CAO host resolves a data collision by choosing a value for the disagreeing fields using rules established for the given data type. For example, the most recent data may be accepted, or a particular source may be predefined as more reliable and its data preferentially selected. The selection rules are called the "selection algorithm".

This could be accomplished, for example, by checking the user identification keys stored in every record (see Section 6.7.3). Figure 5 continues the example discussed in Section 5.2.1 except that both the NGAS source and the operator attempt to edit the water and gas fields in the same system cycle.

The Figure assumes the algorithm has been previously defined so gas volumes from the source "NGAS" are preferentially accepted over other sources, and the oil and water volumes are preferentially accepted from the operator. The figure shows the resulting edit transaction communicated to the network and how the record will appear in all the data sets after full processing.

The types of selection criteria identified for CAO collision handling are outlined in Table 1. Other types of selection algorithms are certainly possible but have not been determined to have any advantage for current and anticipated CAO applications. Section 6.7 discusses various keys which may be examined by the selection algorithm for determining which data to accept. The selection algorithm can usually be assigned to each data field individually of each record type.

DATA COLLISION - WELL TEST EXAMPLE

		Volumes		
		Oil	Water	Gas
WELL 1	89.08.01	11	4	0.23

Original Record in Main File

WELL 1	89.08.01		5	0.27

Operator Revision

WELL 1	89.08.01		6	0.32

NGAS Revision

WELL 1	89.08.01		5	0.32

Edit Transaction Communicated to Network

WELL 1	89.08.01	11	5	0.32

Resulting Record in Station Databases

Figure 5: Data Collision - Well Test Example

Decision Key	Selectable Option
Date and Time	1) Choose most recent. 2) Choose oldest.
User Source	1) Preferentially select data from the transaction with user identifier matching a previously determined character string.
Device Source	1) Preferentially select data from the transaction with the device identifier matching a previously defined SIC.
Processing Sequence	1) Keep first in. 2) Keep last in.
Arithmetic Operation	1) Select greatest value. 2) Select least value. 3) Average colliding values. 4) Sum colliding values.

Table 1: Collision Handling Algorithms

It is possible that the result of a selection algorithm may be indeterminate. For example, neither of the user identifiers may match the preferred source character string. The CAO convention is to have the selection algorithm default to the data from the most recent transaction (by date and time). If both colliding records have identical date and time keys, then the selection algorithm accepts the data from the last transaction received in the processing sequence.

Only one record with a given set of keys is allowed in the collection file and the main file. Thus each incoming transaction, if in collision with a record already in the collection file, is resolved against the existing record with the result replacing the contents of the effected data field in the original record in the collection file and the main file. Should a third transaction be in collision with the prior two, it is resolved against the result of the first collision handling occurrence, with the new result replacing the record in the file. And so on.

It is possible that the automatic reconciliation will result in a desired value being overwritten by an undesired value. However, the automatic selection algorithm is chosen so most collisions will be reconciled with the desired value in most cases. More important, the reconciled transaction will be communicated across the network so the record will be consistent in all databases.

Automatic reconciliation allows the transaction processing to continue without immediate and costly human intervention. In practical economic terms, the automatic resolution is more cost effective and likely to produce fewer errors than a process which relies 100% on manual reconciliation of collisions.

All data collisions are recorded in the transaction file as an exception, including the original colliding records and the record resulting from the host processing.

Data collisions only occur when a work station or host meets conflicting information during a single system cycle. Once a transaction has been successfully processed it may be modified and communicated as a revision on the network. Thus if an operator notices a reconciled collision (e.g. by reviewing the transaction file) he may override the selected data with an edit. This edit will be reflected across the network in a later system cycle provided that no other station sources a disagreeing edit.

A limited version of the collision selection algorithm was built in the pilot. See Section 9.10

5.4 Timing Exceptions

Section 4.4 identified some timing concerns effecting the CAO network which are generally not a concern when dealing with conventional networks. Time delays in the network can lead to data transactions arriving at a station out of sequence, an event called a "timing exception". Specifically, a timing exception occurs when an attempt is made to update an existing record with an edit dated older than the existing record.

When a transaction is received the recipient station always tests to see if the incoming transaction is more recent than the existing record.

If the incoming record is more recent than any existing record in the station, the record is processed normally as described in Sections 5.2 and 5.3. The date and time stamp of the incoming record replaces the date and time of the older record.

Special handling occurs when the incoming transaction is older than the existing record. The particular handling varies with the following circumstances.

1) If the incoming record is a deletion notice:

   a) delete all occurrences of the record in the main file, and set the delete flag to 'Yes' in any occurrences found in the network or collection, and preliminary files.

2) If the incoming transaction is older than the record existing in the main file and there is no matching record in the collection or network file:

   a) the date and time of the transaction is replaced with the current date and time (from the stations date and time-of-day clock).

   b) the edit is processed against the main file.

   c) the edit is copied to the collection or network file for redistribution on the network (the edit is re-sent to the network to ensure all stations are reconciled to the number retained locally and to the new date and time stamp).

d) the time exception is recorded in the transaction file, including the original records from the main file, the incoming record, the result (as stored in the main file), and the redistributed edit transaction stored in the collection or network file.

3) If combining edits in a collection or network file:

a) the edits are combined as outlined in Section 5.2.1.

b) the resulting transaction retains the more recent date and time.

4) If the incoming record is in collision with a transaction in a collection or network file:

a) process the collision as described in Section 5.3.

b) the date and time of the transaction in the main and collection or network file is replaced with the current date and time (from the stations date and time-of-day clock).

5) If an incoming record in a transit file matches a record in the network file, but is not in collision with the network file transaction:

a) process the edit against the main file.

b) the date and time of the transaction in the main and collection or network file is replaced with the current date and time (from the stations date and time-of-day clock).

It is possible the above procedures may result in a desired value being overwritten by an undesired older edit. However, the procedure is expected to result in the desired value most of the time (similar to the collision reconciliation), and the other arguments of Sections 4.4 and 5.3 apply.

A simple algorithm for processing out of synchronization edits would be for the recipient station to reject any data older than that already contained in its main file.

However, since the CAO framework allows data edits on a single field of a record, this algorithm would require that the date and time of last modification be stored for every data field in a transaction. This is considered to have onerous penalties regarding data storage and processing performance. These penalties were deemed to be more onerous than the problem they are intended to correct. (The number of timing exceptions is a small fraction of the number of transactions processed in practical circumstances.)

Every transaction record has one date and time stamp that reflects the most recent modification to the record, even if only a single field was modified (see Section 6.7.2). It is therefore impossible to tell the age of an individual data field. This is insufficient for handling timing exceptions on a per-field basis. For example, suppose an incoming transaction made to the first data field in the record was made at noon on day one. If this record is received at a station where the second data field in the record was modified at 1:00 PM, the station would be unsure of the age of the contents in its first data field and would be unable to correctly determine if it should keep or reject the incoming data edit.

This partial loss of rigor is a theoretical disadvantage which can be compensated for by other techniques (as discussed in this Section, Section 5.9, and 9.13), can be tracked (see Section 5.6) and which does not detract from the economic benefits of the network in practical circumstances.

The automatic handling of timing exceptions allows the system to continue to function without costly delays or manual intervention at inconvenient times. Tracking of the timing exceptions allows manual review at a more convenient, and therefore less costly, time.

5.5 Special Data Import (SDI) and Export (SDE)

The CAO concept provides for the bulk transfer of data to and from specialized computing systems that generally are not within the CAO network. This situation arises most often when highly focussed or commercially available software is purchased for a specific purpose. It is generally true in such circumstances that Amoco has no direct control over the development of these systems and hence they do not conform to CAO standards.

Nonetheless, these systems often generate or analyze information applicable to the general CAO network, and it is desirable to share this information. CAO accommodates these systems as foreign entities to the network. They are called "special" systems in the hierarchy. Information can be imported from or exported to these systems. However, they do not participate in the transaction processing in the general manner discussed in the previous sections of this chapter, nor in the CAO networking discussed in Chapter 4.

Examples of specialized systems which may need to transfer data to CAO networks include:

1) gas volume computing systems,

2) dynamometer systems (well beam pump mechanical performance analysis),

3) gas analysis systems which provide compositional break downs of natural gas streams,

4) dedicated field processors (such as scale-house computers and programmable logic controllers).

To share information with the CAO network, special systems must be able to create and read structured text files containing suitable data keys and independent data for transfer.

The special system does not need to utilize the same keys as the CAO network, provided an unambiguous relation can be established between the special system and the CAO system. A translation routine would then be required in the SDI module to replace the special system keys with the appropriate CAO keys in the transferred data.

The CAO approach is to provide specialized routines to transfer the data in the format of the special systems. Usually the intervening Special Data Import (SDI) or Special Data Export (SDE) files contain data in structured text format. Records may be in any order, but the record fields are interpreted left to right in positionally dependent order. See Section 9.4 for an example of an intervening file.

Transfers into the CAO system are typically done in batch format at the host. Preparation of export files from the

CAO network to the specialized system is also typically done at the CAO host. Any routine file transfer method is acceptable, with diskette being the most typical.

Since the special systems do not conform to the CAO network, the CAO host which receives the information is responsible for screening the data to provide for CAO relevant inputs to the network. Specific problems which can potentially occur when dealing with importing data from special systems include:

1) Unique and unambiguous identification of individual data entities.

2) Handling conflicts between the received "special" data and data existing in the CAO framework.

3) Removing redundant information from the received transfer before processing.

Unique data identification is required to meet the demands of the CAO data control procedures, discussed in Chapter 6. Keys must be present in the SDI transfer format for the receiving host to process. When the host receives a data record for which it does not recognize the key identifier, it rejects the record and no further processing on this record takes place. If desired, the rejected record may be logged to the transaction file. The decision to log or not is made at design time of the custom SDI processor for the particular special system interface.

During an SDI transfer the host stamps each record with the current date and time (from the host's date and time-of-day clock), signifying the date and time the transaction entered the control of the CAO network.

Figure 6 portrays the context of processing an SDI transfer.

The SDE process is much simpler. When the host is requested to provide an SDE data file, it uses the search criteria (defined in the unique requirements of each SDE processing module) to extract the desired data from the main files at the host. The data is then formatted as appropriate for the intended target system, along with any translation of keys which may be required. Figure 7 shows the basic SDE process.

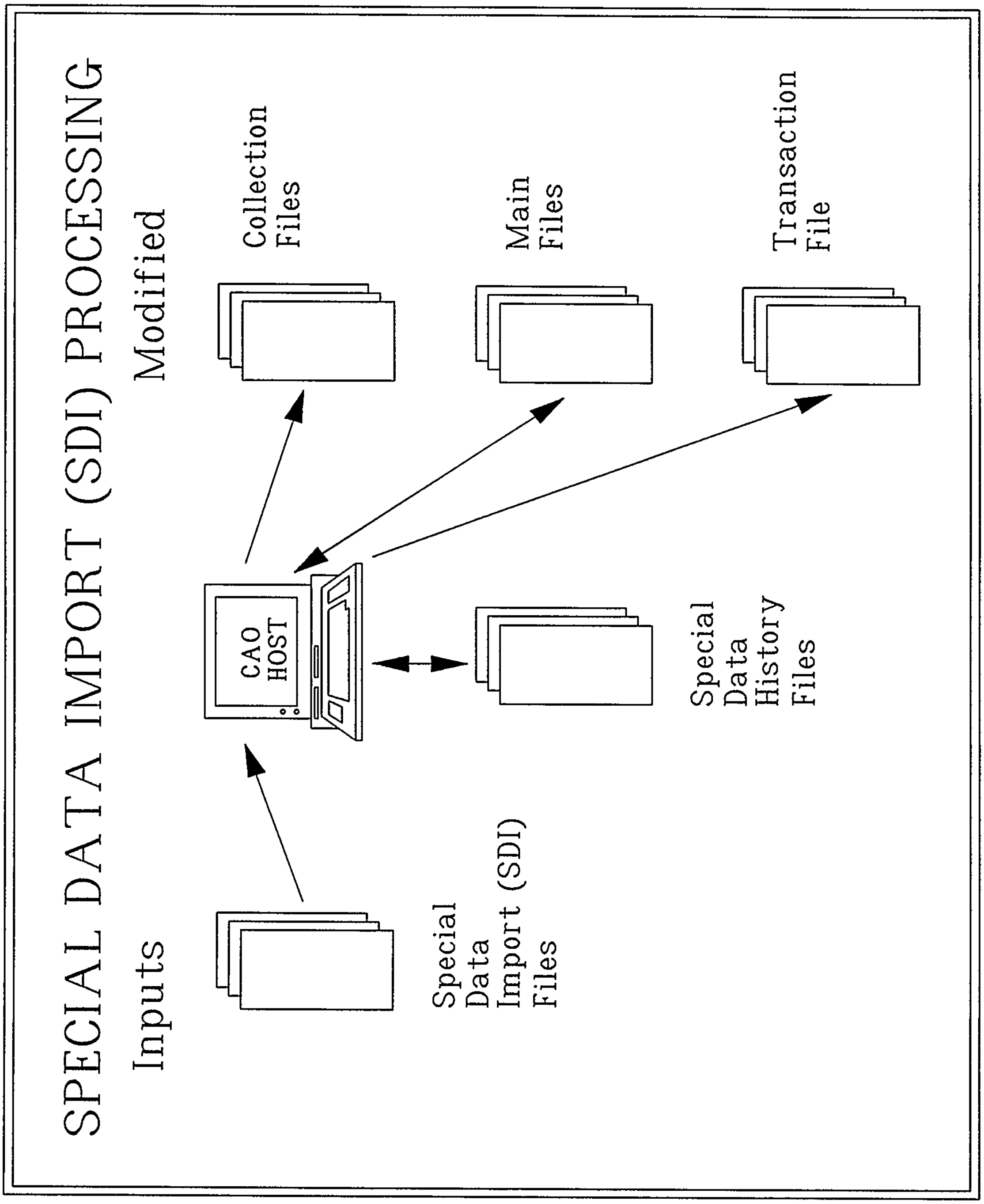

Figure 6: Special Data Import Processing

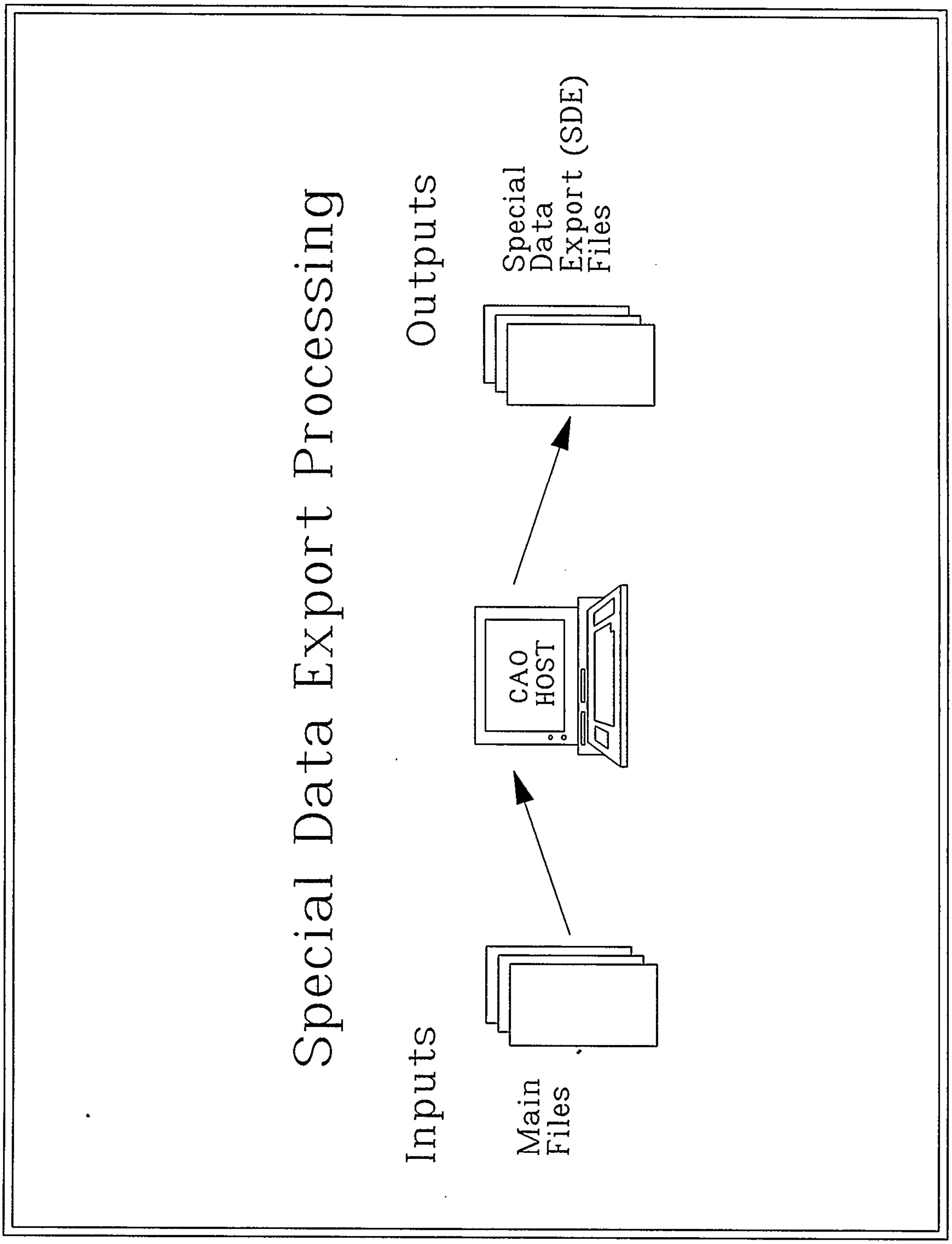

Figure 7: Special Data Export Processing

The SDE process is not transaction processing in the strictest sense. Individual transactions are not worked against any data set during the SDE process. SDE uses selection criteria to extract information in a predefined format. SDE is considered part of the CAO EDT technology since it works with the CAO data set and contributes to the flow of information. See Section 9.13 for an example of the SDE process.

5.5.1 Filtering SDI Transfers

Most SDI systems provide data output based on simple selection criteria. In some circumstances, particularly if data is transferred regularly, the data selection criteria can result in redundant data being passed in a series of SDI transfers. Redundant data handling by the CAO system is undesirable for several reasons. Redundant data:

1) could overwrite modified versions of the data,

2) over populates the transaction file and generated reports with undesirable information that increases storage requirements, slows response time, and reduces report effectiveness,

3) unnecessarily loads the network with redundant information, and

4) causes a performance penalty during transaction and EDT processing.

To protect against these undesirable factors the CAO SDI transfer provides a filtering process which removes from the incoming data set all records which are identical to information previously received. Thus, only the SDI transfer portion of the system processing is burdened with this redundant information.

The filtering is accomplished through the process of maintaining files of the information previously received from the special system. These files of previously received data items are called "special data historical files", or typically just "filter files". As new data is received, each record is compared with records stored in the filter file. If there is an exact match, the record is rejected and processing continues to the next data item. If there is no match, the record is passed to the collection file and a copy is stored in the filter file.

If there is a match on keys but some of the data has been altered, it is assumed a revision has been received and an edit record is passed to the collection file.

The detection of redundant data could be processed directly against the main file, obviating the need for the SDI historical file. The case of data edits, however, causes the special requirement of maintaining an historical record of data received from the special system. Later SDI transfers must recognize that the redundant receipts of the original data is undesirable and should not overwrite the edits made by a user on the CAO network. This can only be accomplished if a copy of the original record, unaltered by later edits on the CAO network, is maintained for comparison with newly received SDI transfers.

Since SDI transfer is one way, modified data is not automatically communicated to the special system for updating. Reconciliation of databases is a unique CAO function. Special systems, by definition, are not built to CAO standards.

Most SDI transfers will involve data that has a definite time window. Thus the special SDI processing, custom designed for each SDI interface, can use a "data wiper" to automatically delete records in the filter file which are older than a given time period.

Figure 8 depicts the three basic cases of how SDI processing filters the incoming data. In the first instance, a record in the SDI file has keys which do not match the keys in any of the records which exist in the filter file. Thus an exact copy of the entire record is passed to the collection file, and is appended to the main file (not shown), the transaction file (not shown), and the filter file. When the processing is complete a copy of the record is appended to the filter file to screen any later receipts of the same record.

In the second instance, a record is received which has keys and data fields which exactly match a record in the filter file. No further processing of the record then occurs and no copies are made or revised in any other set of files.

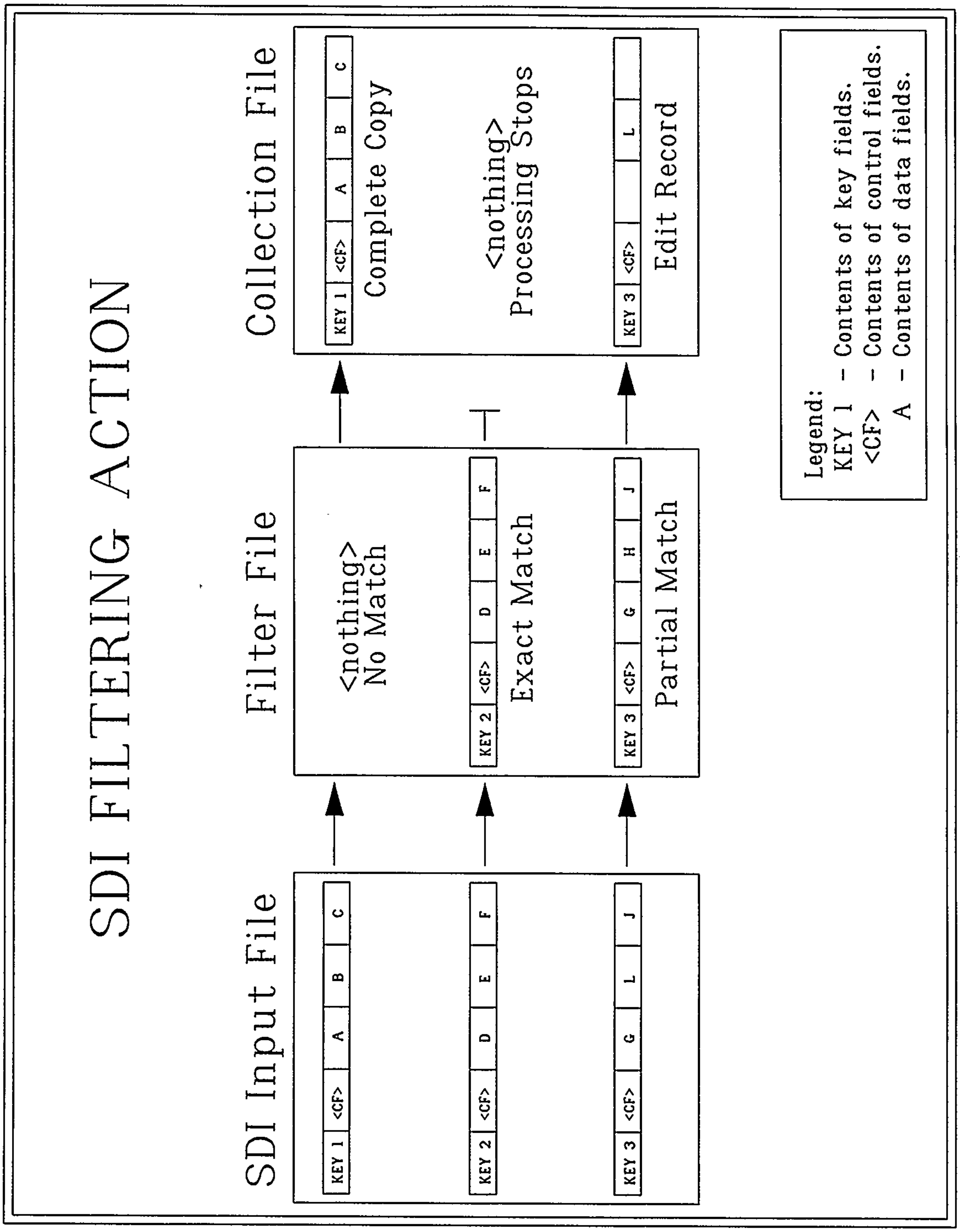

Figure 8: SDI Filtering Action

In the third instance, a record is received for which the keys exactly match a record in the filter file, but one of the data fields differs from the data fields in the filter file. The SDI process then interprets the transfer as an edit and generates an edit transaction in the collection file in the format discussed in Section 5.2. The data field "L" replaces the data field "H" in the filter file (not shown) and the main file (not shown), and a copy of the edit record is made in the transaction file (not shown).

5.6 Transaction Reports

Each transaction processed by the host or received by a work station is recorded in a special file called the "transaction file". The file stores a chronological history of all transactions which have been handled by the local processor so that human review and intervention can be performed for any questionable handling that may be observed. The transaction file is the only file allowed to have more than one record with the identical keys, since all transactions applied against a data set are stored in their original and resulting form.

This is particularly important for regular review of data collision and timing exception handling. The handling procedures described in Sections 5.3 and 5.4 cannot replace informed human judgement and therefore require operator review. In most cases, the automatically selected value does constitute the best one. However, the operator is obligated to review all collision and timing resolutions and correct any that appear suspect.

All data items communicated by EDT, including SDI (but excluding SDE), are contained in the transaction file which can grow to be quite large. However, the relevancy of the contents of the transaction file diminishes over time, thereby allowing a maximum retention period to be applied against the file. For example, in the CAO pilot, all data items older than 60 days are erased from the transaction file when it is updated.

The transaction file stores a copy of each transaction record imported from a special system which was determined from the SDI filtering process to be new or modified data. Those transactions which are redundant with prior SDI transfers are not considered to be true transactions and are filtered, with the result the transaction file does not retain a copy of the redundant transaction. Only new or modified transactions received in an SDI process are added to the transaction file in their new form.

The transaction file feature was built and tested in the CAO pilot. The transaction file structure, an example of tracking a well test collision, and a sample report are discussed in Section 9.9.

5.7 Accumulation and Reconciliation

The transaction processing described in the preceding sections actually occur in two basic steps. The first is the accumulation and reconciliation component described in this section. The second is the clearing and distribution component described in Section 5.8.

The combination of these two steps comprises the "system cycle". The time between the completion of one system cycle and the beginning of another is typically referred to as the "cycle time". Most locations perform their activities with a daily system cycle such that the cycle time is usually 24 hours.

The accumulation and reconciliation process is a series of mini processing batches that occur each time information is acquired at the host. A CAO host has several possible information sources. It can receive data by:

1) keyboard entry at the host,
2) receipt of data from REU preliminary file,
3) receipt of a network file from a work station,
4) receipt of special or SCADA data via an SDI,
5) receipt of a transit file from another host, and
6) receipt of data from the PDR.

As each input type is processed, the transactions are applied against the collection file complete with collision detection and reconciliation and timing exception handling. The transactions are also applied against the main files which reflect the last revision of each transaction (as modified by the collision handling algorithm, if necessary). The host continues to process all incoming transactions against the collection file.

The collection file is only cleared during the clearing and distribution process described in Section 5.8. Thus two or more transactions with identical keys and differing data fields processed into the same collection file are detected as being in collision. A later transaction received after the collection file is cleared, which may have identical keys and differing data fields from transactions processed before the clearing of the collection file, is deemed to be a *new* transaction and is not detected as a collision.

Transactions are processed at the host into the collection file at the time they are received. Thus transactions with a date and time stamp that fall into the time period of a previous system cycle cannot be reprocessed into that system cycle after the collection file has been cleared. The date and time of a transaction creation or modification is therefore irrelevant to collision detection. (It may, however, be a factor in the selection algorithm used to reconcile the data and the way any timing exception may be handled, as described in Section 5.4.) Collisions are detected only at the time they are processed against the collection file in existence when the transaction is received at the host.

The accumulation and reconciliation process also handles any communication processes of the host-controlled nature. Each time a host-controlled communication session occurs the host initiates and controls the communications, reads the network or preliminary file it is receiving, and processes it against the collection file.

For remote-controlled communications, the host has received a file under control of another station which is deposited as a network or preliminary file in the host storage (disk). At some point in the processing cycle these files are read from the disk storage and processed against the collection file. The limitations of the single tasking MS-DOS operating system make it non cost effective to have the remote stations in remote-controlled communications automatically process their file against the host collection file. They must simply deposit the file and let the host process it when convenient.

Figure 9 shows a context diagram of the input sources applied against the collection and main files, with a notification being posted to the transaction file.

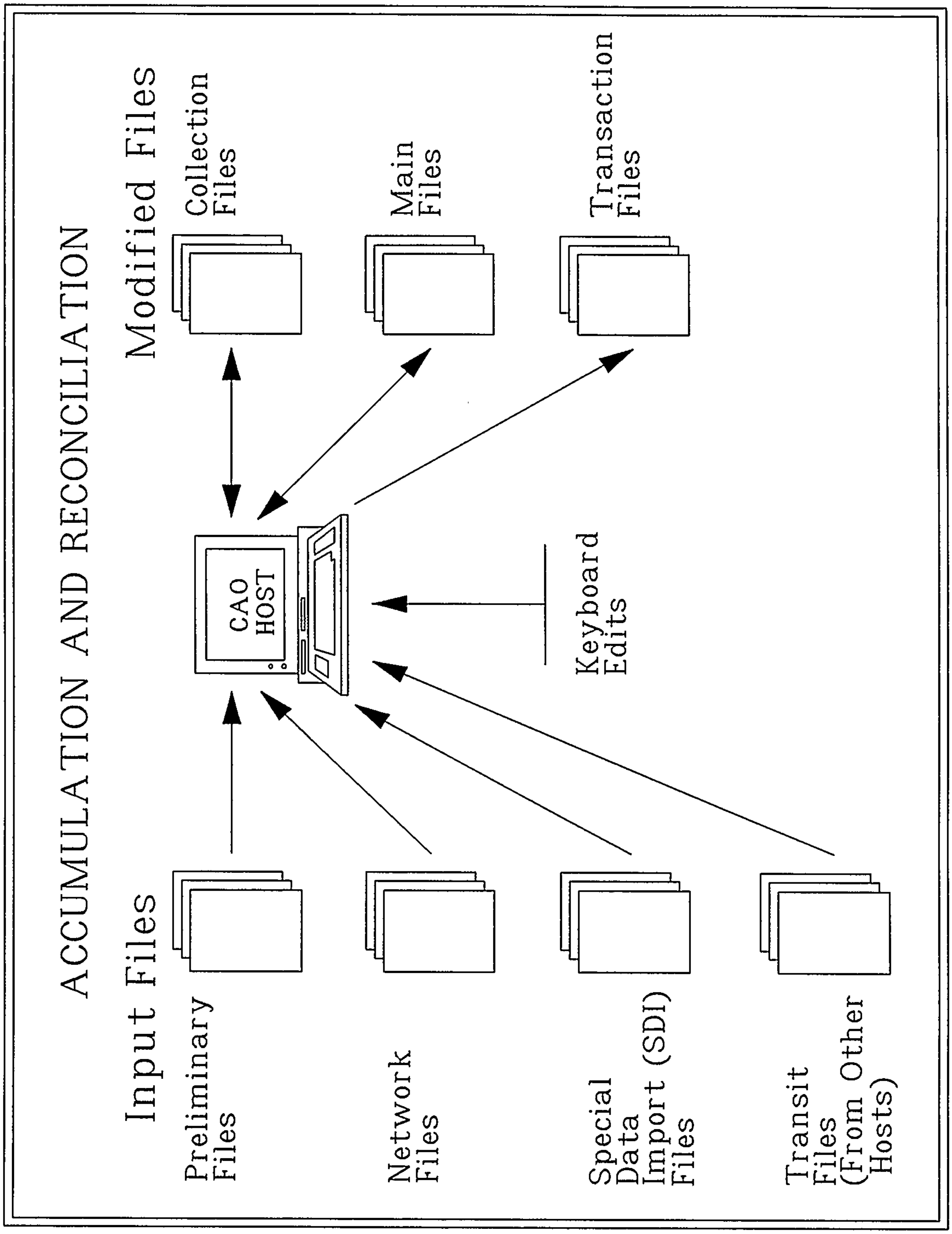

Figure 9: Accumulation and Reconciliation

5.8 Clearing and Distribution

At some point in time, the host processes the collection file to distribute the transactions it contains to the network. In this process, the host takes each transaction from the collection file and uses the keys to determine which stations require a copy of the transaction. Section 6.8 describes how the keys determine transaction routing across the network.

As the host processes the transaction from the collection file, it first looks to see if a transit file has already been created for each station which requires a copy. If not, the host creates the transit file with the appropriate addressing for that station. The host then takes a copy of the collection file record and appends a copy of the record to the bottom of the transit file for the recipient station. Some transactions may be added to the transit file of more than one recipient station. The host clears each transaction from the collection file as it is processed, until the collection file is empty.

If all the stations on the CAO network were to receive its transit file and perform no other processing, the databases across the network would now be synchronized with identical copies of the overlapping data elements. In practice, each of the stations continues to acquire and modify data asynchronously so it is rare that the full network is ever perfectly synchronized at any instant. The clearing and distribution portion of the host processing, which acts to synchronize the databases of all the stations beneath it in the hierarchy, is generally regarded as the end of one system cycle and the beginning of another.

At the end of the clearing and distribution process, the host initiates all host-controlled communications and transfers the transit file to the recipient station. In the event the REUs are the communication medium to remote work stations, the host places the transit file in the designated REU for each remote station.

For those stations which communicate in the remote-controlled mode the host leaves the transit file in its disk storage. Later, when the communicating station initiates the communication session, it searches for and retrieves any transit files addressed to it.

Regardless of the communication mode used, a check is always made to see if there is an existing transit file from a prior system cycle that has not yet been transferred to the addressed station. If so, the host appends the new transit file to the end of the transit file left over from the previous system cycle.

Figure 10 shows a context diagram of the clearing and distribution component of the transaction processing.

5.9 Receiving Transit Files

When a station receives a transit file, it processes the transit file contents against its main files. New records are simply added to the main file. Edit records are processed so the contents of the non blank data fields from the communicated edit record overwrite the contents of the corresponding fields of the matching record in the main file.

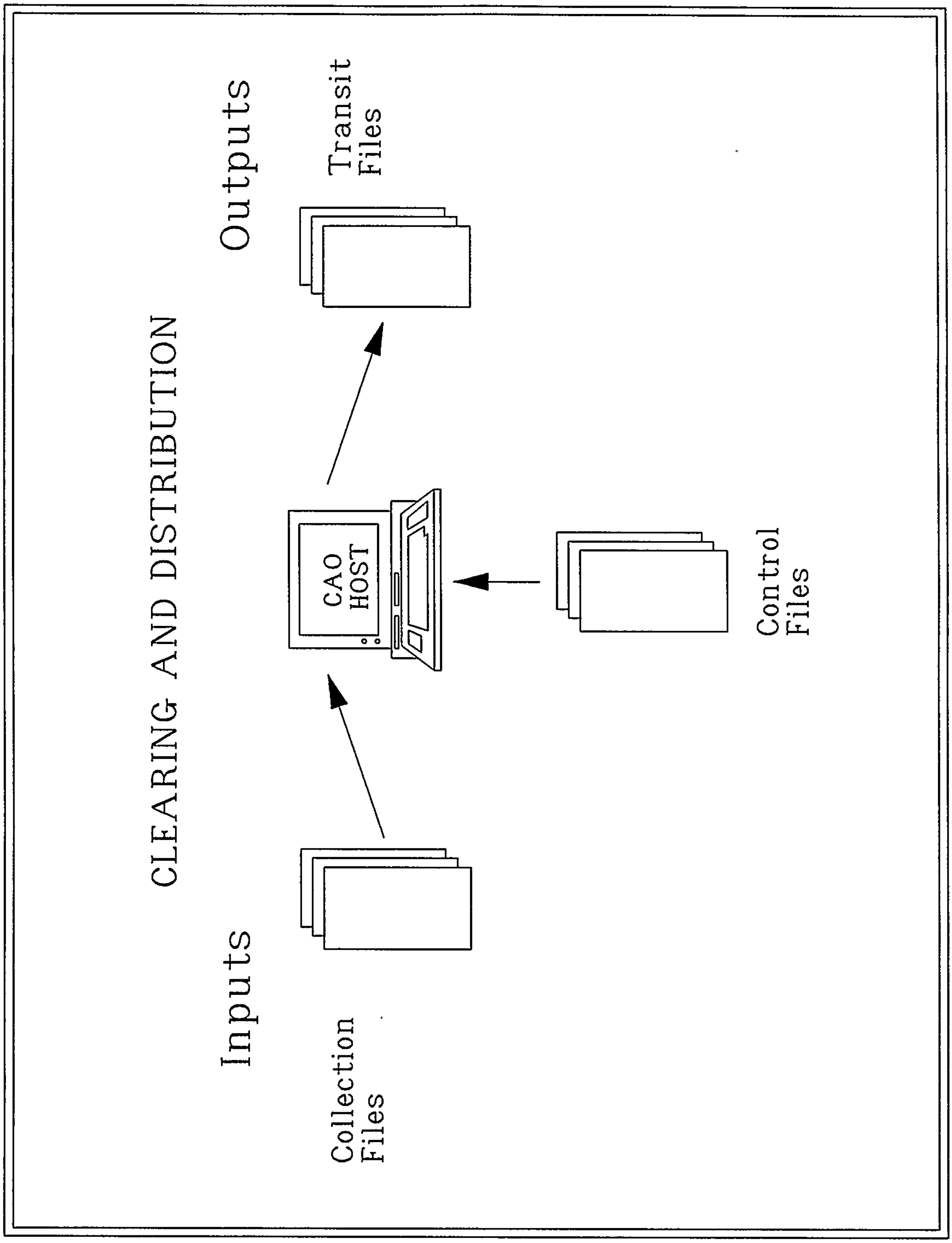

Figure 10: Clearing and Distribution

Special processing is required for handling transactions received in a transit file which match records in the network file. This would happen if an operator has modified a record since the last communication and batch processing at the host. This is handled by treating any matches between the incoming transit file and the resident network file as a potential collision. If data fields differ between the record pair, a collision is signalled and a collision handling process occurs, similar to that described in Section 5.3.

Every field in collision is reconciled using the appropriate selection algorithm, with the result overwriting the previous value in both the main file and the network file. Data fields which are not in collision (either one is blank or the values agree exactly) are not processed for a selection criteria. The non-blank, non-colliding fields from the transit file overwrite the contents in the main file only. The network file contents are not modified by non-colliding transit file data fields.

It is quite possible that a single transit file may contain more than one transaction with a given transaction key. This could happen if it had been several days since a work station had a communication session with a host. The host could have processed a transaction against a particular record, then cleared the transaction by appending it to a transit file. If the transit file is not forwarded to the addressed station, a later edit on the same data item would be processed by the host and *appended* to the bottom of the transit file, which would now contain two transactions for the same data item.

Recipient stations handle this by processing the transit file one record at a time, starting at the top of the file. Each record in the transit file is processed against the data set *in order of occurrence* in the transit file. Later transactions are processed against the results of preceding transactions on the same data item.

Table 2 contains the procedure the work station follows when processing a transit file. In the absence of a collision, timing exception, or failure to recognize a key, the work station will simply apply the transaction against its database. This act brings the local database into synchronization with the same transaction elsewhere in the system (assuming further updates have not been made) and stops the communication chain.

```
            HANDLING TRANSIT FILES AT A WORK STATION
for each record in transit file DO
 * if the incoming record has delete flag set
     find key in main file
     if found
       delete this record
       copy deleted record to transaction file
     endif
     find key in network file
     if found
       set the delete flag to "Y"
     endif
     copy transit record to transaction file
 * else
     find key in the network file
     if found and delete flag in network not set
       if collision
         copy network record to transaction file
         copy transit record to transaction file
         combine records using collision algorithm
         copy result to transaction file
       else
         combine records
         copy transit record to transaction file
       endif
     endif
     if work station recognizes key
       find key in main file
       if found
         update main with most recent date & time
       else
         append to main
       endif
     endif
 * endif
ENDDO
```

Table 2: Procedure For Receiving Transit Files

If the work station does not recognize the transaction key as belonging to the work station, it simply ignores the transaction and updates neither its main nor network file.

An exception to the above processing sequence occurs when a record signals a deletion is required. These records are shown by the setting of a "delete flag" field in the record structure. When a delete record is received, both the main and network files are searched for the matching record. The matching record is deleted in the main file. If a match is found in the network file, the delete flag is set but the record is not deleted from the network file.

6 DATA CONTROL

Data Control in CAO refers to the way transactions are processed and routed through the network. The data control is determined by data "keys" which are contained in each transaction communicated in the CAO framework. A key is a field or group of fields within a data record or transaction that contain information about the transaction, including its identification. Keys are used in several ways within CAO:

1) **To uniquely identify a transaction.** The keys that uniquely identify a transaction generally fall into two distinct categories (although other categories are possible):

   a) a transaction number which is assigned in such a way at the time the transaction is created to guarantee a unique identifier, and

   b) a combination of a predefined entity code (which uniquely defines a field facility or other entity) and a "book date" which associates the transaction with a given calendar date. The combination of a uniquely identified entity and book date uniquely identify the transaction. It is assumed that no more than one transaction of a given type will apply to an entity on any given day.

2) **To determine how transactions should be routed across the network.** The CAO host has relational tables which associate each of these routing keys with a list of SICs.

3) **To provide ancillary control and identification functions** such as determining the transaction source for collision handling purposes. Sometimes identifiers are assigned which provide more meaningful information to users than the cryptic codes assigned for data processing.

4) **To provide an audit trail.** This is not part of automatic processing but represents a requirement imposed by business needs. Formal audits generally require an indication of who originated a record and when he did it. Informal needs also apply. Each operating run has several operators which could be manning the shift on any given day. Since all

operators contribute to the "bookkeeping" of the run it is helpful to the operators to know who collected or modified a given piece of data.

5) **General processing.** Keys are used in other ways within CAO and its applications such as sorting and indexing records, filtering of records for selective processing of information, and establishing relationships between transactions through relating the keys, etc.

Sections 6.2 through 6.7 define and describe the keys used within CAO. It is anticipated that future CAO applications will require additional keys than those identified here.

Table 3 summarizes how the various keys are used to achieve the desired results itemized above.

Key	How Key Is Used				
	Identify Transaction	Network Routing	Other Control	Audit	General
FLAC	YES	YES	NO	YES	YES
Meter Run Number	YES	YES	NO	YES	YES
Transaction Number	YES	NO Note 1	NO	YES	YES
Local Identifier	NO	NO	YES	NO	YES
Book Date	YES	NO	NO	YES	YES
Device Identifier	NO Note 2	NO	YES	YES	NO
Input Date and Time	NO	NO	YES	YES	NO
User Identifier	NO	NO Note 3	YES	YES	YES

Notes:

1) A future work order system might relate a work order transaction number with the originator (by user identifier), and then to an SIC.

2) Device identifiers can be used as a constituent component in building transaction numbers. See Section 6.4.1.

3) A future electronic mail "messaging" system might relate the addressee (by user identifier) to an SIC.

Table 3: Using CAO Keys

6.1 Transaction Keys and CAO EDT

For the CAO EDT process to function effectively it is necessary for each transaction to have one and only one unique key (perhaps consisting of several different key fields). The necessity for uniqueness is clear; ambiguous keys prevent proper identification of the individual transaction as communicated among the distributed databases on the network.

The necessity for each transaction having only one transaction key, regardless of how many occurrences of the transaction there are across the network, is also clear; if a transaction were allowed to have more than one unique key the transaction processing would be incapable of recognizing the transactions as being the same basic business event; the processing would consider the transaction to be two transactions.

Another important requirement for CAO transactions is that the CAO system must be able to relate each transaction to a list of stations which require a copy of the data. Thus each transaction must have at least one key field which is used to identify the stations to which the transaction will be routed.

In summary, the following rules must be enforced for every transaction which will be processed for EDT:

1) Each transaction must have a unique key.

2) All copies of the same transaction must have the same key.

3) Each transaction must have at least one key used to relate the transaction to a list of stations which require a copy of the transaction.

6.1.1 Temporary Records

There are occasions in daily operations when an operator wishes to enter information (such as estimated values) into his work station for local processing without having to wait for the actual data arriving from another source. A similar situation occurs when two operators have access to the same field information which they wish to enter into their work station, but generation of an identical key for the dual captured information item cannot be assured.

This situation can be accommodated by allowing the operator to enter his estimate or redundant entry as a "temporary" record which is only used in the local database; temporary records are **never** communicated to the network. A record matching process which attempts to relate the temporary record with the network transaction (when it arrives at the work station) is then required. The operator would be required to accept each match. As each match is accepted, the temporary record is replaced with the network transaction. No collision checking or time exception handling is required for matching of network transactions and temporary records.

The concept of temporary records was not tested in the CAO pilot.

6.2 Functional Location Accountability Code

Functional Location Accountability Codes (FLAC) are an accounting code unique to Amoco and are used to identify certain chargeable locations or information entities for which data is to be accumulated. These locations or entities include wells, leases, units, non-interest properties, royalties, facilities, plants, and administrative locations. FLAC numbers uniquely identify each location or entity within the Amoco Production Company, which includes Amoco Canada. For example, a single producing oil well has a FLAC assigned to it. FLAC numbers are assigned to simplify computer processing of business information.

The basic FLAC is a six digit code which is paired with a 2 digit "ownership code" or "registered horizon code" to uniquely identify the location or entity. A check digit, calculated on the full eight digits (basic and ownership codes), is generally added bringing the total number of digits in a FLAC number to nine.

6.3 Meter Run Number

All measurement devices tracked and reported to the Amoco main frame production accounting system are assigned a meter run number that uniquely identifies the meter within Amoco Canada. Most field meters in this category are gas volume meters, but all types of meters can have meter run numbers.

The Amoco - Dome merger of September 1988 resulted in a much larger number of meters in the company. The meter run number, formerly a 4 digit numeric code, is now a 7 digit numeric code. To retain compatibility with FDE and to retain compatibility between the Amoco and Dome main frame accounting systems, the 4 digit meter number used by FDE is right justified so the 4 digit numbers still uniquely identify the specific meters.

6.4 Transaction Number

Transaction numbers are a special class of keys which assigns a unique code to each transaction of a given type. These codes may be automatically assigned at the time a transaction is initiated or may be keyboard entered along with the data fields. Keyboard entry of the transaction number would be necessary where data is being copied from another system (such as a paper method) which has a transaction number already assigned.

Transaction numbers are used for record identification but are not generally used for routing on the CAO network. Transaction numbers are generated "on the fly" and thus relationships between transaction numbers and SICs cannot be predefined.

6.4.1 Transaction Number Generation

The Amoco Canada accounting and auditing standards require a unique transaction number for every event of specific field accounting procedures. An excellent example of this is the transaction number for every recorded fluid transfer transaction (see Section 9.8). Before the CAO techniques this criteria was satisfied through paper forms with pre-printed sequence numbers.

It is a goal of CAO to eliminate paper transacting wherever possible in our field operations. It is therefore impractical to rely on a pre-printed sequence of forms for selecting unique transaction numbers for processing. CAO techniques require a new means of generating unique transaction numbers.

CAO generates transaction numbers through piecing together of constituent components. The purpose of using constituent components is to provide CAO devices with an algorithm allowing independent assignment of unique transaction numbers. The constituent form is as follows:

[SIC]JJJNN

where:

"[SIC]" is the four character station identity code of the processor on which the transaction is originated. The SIC is unique for every CAO station used in Amoco Canada.

"JJJ" is the Julian date (from start of year) on which the transaction was *created*. E.g. Jan 01 is 001, Dec 31 is 365. The transaction date, an accounting field within the transaction record, is the date on which the actual field event occurred. The transaction date is often called the "book date".

The Julian date ensures unique transaction number assignment within a given calendar year. It is likely, however, that identical transaction numbers will be generated in different years. This is not a limitation in practice, since the book date uniquely determines which year the transaction occurred.

"NN" is a two digit sequence number for the number of transactions created by a given processor on a given date. Each day the sequence number will reset to begin at 01.

Presently the sequence number is base ten, allowing up to 99 C-19s to be generated by a single station. The sequence number will be changed to alpha characters, allowing 26^2 C-19s to be created by a single station. This will be necessary to accommodate certain sites, such as cleaning plant weigh scales, where daily truck movements regularly exceed one hundred in any single day.

6.5 Local Identifiers

It is always desirable to present information to users in a form they are most familiar with. FLAC numbers and meter run numbers are very convenient for computational purposes but are difficult to users for routine reporting. Thus a table of local identifiers is usually maintained for each application. These identifiers have local significance. For example, a table of Battery identifiers and Well identifiers is maintained for user presentation. Thus a report could show a well test for "Battery 1" - "Well 3", instead of a nine digit FLAC.

The same is true of meter run numbers. FDE (as modified for the pilot) allowed the assignment of a sixteen character meter location name to each meter run number. Thus a gas volume screen would show a gas volume for "Satellite sales" instead of a seven digit numeric code.

Local identifiers are not used for EDT and networking.

6.6 Book Date

The book date is the date a given data record is in effect. For example, a 24 hour gas volume measured throughout a production day is assigned the book date on which the gas volume measurement began. The book date could also be the day on which a well test started, the day a trucked fluid transfer occurred, or other common oil field event.

The book date may be used within CAO as a constituent field in the control key which uniquely identifies a transaction. It is also used by the FDE for the daily bookkeeping of the facility.

6.7 Transaction "Stamping"

Every transaction communicated by CAO is stamped with three types of keys: user, device, and input date and time. The following subsections describe these identifiers in greater detail. These keys are important for several reasons:

1) The collision handling selection algorithms may use these keys to determine the result of a data collision (as discussed in Section 5.3).

2) They identify who to contact for answering queries on a particular transaction.

3) They assist data integrity by validating the timeliness of the data capture.

6.7.1 Device Identification

The device identifier stored in each transaction is the SIC of the device that originated the transaction. Thus the device acquiring virgin data would place its SIC in the device identifier key of the transaction. If a device modifies an existing record, it overwrites the contents of the device identifier key with its SIC.

In the event a collision is detected and a selection algorithm is invoked, the collision handling algorithm overwrites the contents of the device identifier key in the resulting transaction with the contents of the device identifier key from which it selects the data result.

6.7.2 Input Date and Time

Every transaction within CAO is "stamped" with the date and time the transaction originated (called the "entry date and time"). Thus virgin data is stamped with the date and time it is captured. An edit transaction is stamped with the date and time the edit was created. Note that the date and time of origin may be very different from the book date for which the transaction applies.

6.7.3 User Identification

Every CAO application requires entry of a user identity code before accessing the application. Usually a person enters three initials to identify themselves to the program. However, the applications are now being modified for six characters to be consistent with the Amoco standard for corporate wide user identification. Amoco Corporation assigns a unique six-character code for each individual who accesses corporate wide computing facilities. Field staff have not had corporate user identifiers assigned but it is expected they will as more become regular users of corporate computing systems.

In most cases, two user identifier fields are stored for each record. The first field contains the identity of the user who first created the virgin record and is never altered. The second field contains the identifier of the user who last edited the record. This field is updated every time the record is edited.

Special systems and SCADA systems may not supply a "user identifier" in the sense discussed in the preceding paragraph. Thus the SDI process, defined at the time a link is developed between the special system and CAO, would pre-fill the user identifier for each transaction with an established code.

6.7.4 Modification Date

In addition to the input date and time, each record has a modification date (modification time is optional) which reflects the last day on which any of the data fields in the record was modified. It is therefore impossible to determine absolutely the last day (and time) any particular field in a record was modified (unless the record contains only one data field).

6.8 CAO Network Routing Control

The relationship between certain types of keys and the SIC of various stations is of special interest to this thesis. These relationships directly determine routing of information across the CAO network. Each "clearing and distribution" event routes each processed transaction by placing a copy in the transit file of each station for which a relationship between the contents of its key and a list of SICs has previously been established.

The existing CAO applications are very "physical" in nature, meaning the information being processed relates to field equipment that can be touched. The established routing relationships require the relating of entity type identifiers (FLACs and meter run numbers) with the SIC of CAO stations

As noted in Table 3, future CAO applications may relate keys other than facility entity keys with network addresses.

Files containing information that controls how the CAO network or transaction processing operates are called "control files". For example, a control file could contain the relations between keys and the network addresses (SICs). Configured control of collision handling selection algorithm by data file type and field would also be contained in control files.

The number of control files required depends on the different types of information processed and the degree of network control required. For example, the Well Test routing control would be by well FLAC which determines network addressing. Other types of data could also access the FLAC file to determine network routing. Hence there is no fixed relationship between the number of data file types and the number of control files.

Figure 11 portrays how network routing is determined from the facility entity key. During the accumulation and reconciliation phase of transaction processing, the collection files accumulate the transactions which are generally keyed to a facility entity.

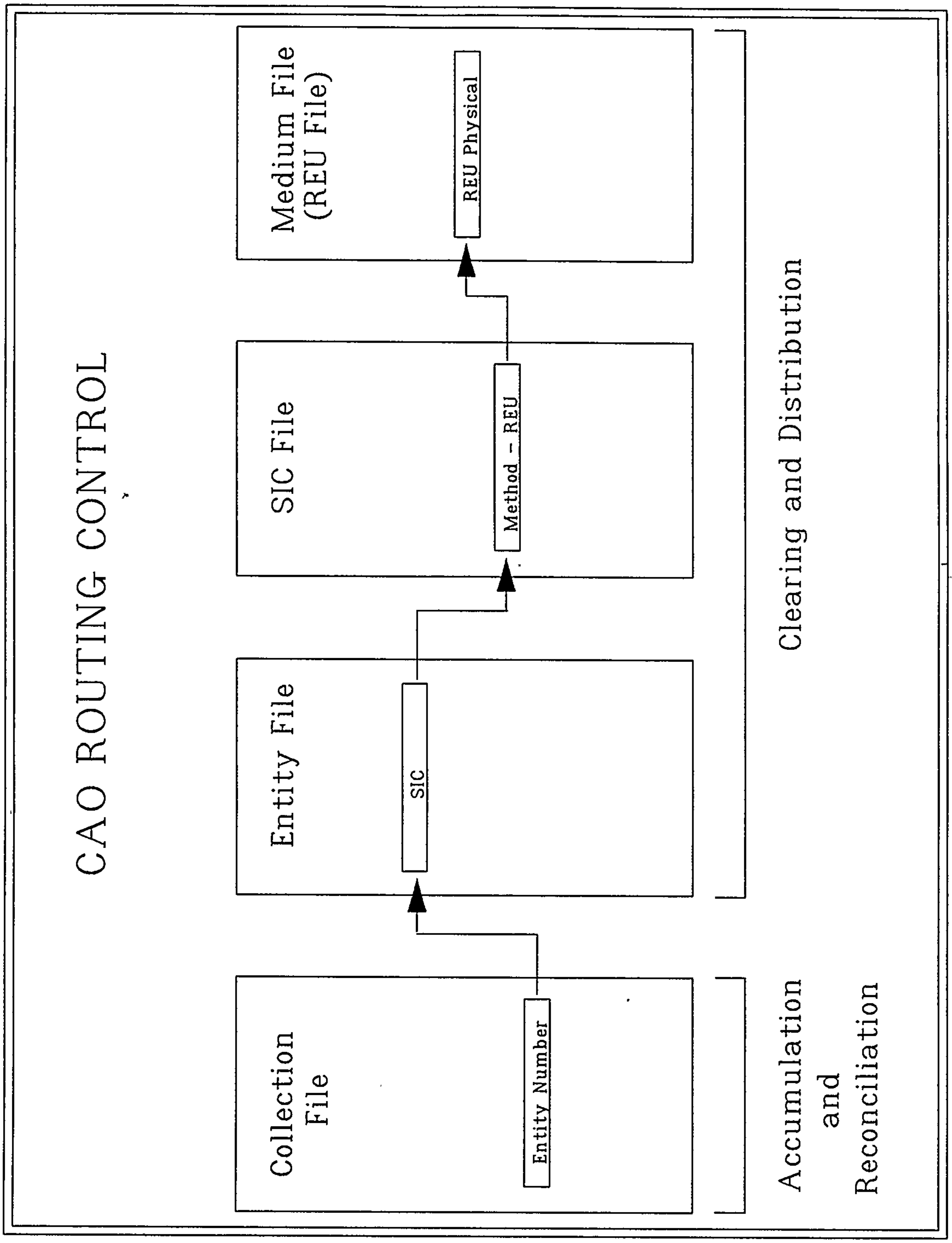

Figure 11: CAO Routing Control

During the clearing and distribution phase, the network routing is determined by using the entity number (a FLAC or meter run number) as a key into the entity file, which contains a list of all the entity numbers. Attached to each entity is a list of SICs which require copies of all transactions effecting the given entity. A copy of the transaction is then placed in a transit file addressed with the given SIC.

In the final step of the clearing and distribution phase, the SIC number is used as a key into an SIC file, from which the communication method to the addressed station is determined. This includes whether the communication is host-controlled or remote-controlled (see Chapter 4) and what the medium is (e.g. modem, designated REU, etc.).

Depending on the medium, additional files may be searched to obtain information about the medium. For example, the SIC file may show the method is host-controlled, that the medium is by designated REU, and also give the SIC of the designated REU. The processor then goes to the REU file to determine the communication method to the indicated REU, such as its physical address. At this point the transit file is written to the REU and the network addressing step has ended.

6.9 Configuration Files

Configuration files are those files which contain information used by an application for its processing. For example, the FDBAT file is a configuration file which provides information describing what types of gas dispositions each battery has and how to treat them in the battery gas balancing equations. See Section 9.6.1.

The data contained in configuration files is generally considered to be static and so is not communicated on the CAO network. There is no theoretical limitation why static data could not be communicated if a situation arose where it was deemed advantageous to do so.

The CAO networking and EDT functions usually do not use the configuration files.

7 DATA ACQUISITION

Data is acquired, or "captured" within CAO in one of three ways:

1) Externally through SDI (from Special or SCADA systems),

2) Keyboard capture at a work station, host, or main frame, and

3) REU capture.

There is potential for a direct link between field devices and CAO stations to provide for an additional source of data capture. However, this will not be addressed until a need arises.

The process of acquiring data through SDI is described in Section 5.5 and will not be further discussed here. Although it is possible to build an SDI or SDE process into a CAO work station, it is unlikely this will be required since most special systems reside at an operations center where a CAO host is present. Furthermore, the work stations generally are chosen of modest processing capabilities, so users will prefer to use the CAO host when given a choice. In practice, therefore, the work stations generally acquire data through keyboard entry and REU capture.

The process of keyboard and REU data capture generally occurs in the field, so this section will discuss the procedure as it occurs in the work station. The procedure is similar at the host except that the network file processing in the work station is replaced with collection file processing at the host. Main frame keyboard capture is not fully defined but would function through the PDR similar to the host.

Figure 12 portrays the way data is acquired and processed at a work station. All acquisition results in processing against the main and network data bases for each data type. For example, there are main and network databases for well tests, main and network databases for group gas volumes, and other data types selected for CAO network data transfer. The keyboard and REU data capture processing are discussed in further detail in Sections 7.1 and 7.2, respectively.

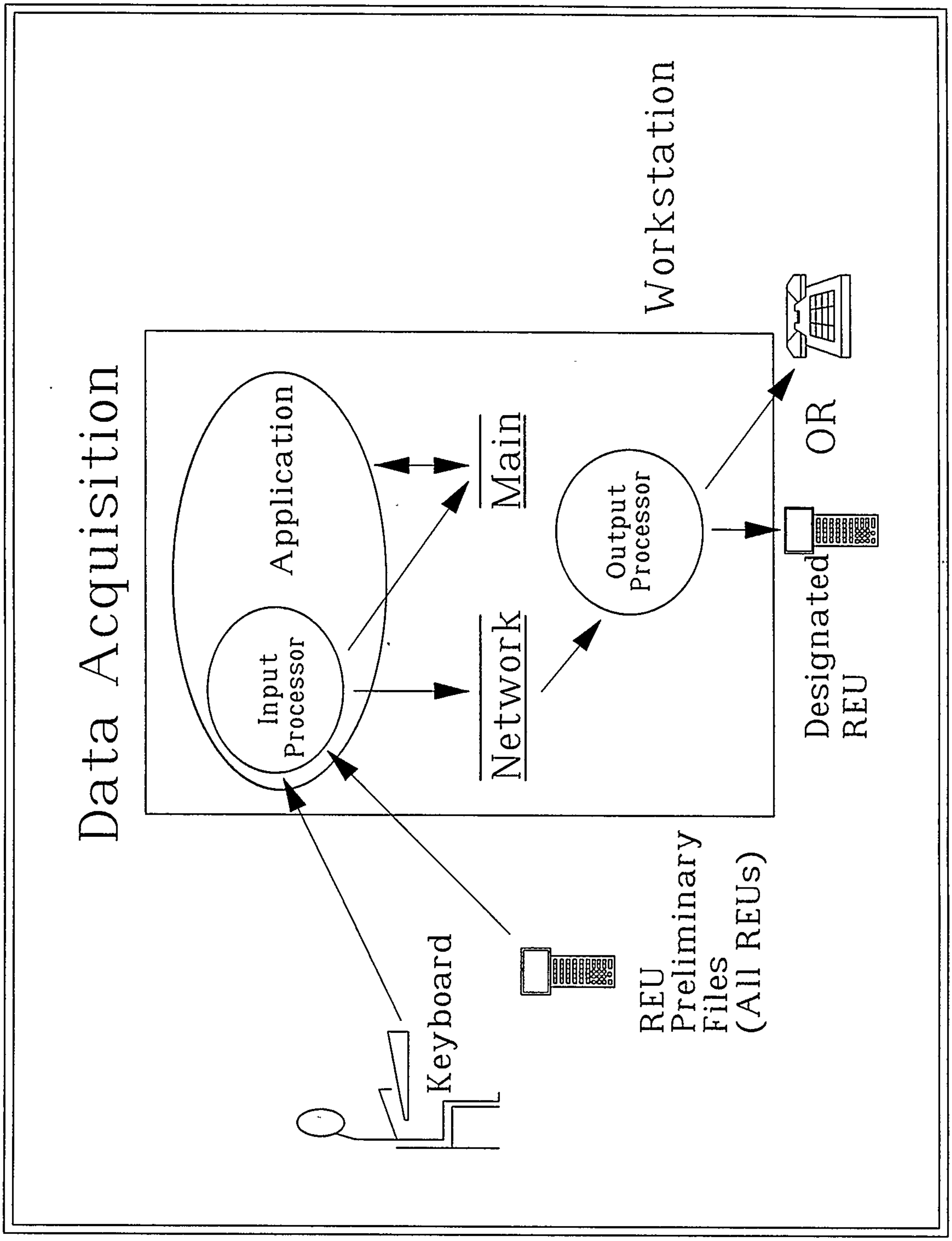

Figure 12: Data Acquisition

7.1 Acquiring Data By Keyboard

Data acquisition by keyboard is generally straight forward. Creation of virgin data is controlled by the application with appropriate validity checking of input values. Dynamic data acquired by keyboard (other than temporary records, see Section 6.1.1) is immediately copied into the network file for distribution to the CAO network. If the data applies to the local data set, it is also copied into the main file. Each record is automatically stamped with the appropriate transaction keys, although some keys may need to be operator entered or selected for the particular circumstances. The new record is always stamped with the current date and time from the system time clock.

Data edits are also a form of data acquisition which can be accomplished by keyboard. When an operator chooses to edit an existing record, he is given full control over the data items deemed changeable by the application. If a matching transaction had existed in the network file before the keyboard edit, the network file version is simply updated to reflect the new changes and the date and time stamp is replaced with the current date and time from the system time clock. No collision or timing exception processing is performed during keyboard changes.

If this is the first edit on a main file transaction, e.g. a copy of the transaction did not exist in the network file, the resulting edits are copied to the network file with the current date and time.

Manual override of data fields, although subject to the particular circumstances of the application, are generally allowed. Manual modification to transaction keys is generally disallowed if that modification will alter the uniqueness, or the standing identity of the transaction.

7.2 Handling Preliminary Files

Preliminary files are the working files of the REUs and are not addressed to an intended recipient station. Thus any time an REU connects to a work station or host the entire file is brought into the master PC for processing.

A work station always accepts all data from an REU and places copies in its network file for communication to the host. This is true even if the work station does not recognize the key for some of the records. It is assumed that the record will flow up the hierarchy (at least to a host) where enough system scope is collected to route the data to the appropriate stations. The work station will not retain a copy of the record in its main file unless it recognizes the key.

When a work station receives a transaction which contains a key it does not recognize, the CAO application will usually not process the data (e.g., it will be "filtered out" from the views, input, and edit screens the application presents to the user). This also prevents the user from modifying the data, which is appropriate since the work station does not have any ownership in the data if it cannot recognize the key.

Table 4 shows the procedure a work station follows when reading preliminary files from an REU. The host does an identical function except it processes against its collection file instead of a network file.

The procedure shown in Table 4 may be compared with the procedure the work station follows when receiving transit files from the host (see Table 2, Section 5.9).

```
                    MERGE PRELIMINARY DATA FROM REU
for each input record in preliminary file DO
 * if the record has delete flag set
     find key in main file
       if found
         delete this record
         copy deleted record to transaction file
       endif
     find key in network file
     if found set delete flag to "Y"
     else copy preliminary record to network file
     endif
     copy preliminary record to transaction file
 * else
     find key in network file
     if found and delete flag in network not set
       if collision
         copy network record to transaction file
         copy preliminary record to transaction file
         combine records using collision algorithm
         copy result to transaction file
       else
         combine records, use latest date & time
         copy preliminary record to transaction file
       endif
     else
       append preliminary record to network file
       copy preliminary record to transaction file
     endif
     if work station recognizes key
       find key in main file
       if found
         if preliminary more recent than main
           update main, use preliminary date & time
         else
           copy to network, use current date & time
           update main, use current date & time
         endif
       else
         append record into main file
       endif
     endif
 * endif
ENDDO
```

Table 4: Procedure For Handling Preliminary Files

Processing of deletions, collision handling with the network file, timing exceptions, and combining edits are handled identically between the two procedures. However, a copy of each record of the preliminary file (perhaps modified for edits, collisions etc., but not necessarily) will always be left in the network file. A copy of the transit file record will only be left in the network file if it has been modified for collisions, etc.

The network file will always retain a copy of each preliminary record for forwarding to the host, even if the work station does not recognize the transaction key as belonging to it. The work station will not update its main file if it does not recognize the key.

8 DEVICE AND SOFTWARE DESCRIPTION

The CAO architecture includes several computing devices and operating systems. The PDR resides on an IBM 3090 series processor within the VM/CMS operating system. Applications are written in a variety of languages using both relational and sequential databases. Connectivity of the CAO hosts to the main frame facilities is provided using commercial modem and main frame connecting software.

The operating systems for the CAO hosts, work stations, and REUs is MS-DOS, and the preferred programming language is currently DBASE compatible (FOXBASE or CLIPPER).

8.1 Why MS-DOS

The reasons for choosing MS-DOS as the operating system platform for CAO are:

1) Remote and portable field computing requires a personal computer platform. MS-DOS is the Amoco Corporate standard.

2) The disparate locations of the field staff make the support logistics from Calgary quite difficult. Choosing MS-DOS as the CAO platform levers the training and support capabilities of the Calgary based support staff since many community colleges, mail order courses, computer hardware vendors, and other sources provide a plentiful array of MS-DOS training courses for motivated employees. Many employees have the course fees subsidized by Amoco.

3) Most field staff who own PCs at home have MS-DOS compatible systems. Thus there was a higher degree of familiarity with MS-DOS among our intended user community.

4) Competition among MS-DOS compatible PCs makes them more cost effective than other operating system platforms.

5) Our Information Services department feels there is a lower cost to develop, train, and support in the MS-DOS environment than in alternative systems.

8.2 DBASE Programming Language

The CAO hosts, CAO work stations, and REUs are generally programmed in a *DBASE III* compatible language and use the *DBASE III* file structure. Using a consistent programming language and data structure across three of the four CAO hierarchy levels simplifies the sharing of data, and enhances development efficiencies. Many commercially available software packages can convert DBASE files, ensuring CAO data is easily accessible.

REUs are typically constrained by available memory. *DBASE* and compatible programs generally have high memory overhead. As the sophistication of REU based applications grow more memory efficient languages may be required.

8.3 CAO Host Configuration

The CAO host function is very processing intensive. The host consolidates the data of all the stations beneath it in the hierarchy, thus its data set is substantially larger than the data sets of CAO work stations. Thus the hardware of choice for the CAO host machines are 80386 based processors with large hard disks.

The CAO host used in the pilot is an IBM PS/2 Model 80 with 16MHz 80386 processor, 4MB system RAM, and 80MB hard disk, running MS-DOS 3.3. More recently, Amoco has been purchasing Dell Model 325 computers with 25MHz 80386 processors, 4MB system RAM, and 150MB hard disks, running MS-DOS 4.0.

8.4 CAO Work Station

The CAO work station computers perform less processing than CAO hosts and typically have a much smaller data set to operate on. Thus smaller capacity systems are often adequate for CAO work stations. In the pilot an IBM PS/2 Model 50 with 10MHz 80286 processor and 20MB hard disk, and a Toshiba model 3100 with 10MHz processor and 20MB hard disk were used. These machines run MS-DOS 3.3 and 3.2, respectively.

More recently, CAO work stations have been ordered with an 80386 processor and larger disk, in anticipation of 32-bit programs and operating systems and increased number of CAO applications requiring larger overall storage capacity. CAO work stations are now being ordered as Dell Model 310 with 20MHz 80386 processors, 90MB hard disks, and MS-DOS 4.0.

8.5 PC/5000 Technical Summary

The hand held computer chosen for implementation in the CAO pilot is the Micro Palm PC/5000. The PC/5000 is a MS-DOS 2.25 compatible hand held computer with up to 2.2MB of memory and an 80C88 processor. Table 5 is a technical summary of the PC/5000.[6] Micro Palm distributes with each unit a proprietary software product called CONNECT which causes the PC/5000 to appear as a disk drive to the work station and makes data transfer effectively transparent to applications.

The required multiplexer arrangement for the CAO host is provided by the Micro Palm four unit rack. Up to ten racks can be daisy chained together to provide connection to forty PC/5000s from a single host connection. Each PC/5000 responds to its own identity code so REUs do not have to be placed in preassigned slots. The host uses CONNECT to communicate with REUs in the rack. The transfer software simply issues the identification of the REU it wishes to talk to and then accesses the REU as a disk drive.

Feature	Details
Processor/OS	- 80C88, 4MHz clock, 0 wait-state. - MS-DOS version 2.25. - Built in system clock.
System Memory	- CMOS, 256KB standard, up to 1024KB. - 64KB PROM.
Disk Storage	- CMOS virtual disk memory. - 128KB, 256KB, 512KB, 1024KB modules. Fixed and removable memory can total 2.2MB. The removable memory is always configured as virtual disk drive. The fixed memory can be allocated to system RAM (maximum 640KB) and a second virtual disk drive, in 64KB segments.
Display	- 20-character by 8-line LCD. - PC compatible (80X25) virtual display buffer. - Back-light on panel.
Keyboard	- 51 key sealed full alphanumeric. - Side mounted shift and back-light keys. - Full 10 function keys. - Keys software programmable.
Input - Output	- Dual RS-232 ports (300 to 19.2K bps). - Full modem interface control. - Bidirectional 8-bit parallel port. - PC file transfer using CONNECT.
Environmental	- MIL standard for blowing rain, transit drop. - Petro chemical resistant and UV stabilized. - -30°C to 70°C operating range.
Dimensions	- Length: 26 cm. - Width (keyboard): 13.5 cm. - Width (grip): 8.75 cm. - Height: 8.1 cm. - Weight: 1.3 kg.
Power	- 3 "C" cell batteries. - Rechargeable NiCd or alkaline. - Lithium battery back-up.

Table 5: PC/5000 Technical Summary

9 CAO PILOT

> "Some people have ideas. A few carry them into the world of action and make them happen. These are the innovators."
> Andrew Mercer
>
> "Lord grant that I may always desire more than I can accomplish."
> Michelangelo
>
> "If you hit every time the target is too near or too big."
> Tom Hirshfield

A pilot of the CAO framework was tested at the Pembina Cardium facilities near Drayton Valley, Alberta, Canada. The Cardium oil pool is being water flooded and most of the wells are produced by beam pump. The purpose of the pilot is to test the feasibility of the CAO techniques under practical field conditions.

Operations personnel in Drayton Valley have been using FDE since late 1986. As discussed in Section 2.3, FDE was an excellent candidate for data transfer and was chosen for the pilot test. It was necessary to modify FDE to include the required EDT functionality. The data transfer software developed in the pilot was designed for FDE use only and thus is not transportable. However, the methods are generic and can be used in other applications.

Three types of data in the application (well tests, gas charts, and fluid transfers) are ideal for pilot testing because of the high degree of communication and data sharing required across the network. Two of these data types are generated by a special system and supplied to the CAO network in an SDI process.

District personnel directly involved with the pilot include five operators, one field foreman, one District foreman, one systems support person, and one administrative clerk.

9.1 Pilot Project Development

Chapters 1 and 2 briefly discussed the history of CAO and why the pilot project was required to test data transfer. A greater understanding of the prevailing business, market, and project events is required to understand why the CAO pilot technology was developed the way it was. These "external" factors eventually led to the decision mid way through pilot development to abandon the hand held system selected at the beginning of project development for a different vendor (Micro Palm). This significantly impacted the system implementation design and the way the PCs communicated with the hand held computer systems.

Throughout late 1986 and 1987 Amoco Canada made several tentative steps in the use of hand held computers and in interfacing them to FDE. In late 1987 a proposal for a five year plan to develop the full CAO philosophy of portable field computing, one time data capture, and multi-directional data transfer was developed and worked out with the Pembina District who agreed to sponsor a field test. In January of 1988 the proposal was taken to the Amoco Canada Production Committee (senior management) for approval. Management liked the plan but wanted a more aggressive approach. They requested a plan to install a CAO field test within six months.

Although it was believed the project could be developed in six months, it was felt the technical capabilities of products on the market at the time, combined with the fact the data transfer techniques had never been done before, gave this project a much higher risk of failure than is normal for business computing development. The project was therefore recommended as an experimental pilot project.

This project plan was taken back to committee in early February with a recommendation to proceed as an R&D pilot. Delivery by August was promised on a "best effort" basis. Project cost, with a 20% contingency due to risk, was estimated at $250,000. Management approved the project on the R&D basis.

An extensive market search and vendor screening for hand held computers had been completed in late 1987 as part of our earlier initiatives. A preferred hand held vendor had been selected. Although all products available at the time had limitations, the chosen product was capable of meeting the project requirements at a reasonable cost. Since the vendor ultimately failed to meet its commitments (described below) they will not be identified in this thesis. The author, and Amoco Canada, have no wish to embarrass the vendor.

Representatives from the vendor were invited to Calgary at the beginning of the project to discuss their role. The vendor's product used remote-controlled communications, whereas the pilot required host-controlled communications. The vendor agreed to supply a custom communications module to provide the desired host-controlled communication method.

The vendor sent several revisions of the module to us but for various reasons was unable to supply a final working version. By early August the Amoco development portion was complete and ready for system testing. We were waiting solely on the communication module.

The first of the truly MS-DOS compatible hand held computers were just emerging on the market. The Micro Palm unit, described in detail in Section 8.5, met all criteria and was chosen as a replacement. So the original vendor was dropped.

The delays due to the original vendor, time to perform the technical assessment of Micro Palm, and the additional cost of the new hardware not planned for, were significantly impacting the project budget. To complete the pilot within the allotted funding it was necessary to make two critical decisions about system performance.

The first decision impacted the REU data capture function. The data capture and processing software for the original brand of REUs was developed and expensed to the pilot project between February and May of 1988, before discovering the vendor would not be able to deliver the communication module. The programs to do this were written in the proprietary language of the vendor's product which was completely incompatible with Micro Palm. Thus the development cost for these modules had to be written off. The cost to redevelop them would have overrun the project budget.

To stay within the budget it was decided to drop the objective of REU data entry for the pilot so we could concentrate on our primary objective of proving the data transfer techniques. It was considered less necessary to test hand held data capture for the following reasons:

1) The data capture functionality, including processing of REU entered data during the work station and host transaction processing, had been fully tested during development tests and was considered functional.

2) Several tests (not directly related to the pilot) were completed in 1987 using REU data capture for up load of information to FDE. These tests had all been proven reliable and functional.

3) There was a related project in Amoco Canada that was using REUs purely for data capture and up load

to a PC system. This project was installed and fully operable. Other Amoco subsidiaries had developed REU based data capture systems (Glynn, [7]).

4) Other oil companies have developed systems using hand held computers or calculators for data capture and up load to PC systems (Moore [8], and Bettis and Shreve [9]).

5) Systems of simple data capture using hand held computers were quite common in several North American industries. Data acquisition by REU, although desirable in the pilot, was not considered to be a substantial step forward in technology and could be sacrificed in the interest of testing more progressive contributions.

It would still be necessary to use the Micro Palm REUs as the communication medium for the following reasons:

1) We wished to test the practicality of using the REU as a communication medium between processors for those field sites which did not have telephone or other convenient communications available.

2) The REUs provided the addressing and multiplexing of more than one communication session at the host in a single unattended batch process. This capability could not be duplicated using diskettes.

3) The system software had been developed on the expectation the REU would perform the communication function.

4) The REU data acquisition software could be easily added to the system should additional funding become available.

The second area of impact caused by the switch in vendor technology was with the interface routines between the Amoco developed transaction processing software and the communications software for the REUs. A cost analysis of the system determined an interim solution could be built by modifying the existing interface routines to the minimum extent necessary to recognize the Micro Palm REUs. The minimum modifications would retain several intermediate communication steps required by the original vendor's technology but not required in the more advanced Micro Palm. Despite the non-optimum nature of the communications software, it was more cost effective to perform the minimum modifications. The communications interface became known as the "black box" and is briefly described in Section 9.11.

9.2 CAO Pilot "As Built" Hierarchy

The CAO hierarchy was used as the basic model for constructing the CAO pilot system. However, the pilot scope included only the PC portion of the hierarchy. One special system (NGAS) was involved, but no SCADA computers were included. Figure 13 shows the CAO hierarchy built for the pilot. This figure may be compared with Figure 1 in Chapter 3.

The pilot includes one CAO host, two field work stations, and three REUs. Also included in the pilot is a second computer at the field area office which provides gas volumes from chart integration to the CAO network (called the "NGAS" computer, see Section 9.4). The CAO host provides information transfer to the main frame but two way transfer over this connection was not built for the pilot.

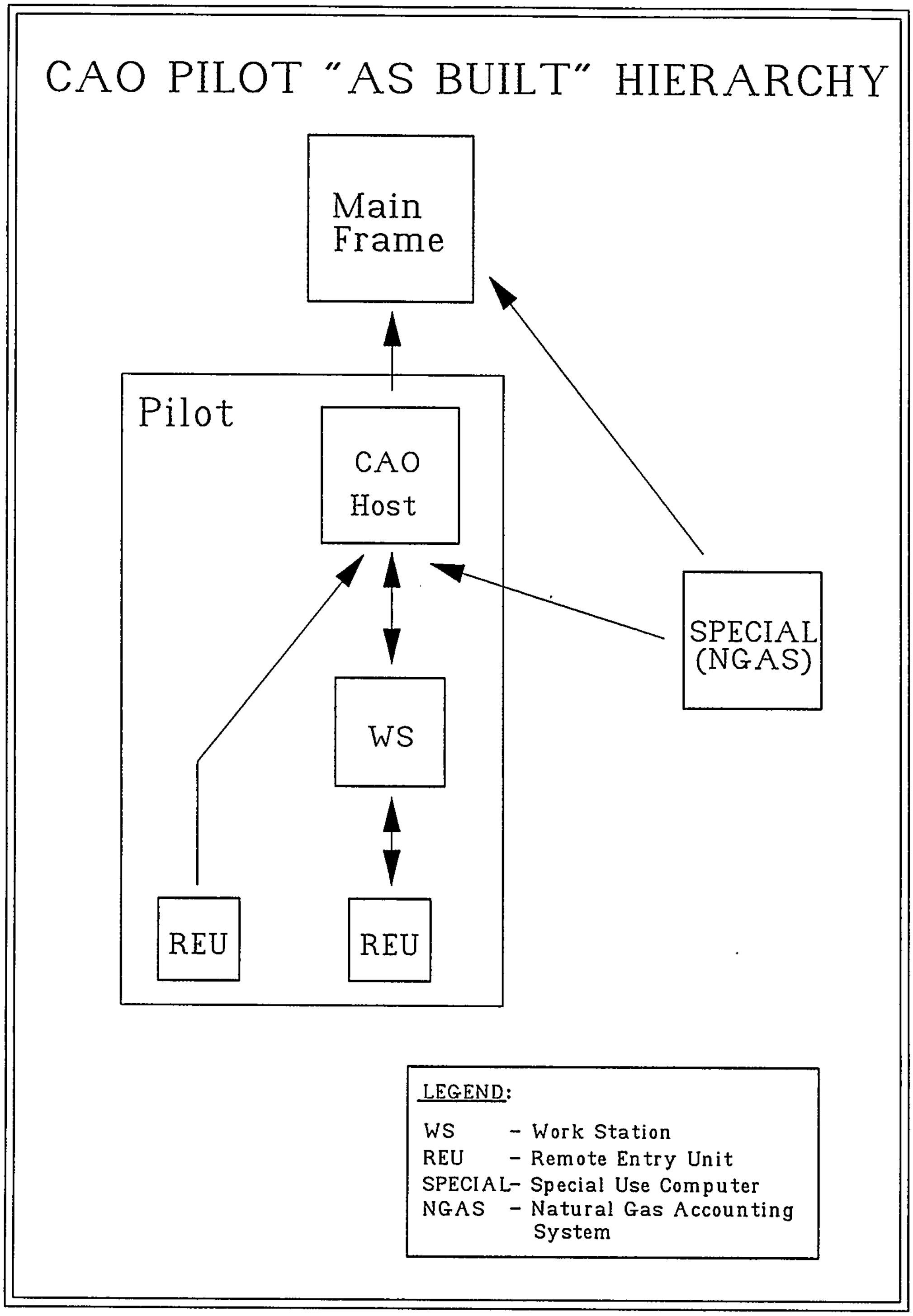

Figure 13: CAO Pilot "As Built" Hierarchy

Figure 14 shows how the pilot stations interact. Not shown in the figure are the distances involved. The CAO host and the NGAS computer are located in the Pembina District Office in Drayton Valley. The two remote work stations are 10 and 20 miles, respectively, from the District Office. There are no communication facilities where the work stations are located so two of the three REUs are designated for CAO networking.

All of the processing software to handle the data acquisition from the "free" REU was developed and installed in the system. It also passed development testing in Calgary. For reasons outlined in Section 9.1 the data capture application software was not developed for the REU so the third "free" REU was not actually used in the pilot.

9.3 CAO Pilot "As Built" Network Routing

Figure 15 shows the data control sequence for meter run numbers as built for the CAO pilot. This method differs somewhat from that shown in Figure 11 in Section 6.8 in that the meter run number is related to a battery, then a FLAC, and finally the designated REU for communication.

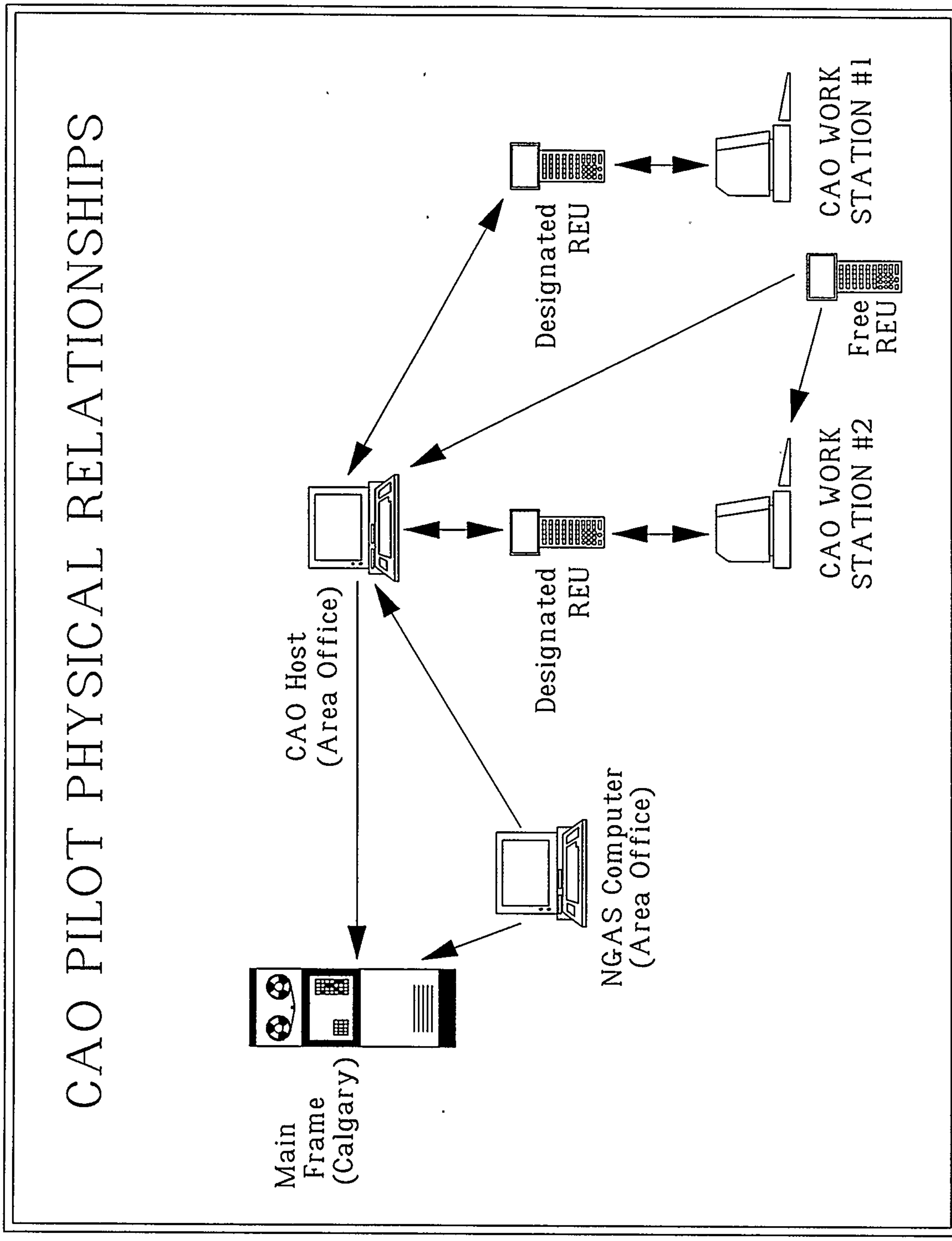

Figure 14: CAO Pilot Physical Relationships

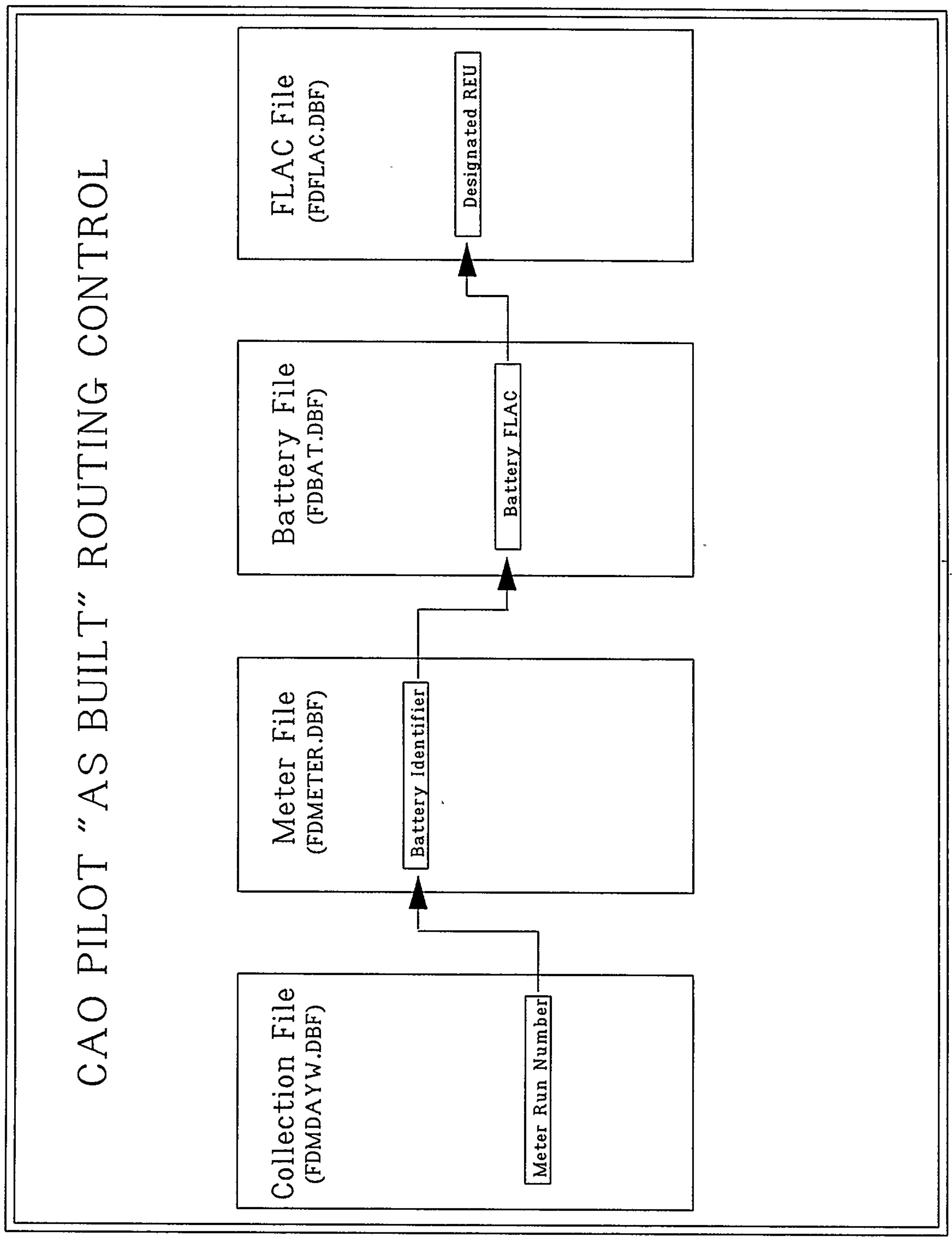

Figure 15: CAO Pilot "As Built" Routing Control

The collection file (FDMDAYW) shown in Figure 15 contains the accumulated transactions of the daily meter volumes for each meter run number tracked within the system. The meter run number is then used as a key to the meter file (FDMETER) from which the battery identifier for the group gas meter is found. The battery identifier is then used as a key to the battery file (FDBAT) where the battery FLAC identifier is obtained. Finally, the battery FLAC is used as a key to the FLAC file from which the SIC of the designated REU is obtained.

Relating a meter number to a battery FLAC was a matter of convenience for the pilot and works for this specific application of metered sales gas; metered sales only occurs from a battery which has an associated battery FLAC. This form was considered the most convenient from a system administration viewpoint; only the FLAC file needed to be configured in detail for the data routing. However, this short cut would not work in the general case and the meter run number would need to be directly associated with a list of related stations as described in Section 6.8.

Table 6 gives the structure for the FLAC database file used in the CAO pilot (FDFLAC.DBF). There is one record for each FLAC assigned.

Table 7 gives the structure for the meter run database file used in the CAO pilot (FDMETER.DBF). There is one record for each meter run number defined. FDMETER contains the static data which describes each meter run and how it is used in battery gas measurement. It was decided to keep the FDMETER file structure identical to the same file used in the POWAR database. The POWAR application provides field estimation of gas volumes by entry of field estimates of the gas volume measurement parameters from which POWAR calculates the estimated gas volume. FDE does not have this capability so many of the data fields are not used. The compatibility was enforced to simplify integrating POWAR to the CAO network and EDT procedures.

Database: FDFLAC.DBF

Field	Field Name	Type	Width	Comment
1	REU	Character	4	Designated REU SIC
2	WELL_ID	Character	8	Well identifier
3	BAT_ID	Character	8	Battery identifier
4	FLACCODE	Numeric	9	FLAC code
5	T_TYPE	Numeric	1	Reserved (not used)
6	DESC	Character	35	Facility descriptor
7	OPERATOR	Character	8	Operator identifier
** Total **			74	

Table 6: CAO Pilot FLAC File Structure

Database: FDMETER.DBF

Field	Field Name	Type	Width	Dec	Comment
1	BAT_ID	Character	8		Battery identifier
2	FDGID	Character	3		Meter group id.
3	FDMDATE	Date	8		Meter active date
4	FDMNUM	Character	4		Meter run number
5	FDMSEQ	Numeric	2		Not used
6	FDMGID	Character	1		Not used
7	FDMPRODID	Character	1		Not used
8	FDMTYPEID	Character	1		Not used
9	FDMCALCID	Character	1		Not used
10	FDMLOC	Character	16		Local meter name
11	FDMCALCF	Character	17		Not used
12	FDMDIFFLG	Character	1		Not used
13	FDMPRESFLG	Character	1		Not used
14	FDMBATACCT	Character	1		Not used
15	FDMGRPACCT	Character	1		Not used
16	FDMDIFFRNG	Numeric	9	5	Not used
17	FDMPRESRNG	Numeric	9	4	Not used
18	FDMFACTOR	Numeric	7	4	Not used
19	FDMEMUL	Numeric	6	2	Not used
20	FDMORIFICE	Numeric	6	2	Not used
21	FDMRUN	Numeric	6	2	Not used
22	FDANUM	Character	3		Not used
23	FDMFLAC	Numeric	9		Not used
** Total **			122		

Table 7: CAO Pilot Meter File Structure

9.4 NGAS

The Pembina District uses a commercially available natural gas accounting system to integrate and store the gas charts generated in the local operations. The program's commercial name is the "Natural Gas Accounting System", or "NGAS".

Although SCADA systems and other forms of electronic gas measurement are changing the way produced gas is measured in oil fields, there are still many sites that measure gas using a standard orifice meter. Paper charts record the static pressure, differential pressure, and temperature in the orifice meter over a given time period (usually 24 hours or 7 days). These paper charts are then integrated and converted to gas volumes using programs like NGAS.

The Pembina Cardium pool, where the pilot is located, is a very mature oil field that uses paper charts for all field gas measurement where SCADA monitoring systems have not been installed. Approximately 5000 paper charts are processed in the Drayton Valley office each month for District production accounting.

Gas volumes are calculated from observed process parameters which form the independent variables. The principle variables are the differential pressure across the orifice, static pressure, and flowing temperature. Gas volumes are therefore derived data values.

Although each station could recalculate the gas volumes given the independent data, CAO treats gas volumes as independent data; there are three reasons:

1) The calculations rely on many factors and coefficients. It is easier to communicate just the gas volume which is usually the value of primary interest.

2) Rigorous gas volume calculations could significantly degrade operating performance in low powered CAO stations (such as REUs).

3) Auditing standards require that only a single source of gas volumes be acceptable for any one gas meter. Typically an accounting or SCADA system is chosen as the audit source. Thus the audit source

numbers are transferred to the CAO network for communication, thereby providing consistent numbers for all parties.

9.4.1 SDI File Structure For NGAS

Table 8 is an example of the SDI file (AMOCO.XE1) containing sample well test and gas chart information as supplied by the NGAS computer. The structure of the file was determined in design sessions with personnel handling the main frame accounting system and the SCADA engineering section. For explanatory purposes a brief discussion on the file structure is given following:

In the record:

MNTH900190012514560

MNTH - identifies the transaction card type as a header for the monthly production data.

9001 - identifies the batch production month as January 1990.

900125 - is the date the batch was generated.

AMOCO.XE1 SDI File

```
MNTH900190012514560
CHT 1010      900101    3.98 N
CHT 1010      900102    3.66 N
CHT 1010      900103    3.69 N
CHT 1010      900104    3.30 N
CHT 1010      900105    3.84 N
CHT 1010      900106    2.56 N
CHT 1010      900107    3.82 N
CHT 1010      900108    3.24 N
CHT 3961      900101    1.59 N
CHT 3961      900104    1.32 N
CHT 3961      900105    1.80 N
CHT 3961      900106    1.67 N
CHT 3961      900107    1.77 N
CHT 3961      900108    1.78 N
CHT 6139      900101    4.10 N
CHT 6139      900102    4.11 N
CHT 6139      900103    5.33 N
CHT 6139      900104    4.38 N
CHT 6139      900105    4.27 N
CHT 6139      900106    4.13 N
CHT 6139      900107    4.04 N
CHT 6139      900108    4.05 N
WTST990814014 9001012011 1.44 0.32 0.180068624.000 N
WTST990723017 9001032011 3.95 1.37 1.440067824.000 N
WTST990770018 9001042011 1.15 0.39 0.070069324.000 N
WTST990759011 9001052011 3.53 3.16 8.800068824.000 N
WTST990758013 9001072011 1.60 1.65 1.250068024.000 N
WTST990780017 9001082011 1.68 4.58 9.510069724.000 N
WTST870561016 9001026113 1.18 0.18 0.190041124.000 N
WTST870520012 9001046113 2.19 0.36 6.090040624.000 N
WTST870515012 9001066113 2.35 0.35 0.200042024.000 N
WTST870516010 9001086113 1.73 0.23 0.200041924.000 N
WTST870770013 9001056120 2.20 0.28 9.400027224.000 N
WTST870771011 9001076120 2.20 0.26 8.100027524.000 N
WTST877828012 9001046132 0.32 1.20 0.000027424.000 N
WTST879201010 9001066132 0.30 1.43 0.000028424.000 N
```

Table 8: Example of the AMOCO.XE1 SDI File

14560 - is the time the batch was generated, in military format (equal to 2:33:36 PM)

In the record:

```
CHT 1010      900101    3.98 N
```

CHT - identifies the transaction card type as a basic (or group) gas chart.

1010 - is the gas meter run number (see Section 6.3).

900101 - is the chart date (beginning of gas flow measurement period)

3.98 - is the chart period calculated gas volume (10^3m^3).

N - shows the data has not been flagged questionable by the operator.

In the record:

```
WTST990814014 9001012011 1.44 0.32 0.180068624.000 N
```

WTST - identifies the transaction card type as a well test gas chart.

990814014 - is the well FLAC, horizon sub code, and check digit (see Section 6.2).

900101 - is the date the well test began.

2011 - is the gas meter run number (see Section 6.3) which also identifies the test vessel in this context.

1.44 - is the measured oil production during the well test period (m^3).

0.32 - is the measured gas production (as calculated by NGAS from the gas chart) during the well test period (10^3m^3).

0.18 - is the measured water production during the well test period (m^3).

00686 - is the average static pressure in the test vessel over the test period (kPag).

24.0 - is the time period of the well test (hours).

00 - is a code to identify the reason a well was not tested. "00" shows a normal well test.

N - shows the well test has not been flagged questionable by the operator.

Table 8 gives some example records of the AMOCO.XE1 file. The actual number of records transferred between NGAS and the CAO host grows each day of the month reaching approximately 5000 records at month end, although only 15% of these charts actually relate to the pilot facilities.

The SCADA, NGAS, FDE, POWAR, and main frame accounting systems all use the identical file structure for transfer of gas volume and well test data. This was done for several reasons including the ease with which various systems can be linked for transfer of data. The Pembina field facilities chosen for the pilot are not monitored by SCADA systems although nearby Amoco operated facilities are. The identical file structure could then be exploited easily when this technology is extended to the SCADA monitored leases. SCADA data could be transferred to the CAO host as easily as NGAS data. The SCADA systems automatically determine gas volumes and perform well test and group gas measurement under automatic control.

9.5 Well Tests

Technical, operational, contractual, financial, and regulatory concerns require that oil, water, and gas production be determined separately for every oil well (specifically, each oil well completion horizon). It is generally uneconomic to accurately measure these three separate commodities at each well on an ongoing basis. The practical alternative is to share the process and measurement facilities among several wells and sequentially sample each well for measurement on a periodic basis. The measurement sample is called a "well test".

The minimum frequency and duration of tests for each well is set by the provincial regulatory agencies (the ERCB in Alberta) and possibly by contractual stipulations. The minimum is one twenty-four hour test per month for most mature fields under conventional (water flood) recovery .

The testing equipment consists of a process vessel and associated oil, water, and gas measurement equipment. Usually this consists of separate positive displacement meters for the oil and water, and an orifice meter on the gas line.

Orifice measurement of gas at the pilot locations is recorded on a standard dry flow paper chart integrated at the District Office, using the NGAS computer.

Figure 16 shows the general flow of well test information. It begins with a paper chart being removed from the chart recorder on the test vessel at the completion of a well test. This chart is then delivered to the chart integration clerk in the District Office who integrates the chart on the NGAS computer to obtain the gas volume. The measured values for oil and water, which had previously been written on the chart by the operator, are keyed into NGAS by the clerk. Once each day the clerk has the NGAS computer prepare the AMOCO.XE1 file which is copied by diskette to the CAO host. The host then reads the file through an SDI, and performs full transaction processing on the records that pass through the filter process. The well test in question is related by FLAC number to the SIC of the operators work station. The record is placed in the transit file for that PC. The transit file is then deposited in the designated REU for that station. The next morning the operator retrieves his REU and drives to his lease. When he reaches his work station location he connects the REU to the PC. The well test is then copied into the main file of his work station.

Figure 17 is a context diagram showing the total EDT context for well test data on the CAO pilot.

9.6 Gas Charts

Gas production at the pilot facilities is measured using standard orifice meters and paper charts. Before installing the CAO pilot operators had to rely on a paper system to obtain gas volume information. Gas charts were brought to the Drayton Valley office each afternoon to be integrated the following day. Every morning each operator was given a printed report of gas volumes from all the newly integrated charts. It was the operator's responsibility to scan the report to pick off the chart volumes relating to his facilities and enter them into his PC. Manual transfer of gas volume information was a time consuming and error prone process.

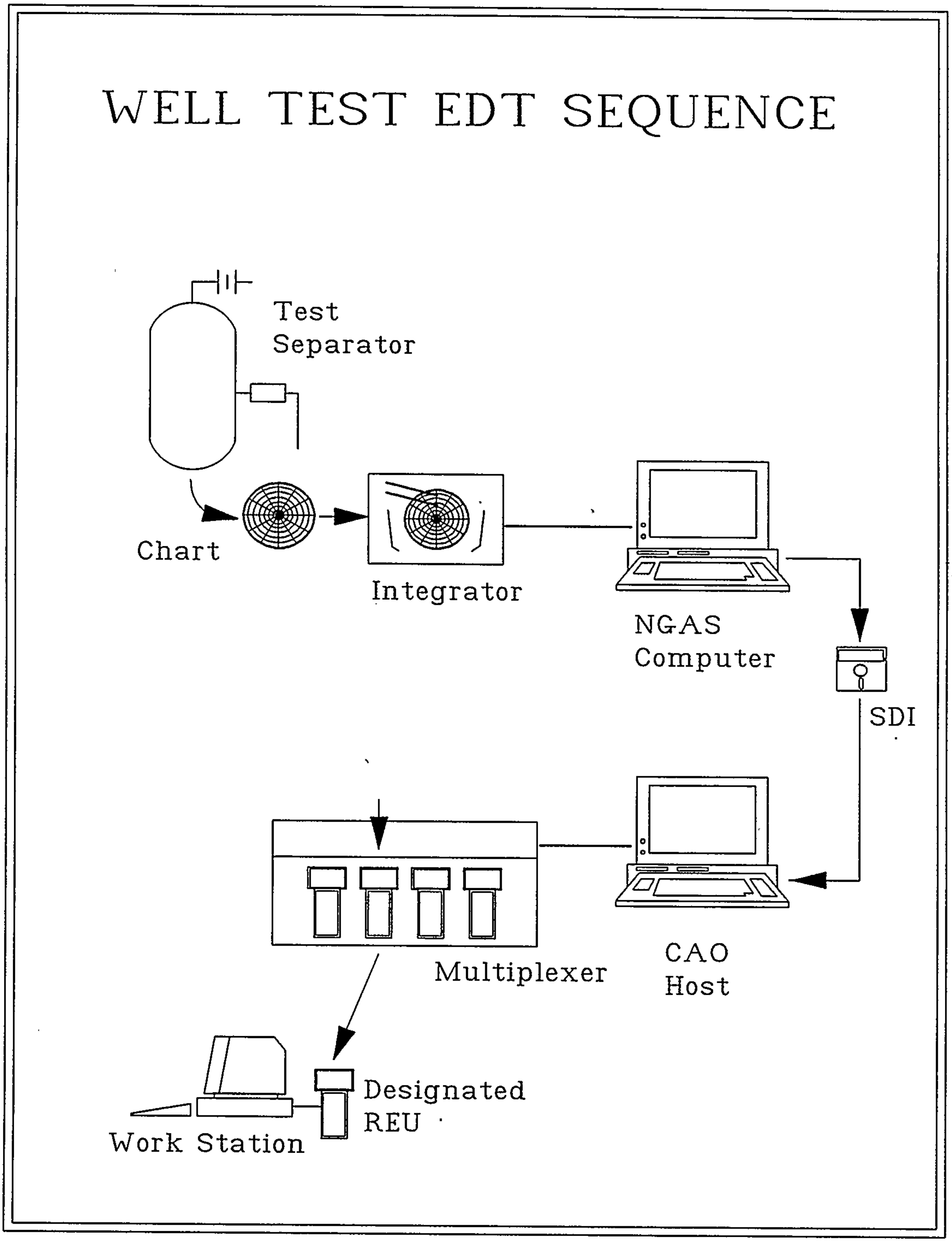

Figure 16: Well Test EDT Sequence

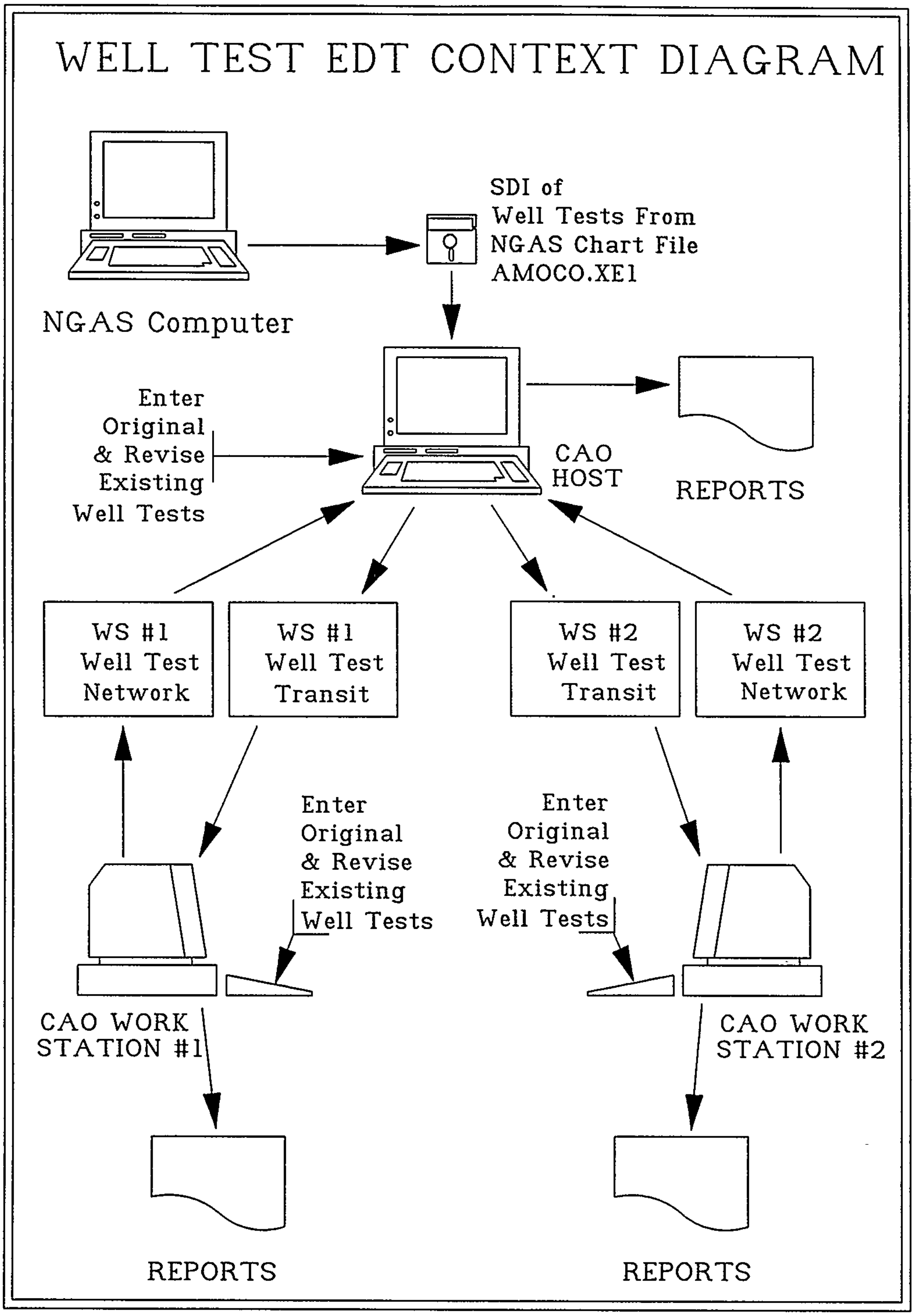

Figure 17: Well Test EDT Context Diagram

Transfer of gas volume information to the field PCs is automatic with CAO. The gas chart integration computer, running a commercially available gas chart integration program, was modified to provide gas chart records in the AMOCO.XE1 format. Each day the gas chart integration computer prepares a data file of all newly integrated charts. The CAO host processes the data file treating each gas record as a transaction. The gas volumes are then passed over the network to the recipient stations using the REUs as the communications link. Each station receives only the gas volumes relevant to it. The CAO work stations automatically use the gas volumes to calculate actual and theoretical production.

9.6.1 Meter Modeling

Meter modeling consists of grouping gas meters (by meter run number) which are summed to provide the total gas applicable for a particular receipt or disposition in the battery balancing equations.

The original FDE required all gas volumes to be summed as appropriate and entered each day by the operator. (The only exception is the estimated daily lease fuel gas which is established for the battery at the time the program is configured and is carried forward each day by FDE.) This method was not applicable for the CAO pilot where individual gas volumes are transferred electronically. The individual volumes must be combined, as appropriate, for each type of gas receipt or disposition. This computation is referred to as meter modeling.

FDE was modified to handle meter modeling and to configure the desired form of entry handling. Analysis of the various scenarios identified four distinct handling methods. The desired handling method is selected at system configuration time by choosing one code for each meter group at a facility. The modified FDE program will process the gas information and perform screen and entry handling as directed by the code selection. The codes are:

- **M - metered input:** each meter in a group is assigned a volume and is summed by the program to arrive at a group total. The meter volumes normally come from the NGAS system but can be hand entered or modified by the operator.

E - keyboard entry: the associated gas volumes are prepared by the operator and manually entered into the computer as one volume.

C - carry forward: the previous days value is carried forward. It cannot be modified.

B - both keyboard entry and carry forward: the previous days value is carried forward but can be modified by the operator if desired.

The four handling methods portrayed above determine how an individually metered gas volume would be handled. It is also necessary to determine the type of receipt or disposition of each metered volume. The pilot facilities have eight types of receipts and dispositions which formed a sufficient number of groups for modeling the gas measurement at these batteries.

Table 9 lists the groups, the result they have on battery gas measurement, and lists the group code for each type. The group code is stored in the FDGID field of the FDMETER file (see Table 7, Section 9.3). FDE uses this code to determine how to include the gas volumes of the particular meter in the battery gas measurement model.

Group Code	Group	Result in Battery Balancing Calculation
G01	**Metered sales gas:** Gas sold to customer through one or more meters.	**Credited** to Production
G02	**Up stream receipts:** Gas received from another facility, and commingled with process gas upstream of sales meters.	**Subtracted** from Production
G03	**Up stream fuel:** gas removed up stream of sales meters; used to supply process energy.	**Credited** to Production
G04	**Up stream flare:** gas discharged to flare upstream of sales meters.	**Credited** to Production
G05	Not used.	N/A
G06	Not used.	N/A
G07	Reserved for Tank Vapors. Not used.	N/A
G08	**Down stream receipts:** gas received from another facility, but tied into the process stream after the sales meter.	Tracked for accounting purposes, but has **no effect** on the battery production calculation.
G09	**Down stream fuel:** gas removed down stream of sales meters; used to supply process energy.	Has **no effect** on battery production calculation, but is **subtracted from metered sales** for revenue purposes.
G10	**Down stream flare:** gas discharged to flare downstream of any sales meters.	Has **no effect** on battery production calculation. Treatment varies for revenue calculations.

Table 9: Gas Meter Group Identifiers

The FDBAT file (whose structure is given in Table 10) stores the information which describes the static information about each battery. Twenty-two out of forty fields describe the various aspects of the battery gas measurement necessary for FDE to model. For example, "HAS_USFLAR" contains a logical indicator of whether the battery has a flare upstream of the sales point. "USFLAR_TYP" stores the code to determine if the volume entry is metered input, keyboard entry, etc. If this code shows metered input, FDE will sum all the volumes for each meter in the FDMETER file which have the given battery FLAC and the group identifier code "G04". The daily volumes are stored in the FDMDAY (meter-day) file.

Database: FDBAT.DBF

Field	Field Name	Type	Width	Dec	Comment
1	BAT_ID	Character	8		Battery identifier
2	BAT_NAME	Character	30		Battery descriptor
3	BAT_FLAC	Numeric	9		Battery FLAC number
4	PAGES	Numeric	2		Pages of well data
5	INJ_PAGES	Numeric	2		Pages of inj. wells
6	FRE_PAGES	Numeric	1		Pages of water wells
7	HAS_MSALES	Logical	1		Has metered sales
8	MSALES_TYP	Character	1		Entry handling
9	HAS_USRCPT	Logical	1		Has u/s receipts
10	USRCPT_TYP	Character	1		Entry handling
11	HAS_DSRCPT	Logical	1		Has d/s receipts
12	DSRCPT_TYP	Character	1		Entry handling
13	HAS_USFLAR	Logical	1		Has u/s flare
14	USFLAR_TYP	Character	1		Entry handling
15	HAS_DSFLAR	Logical	1		Has d/s flare
16	DSFLAR_TYP	Character	1		Entry handling
17	HAS_USFUEL	Logical	1		Has u/s fuel
18	USFUEL_TYP	Character	1		Entry handling
19	HAS_USFMTR	Logical	1		U/s fuel metered?
20	USFMTR_TYP	Character	1		Entry handling
21	HAS_DSFUEL	Logical	1		Has d/s fuel
22	DSFUEL_TYP	Character	1		Entry handling
23	HAS_DSFMTR	Logical	1		D/s fuel metered?
24	DSFMTR_TYP	Character	1		Entry handling
25	HAS_TNKVAP	Logical	1		Has tank vapors
26	HAS_GRTPRS	Logical	1		Has grp. trt. press.
27	HAS_VRU	Logical	1		Has vapor recovery
28	GAS_SOLNGF	Numeric	9	7	Solution gas factor
29	PF_LIM_OIL	Numeric	6	4	Oil proration limit
30	PF_LIM_WTR	Numeric	6	4	Wtr proration limit
31	PF_LIM_GAS	Numeric	6	4	Gas proration limit
32	BAT_INIT	Character	3		User initials
33	DISTRICT	Character	10		District name
34	VERSION	Date	8		Program version
35	LAST_OIL	Date	8		Last oil entry
36	LAST_WTR	Date	8		Last water entry
37	LAST_GAS	Date	8		Last gas entry
38	LAST_LOAD	Date	8		Last load oil entry
39	LAST_PRAT	Date	8		Last proration
40	RECALC_DAT	Date	8		Date to recalculate
** Total **			170		

Table 10: FDE Battery File Structure

9.7 Screening NGAS Data

The NGAS system is capable of producing a list of all charts (group gas and well test) that apply within a given accounting day. Most charts are integrated in the Drayton Valley office within twenty-four hours of receipt. However, for some locations there is a delay between the chart date (the day the chart was used to measure gas flow) and the day it is received in Drayton Valley. To transfer all the newly integrated charts each day to the CAO host, the NGAS operator had to pull all charts with chart dates falling within a certain range.

For practical purposes, the regular routine of the operator was to pull all charts with a chart date from the beginning of the month to the present date. This meant that redundant information was being transferred to the host from NGAS each day. The SDI filtering process discussed in Section 5.5.1 was built for NGAS to eliminate the redundant transfer and handling at the host.

9.8 Fluid Transfers (C-19s)

A common occurrence in field operations is the transfer of produced fluids between facilities for processing or well repair work. For production and revenue purposes these transfers must be appropriately tracked. The PC based production monitoring programs record the fluid volumes as fluid transferred out from the facility where the truck received its load and as fluid transferred in where the truck delivers its load.

Before CAO, a paper form called a "C-19" was manually prepared by the operators for each shipment of fluid between facilities. The operator of the shipping facility would complete the form including the quantities of oil and water shipped. A copy of the form would be forwarded to the operator of the recipient facility. Both operators would then enter the transfers into their PC production monitoring system. A data entry clerk at the field area office would key a third copy of the transaction into the main frame accounting system.

The CAO network eliminates the redundant entry of C-19s and reduces the potential for data errors by automatically networking the transaction. The issuing operator enters the transfer information into his PC as before. An electronic copy is automatically forwarded through the CAO host to the recipient operator's PC. No further manual work on that item is required.

C-19s are the most complex of the data items built in the CAO pilot because of the dual ownership of the data and the emphasis placed on the control of the data.

9.8.1 C-19 Number Assignment in the Pilot

The pilot involved only two stations in an operating area where many other facilities regularly transfer fluids. The Cardium oil production presently uses a paper system for reporting C-19s. It is impossible to eliminate the paper system for the two sites participating on the pilot as they both transact paper C-19s with other batteries.

It was therefore decided to modify FDE for entry of C-19s from the paper system. The C-19 test on the pilot would therefore involve the electronic capture of C-19s resulting from all fluid transfers where the pilot facilities were involved. The pilot would further test the full three-way electronic transaction processing and networking for fluid transfers which occurred directly between the two pilot station locations.

The District would still use paper C-19s to track fluid transfers. Thus it was necessary to key punch entry of the C-19 number into FDE from the paper form. The C-19 number generation algorithm described in detail in Section 6.4.1 was developed directly from the work of this thesis, but was not included in the pilot. It was built into POWAR, however, in readiness for converting POWAR to the CAO EDT process. POWAR also allows keyboard entry of paper C-19 numbers until the conversion is complete.

9.8.2 Three Way Notification

C-19s are a transaction type which tend to be mutually owned by two stations. The host also retains a copy of the C-19 where it can be processed, viewed, and edited, similar to the work stations. Unlike the group gas and well test transactions, C-19s regularly involve three stations in each event.

The C-19 number is the transaction key which uniquely identifies each C-19 transaction. C-19s represent transfer of fluids between two facilities so the source and destination facilities are identified in the C-19 record by their FLAC number. The FLAC numbers are used to identify the work stations which require a copy by relation to the SIC. The host always retains a copy of the C-19 by virtue of its role in the hierarchy.

The C-19 number is assigned by the creating station when a C-19 is created. Administrative procedures in the Pembina District stipulate that the operator of the facility that sourced the fluid is responsible for generating the C-19. Once into the system a copy is forwarded to the CAO host. The host uses the FLAC numbers to route the C-19 to the associated stations.

Sometimes both the source and destination facilities reside on the same station; only one copy of the record is forwarded to the recipient station in this case.

Edits made to the C-19 transactions are routed using the same techniques. The C-19 edit records always contain the transaction keys, and hence always contain the information necessary for transaction identity and routing. Only the data fields are edited and only non modified data fields are ever blank.

FLAC Code Changes

Although FDE has been designed to simplify selection of FLAC numbers when creating C-19s, it is still a common error in the field to assign the wrong FLAC to the source or destination record field. Operators are allowed to modify the FLAC field to accommodate correction of FLAC code errors.

When a C-19 record has one or more FLAC fields edited, two copies are generated for communication to the network. The first record contains an image of the C-19, complete with the original FLAC of the field (or fields) which have been modified. The delete flag is set to "Yes". In this way a delete notice is sent to the station that incorrectly received a copy of the transaction.

The second transaction created includes the proper (as modified) FLAC or FLACs, complete with all the data fields in the C-19. This transaction is then communicated to the network and ensures the latest version of the record exists in each of the involved stations.

C-19 Number Changes

Circumstances of the pilot required operator entry of C-19 numbers from paper forms. Although errors were rare, it was nonetheless possible for an operator to enter the wrong C-19 number. Thus it was necessary to provide editing of the C-19 number for the CAO pilot.

The procedure is similar to the changing of the FLAC numbers. A copy of the transaction with the incorrect C-19 number is made in the network file with the delete flag set to "Yes". A copy of the transaction with the correct C-19 number is then also made to the network file. Thus occurrences of the C-19 with the incorrect C-19 number are erased and the transaction with the proper C-19 number is delivered to each of the interested stations.

9.8.3 Posting C-19s

Fluid transfer accounting is a very sensitive issue. Security and verification of transacted data is essential. Operations staff who participated during the system design process identified control of C-19 data as being the dominant concern in any design alternative. They were specifically concerned that they be able to screen incoming C-19 transactions, and accept them, before the C-19 effected the production accounting at their battery.

One of the batteries participating in the pilot averages almost one hundred and fifty fluid transfer transactions each month. The total volumes trucked (transfers in and out) averages over 900 m^3 of oil and over 1700 m^3 of water each month. The receiving operator is at risk of a sending operator over stating the oil content in gross fluid shipped and must therefore carefully monitor transfers for this occurrence. (Over stating the oil content shipped inflates the assumed oil production and thereby the earned revenue of the sending operator's facility while reducing these factors for the receiving operator's facility.) Errors or misstatements in fluid transfers, amounting to 10% of the total volume shipped at this facility, represents revenue of approximately $14,000 (gross Canadian) per month.

Fluid transfer accuracy can also affect the deemed production of a battery and may have adverse effects to the battery proration factors. The battery above has an actual average monthly production of 7,300 m^3 oil and 34,000 m^3 water. A 10% error for fluid shipped results in a deemed production error of 1.2% for oil and 0.5% for water. Government regulations limit total measurement error from all sources to 3% of oil and 5% of water for this battery. Exceeding these limits results in a query from the ERCB. A prolonged period outside these limits could result in severe penalties being imposed, such as shutting in production until the ERCB is satisfied sustained operation within the limit can be reached.

Errors or misstatements in fluid transfers would easily amount to 10% if strict controls and monitoring procedures were not in place.

A scheme was developed to provide for operator acceptance of the information before it would be used for his battery accounting. This special level of control is not required by the well test or group gas transactions. The term "controllable transaction" was coined to describe the general case when a transaction must be accepted before it will be used by a CAO application.

Controllable transactions are processed in CAO similar to any other communicated transaction type. However, a special flag is added to the transaction for each interested party. The transaction will only be used by the application on a given CAO station if the associated flag indicating acceptance has been set.

All controllable transactions, whether accepted locally or not, reside in the application database and are visible on the relevant screens and reports. However, the data contained in the transactions is excluded from all calculations. The screens and reports on which the controllable transactions appear show the status of the acceptance flag.

The act of accepting a controllable transaction is called "posting" the transaction. The term is borrowed from general ledger accounting due to a loose similarity between "posting" an item to the ledger and "accepting" a transaction to the application. An accounting transaction does not appear in the "books" until it has been posted to the ledger. This is similar to the CAO controllable transaction case, so the term "posting" the controllable transaction was coined. The acceptance flag is called the "posted flag".

The posted flag defaults to "Yes" when it is entered at a station, although the operator can override the status. The transaction copy communicated to the network defaults the posted flag to "No" for all other stations having an interest in the controllable transaction.

Similarly, if an edit is received which alters a controllable transaction which has previously been posted at a station, the edit is made and the transaction is "de-posted". This ensures the operator is able to review the result of the edit and reapprove the transaction for his use.

CAO applications provide sorting or selecting only posted or un-posted transactions to permit an easy check by the operator for any outstanding transactions which must be posted.

There are two interested parties in any C-19 transaction: the sending facility and the receiving facility. These batteries may both reside on the same station or different stations. It makes no difference to the application. A C-19 entered by one operator for a given battery is posted for that battery, but remains un-posted for the second battery until that operator reviews and posts the transaction. This is true whether the operators use the same or separate CAO stations to do their bookkeeping.

C-19 Date and the "Received" Date

Another modification required to FDE resulted from the automatic processing of C-19s for total fluids transferred in and out of a battery. It is possible for a fluid shipment to take more than a day for delivery and processing. Thus the shipping date and the receiving date are different in many circumstances.

Before the CAO pilot, Amoco accounting rules stated the receiving battery must book the fluid on the same date as the fluid was shipped from the sending battery. This was acceptable for monthly accounting, but FDE is designed for daily battery balancing for up to date production monitoring. Booking of fluid received on a date other than when it actually was received could cause gross distortions in deemed daily production and mask any real process errors that FDE was designed to illuminate.

To solve this problem, the C-19 transaction was structured with a received date and a shipped date. The shipped date was chosen to be the C-19 date (consistent with the paper form procedure) since the shipping operator has control over this element. A separate field was established for the receiving operator to enter the date the fluid effects his production calculation.

There are two limitations to the dual date arrangement:

1) The received date must be greater than or equal to the shipping date.

2) The received date must fall within the same production accounting month as the shipping date, even if the fluid did not actually arrive or be processed until the following month.

However, neither of these two limitations are a significant concern in practice. The application is designed to enforce these two constraints to prevent any accounting errors.

9.9 Transaction File

The transaction file used in the pilot stores a record for every well test, group gas, and C-19 transaction (entry, edit, or deletion). No other information is stored since these three data types are the only ones included in the pilot EDT process. Table 11 shows the database structure of the transaction file as used in the CAO pilot.

Figure 18 is an example of a transaction report generated for well tests processed on a single day. The report was generated from a sample data set designed to show how the transaction processing detects and handles data collisions.

Database: TRANS.DBF

Field	Field Name	Type	Width	Comment
1	PRG_ID	Character	8	Program name
2	DEVICE_ID	Character	6	Originating SIC
3	BAT_ID	Character	9	Battery identifier
4	WELL_ID	Character	9	Well identifier
5	DATE	Date	8	Book date
6	INPUT_DATE	Date	8	Date last modified
7	INPUT_TIME	Character	8	Time last modified
8	INIT1	Character	6	Initials - creator
9	INIT2	Character	6	Initials - modifier
10	F1_1	Logical	1	Contents per type
11	F2_2	Character	2	"
12	F3_7	Character	7	"
13	F4_7	Character	9	"
14	F5_9	Character	9	"
15	F6_5	Character	5	"
16	F7_1	Logical	1	"
17	F8_1	Character	1	"
18	F9_5	Character	5	"
19	F10_5	Character	5	"
20	F11_1	Character	1	"
21	F12_1	Character	1	"
22	REC_TYPE	Character	3	Record type
23	MESS	Character	40	Description
** Total **			159	

Table 11: Transaction File Structure

Report Date: Aug 9, 1989 | AMOCO CANADA PETROLEUM COMPANY LTD. | Page: 1

From: Aug 7, 1989 To: Aug 8, 1989 | COMPUTER ASSISTED OPERATIONS

TRANSACTIONS REPORT FOR WELL TEST DATA

DATE	TIME	DEVICE	USER1	USER2	BAT_ID	WELL_ID	TEST_DATE	EST	HOURS	OIL	WATER	GAS	PRESS	DESCRIPTION
89.08.08	13:23:14	6H01	NGAS		Bty 1	Well 1	89.08.01	F	24	1.2	82.3	2.31	710	NGAS DATA **
89.08.08	13:23:14	6H01	NGAS		Bty 1	Well 2	89.08.01	F	24	2.5	17.3	1.98	707	NGAS DATA **
89.08.08	13:23:14	6H01	NGAS		Bty 1	Well 1	89.08.01	F	24	1.2	82.3	2.31	710	OLD COLLISION *****
89.08.08	13:23:14		NGAS		Bty 1	Well 1	89.08.01	F	24	2.7	4.2	0.65	1004	NEW COLLISION *****
89.08.08	13:26:58	6H01	NGAS	NGAS	Bty 1	Well 1	89.08.01	F	24	1.2	82.3	0.65	1004	OLD COLLISION *****
89.08.08	13:26:58		NGAS		Bty 1	Well 1	89.08.01	F	24	2.2	22.2	2.2	710	NEW COLLISION *****

Figure 18: Transaction Report Example

Referring to Figure 18, the SDI first processes virgin records for two wells that were tested on August 1. Then a third record is processed with conflicting data for the "Well 1" well test. The transaction report repeats the original record as the "OLD COLLISION" and then shows the new record as the "NEW COLLISION". The two records disagree in all four of the data fields: oil, water, gas, and test vessel pressure. The collision handling processes the record by accepting the new gas and pressure values from NGAS. (It was decided for the pilot that NGAS well test data would not be allowed to overwrite oil and water numbers existing in the database. See Section 9.10)

When a third colliding record is processed for "Well 1" the result of the first collision is reported as "OLD COLLISION" where the revised gas and pressure numbers from the first collision can be seen. Similar processing to the first collision now occurs, where once again the new NGAS gas and vessel pressure numbers are accepted, and the new oil and water numbers are rejected. The transaction report does not display the result which is available from the main data set.

9.10 Collision Handling in the Pilot

The collision selection algorithm for each data type was done in source code for the pilot since this was the most expeditious route for prototype development.

A limited implementation of the selection algorithm was tested in the pilot; the complete implementation was deferred for reasons of cost control, project completion timing, and local concerns. Most data collisions were reconciled by selecting the most recent of the colliding records.

However, a special selection criteria was developed for transactions involving gas volumes and vessel pressures. Well tests and group gas volumes calculated by NGAS (see Sections 9.5 and 9.6) are more accurate than field estimates and are the numbers used for accounting and auditing. Data collisions between NGAS generated gas volumes or vessel pressures, and field estimates, are reconciled in favor of the NGAS value. However, field values for oil and water, which are captured by the operator in the field, are considered to have priority over NGAS values. Therefore collisions for oil and water are reconciled in favor of the field entered data.

For the pilot only, it was decided that NGAS oil and water numbers would be accepted by the host only if the NGAS transaction resulted in a creation of a data record in the host main file. NGAS oil and water values would not be accepted as an edit against the data set, whether in collision or not.

9.11 Communications

As discussed in Section 4.2 the REU can be used as the communication channel to the field work stations. The pilot locations do not have telephones, so communications in the pilot were exclusively through REUs carried between the host and field work stations. REUs are physically connected to work stations by a single communications cable. REUs are connected to the host through a multiplexer.

Each REU was configured with the CAO SIC used for the network addressing purposes. This was done by placing in the REU virtual disk memory a file with a file name containing the SIC of the REU. The file contained a data field which also stored the SIC. For example, the REU configured to have the SIC of 6R39 had a data file in its disk memory named "6R39.DAT" which contained only the four characters "6R39".

The Micro Palm PC/5000 also has a separate addressing scheme which is used by the vendor supplied communication software. This was a simple matter for the pilot software to handle; the SIC of each REU was listed in a data table with the address for the vendor addressing scheme. The software simply used the vendor address scheme to make the communication link. The host also then checked the SIC contained in the REU data file to ensure it had the right CAO station and that it had been configured properly.

The CAO software, installed on each CAO station, had two communications items on the main menu. The first item "Read Data From Hand Held" caused the station to copy the preliminary files from the connected REU. If the reading station was a work station, it would then determine if the connected REU was the designated REU by testing the REU station identity code contained in a special file. If the REU was the designated REU the work station would then read any transit files addressed to it from the host.

The host worked similarly except it communicated through a multiplexer to several REUs in a session. Work stations in the CAO pilot system communicated all network files to the host by appending the network files to the preliminary files of its designated REU. This is consistent with the second approach of REU medium communication identified in Section 4.2. Thus the host only needed to access the preliminary files from each REU.

The second menu item on the main menu was "Send Data To Hand Held". When this item was selected the work station would check to see if the attached REU was the designated REU. If it was not, the work station halted the process and informed the operator that the connected REU was not the designated REU and that no data would be transferred. If the connected REU was the designated REU the work station would transfer the data files to the REU for transport to the host.

When the "Send Data to Hand Held" item was selected at the host, the host would sequentially select each REU which was identified as a designated REU and then transfer the transit files of the associated work station into the REU.

The communication scheme had been designed and developed to work with the original vendor product, and worked in the following manner:

To create a file for download to the REU the processing software first built a file in DBASE format (DBF). This file was then converted to a structured data file with the extension "TXT". The vendor supplied software would read the text file and convert it to a proprietary format for down load to the REU. (This software package was the vendor component which was never satisfactorily completed and lead to the switch in REU hardware.) Reading files from the REUs followed a similar but reverse process. The effect of this design meant Amoco developed software interfaced with the vendor developed software through the passing of text files.

This communication method was re-examined when the decision was made to switch to Micro Palm. The Micro Palm is MS-DOS compatible and can be communicated with using simple MS-DOS commands like COPY. It is desirable to eliminate all the intermediate processing required in the case of the former vendor and simply copy the DBASE format file to the REU. A cost analysis showed it would be more expensive to remove the conversion software (which was built into several modules) than to write a new routine to convert the text file back to a DBASE format before copying to the REU, or to convert back to a text file when copying from an REU. It was decided to stretch the diminishing project budget by building the extra conversion module which became known as the "black box". The system design, although functional, requires unnecessary intervening steps, complexity, and performance degradation. The pilot communicating scheme is to be altered to remove those intervening steps prior to expanding CAO as a production system.

9.12 Logging of Communication Error Events

All communications in the pilot were monitored by the communications handling software for errors. Typical errors trapped and reported include the attempt to access an REU which was not connected and the attempt to download information to a REU which was not the designated REU. A log of these errors is maintained. Figure 19 is an example of a communication error report generated from the communication error log. These errors were deliberately generated for example purposes. Few errors actually occurred in practice.

Referring to Figure 19, the first error shows a situation where the host connected to an REU using the vendor addressing scheme. It then checked to see if the REU had the correct station identity code to verify it was the designated REU for transport of the particular data files to the work station. The error shows it could not find the SIC file which could mean it was the wrong REU or the file did not exist.

The second error shows an attempt to connect to the vendor address associated with the SIC 6R40. The REU with that address did not respond indicating the REU was not connected, was not in communication mode, or had failed somehow.

The following seven errors show a situation where an attempt was made to access a particular REU to read the preliminary files in text format (such as "C-19.TXT") but the REU failed to respond to the connect command, similar to the preceding paragraph.

AMOCO CANADA PETROLEUM COMPANY LTD.
COMPUTER ASSISTED OPERATIONS
COMMUNICATION ERROR REPORT

Page: 1

Report Date: Aug 9, 1989
From: Aug 7, 1989 To: Aug 8, 1989

DATE	TIME	REU ID	ERROR MESSAGE
89.08.07	07:52:47	6R39	* ERROR 02 Send data: 6R39.DAT to 6R39 HHT File Not Found
89.08.07	07:53:25	6R40	* ERROR 03 Send data: 6R40.DAT to 6R40 HHT Not Connected
89.08.07	08:01:54	6R40	Read C-19.TXT: Handheld to Host * ERROR 03 HHT Not Connected
89.08.07	08:02:02	6R40	Read WLTEST.TXT: Handheld to Host * ERROR 03 HHT Not Connected
89.08.07	08:02:10	6R40	Read GAS.TXT: Handheld to Host * ERROR 03 HHT Not Connected
89.08.08	07:25:44	6R39	Read C-19.TXT: Handheld to Host * ERROR 03 HHT Not Connected
89.08.08	07:25:49	6R39	Read WLTEST.TXT: Handheld to Host * ERROR 03 HHT Not Connected
89.08.08	07:25:53	6R39	Read GAS.TXT: Handheld to Host * ERROR 03 HHT Not Connected
89.08.08	07:26:13	6R40	Read C-19.TXT: Handheld to Host * ERROR 03 HHT Not Connected

Figure 19: Communication Error Report Example

9.13 Main Frame Transfer

The work stations report their accounting information to the main frame computer at each month end. This transfer is to an application which is a main frame resident special system. The up load to the main frame is a special case of SDE. The NGAS computer also reports its gas volume and well test information. Any discrepancies between the data received is detected and reported to administrative staff in exception reports.

Section 4.4 described some network timing considerations which could theoretically cause discrepancies between common data records reported by separate processors. There are other sources of discrepancies, including last minute edits done at month end, which are sent to the main frame before the CAO network has a chance of notifying the other effected stations.

The reconciliation of the main frame accounting system on all reported data items traps and reports any deviations between sources for common data items. These are resolved by administrative and operations staff before closing the month end books.

Although the CAO system can itself be a source of errors, the electronic networking does contribute toward reducing the overall amount. Error sources from non network (manual) methods include transcription errors, lost information, mathematical errors, incorrect FLAC assignments, etc. The total of these errors greatly outweighs the errors which may result from theoretical timing or collision handling limitations of the network.

The method for transferring data to the main frame is similar to the facility which existed before the pilot. However, the design of the data transfer interface was performed concurrently with the design of the CAO pilot and reflects the CAO system philosophy. The dual reporting with exceptions feature was installed in the transfer software to accommodate CAO.

10 RESULTS AND EVALUATIONS

> "What concerns me is not the way things are, but rather the way people think things are."
> Epictetus

The CAO pilot operated in the field from mid February 1989 to the end of November 1989. The system was installed in two phases. The first was the installation of the well test transfer from NGAS to the CAO system in February of 1989. The second phase added the group gas transfer and the C-19 networking in August of 1989. The pilot was stopped December 1, 1989 when the FDE system was replaced with POWAR for consistency with the rest of the District.

The pilot was assessed by the development team for performance within the design expectations and by District personnel for impact on operations and staff. This section discusses these aspects of the pilot performance.

10.1 Performance

The pilot system operated as intended with no anomalies or failures that could not be attributed to hardware problems or prototype bugs. Detailed alpha testing along with the field test was performed on each aspect of the CAO pilot.

When the well test transfer was installed operators began to review every electronically received test record with their personal records of well tests performed and the oil and water volumes measured. There were no discrepancies found in any of the data transferred except for a few instances were it was determined that the NGAS operator had keyed in the wrong oil or water value. The network successfully transferred all well tests received for the pilot batteries. The filtering process also worked, as did the transaction file recording. Editing of well tests also worked, but the high accuracy of the electronic transfer meant that very few edits were required in the field. It was therefore difficult to say the editing process was sufficiently field tested. Those edits that were attempted operated properly.

One problem was traced to a software bug: when operators deleted a well test it would reappear at the end of the next system cycle. This was traced to a software implementation error which was fixed.

Another problem occurred when the NGAS vendor altered the AMOCO.XE1 file format without advising the CAO pilot team. This problem was quickly remedied.

The group gas and C-19 system was then installed and it operated with similar success. Problems met were generally traced to sources not involved with the CAO EDT process. For example, District staff had failed to configure the system to include the FLACs of some new entities that exchanged fluid with the pilot battery. This made it difficult for the operators to enter the correct FLAC. In another case, the gas modeling apparently failed when the system improperly calculated the solution gas in the treated oil. This was traced to an improperly entered solution gas coefficient.

Throughout the group gas and C-19 test there were no recorded errors in the CAO networking procedure. From

August to the end of the pilot the system operated primarily as a routine production system and required minimal support from the Calgary office.

One problem did arise when using the REU for data transfer. The Micro Palm showed a sensitivity to static electricity. The CAO host was located in a carpeted, air conditioned room where static electricity was a problem. Even the CAO host (IBM PS/2 Model 80) would occasionally lock up when a static spark occurred when a user touched the keyboard. The REU would also lock up occasionally, requiring a system re-boot. This re-boot sometimes caused loss of data files between the host and work station. Fortunately this problem was controllable by anti static spray on the carpet around the CAO host.

Another problem was met with the Micro Palm that also caused loss of data files. The pilot purchased the first units of a new product upgrade that Micro Palm developed. An apparent flaw in the PC/5000 BIOS (Basic Input Output System) caused the memory where the communication parameters were stored to be overwritten with random settings. The operations staff soon learned to recognize this problem and would reenter the communication parameters before data transfer. Micro Palm also corrected the BIOS problem and repaired all the REUs in the pilot which permanently corrected the problem.

10.2 District Assessment

The District assessment included a review by field and management staff in several functional areas for the three main staff groups in the Pembina District: operations, administration, and engineering. They concluded the CAO system was an asset and was a definite direction for the company to proceed.

Figure 20 is a reproduction of a letter from the Pembina District management to senior Calgary management.

The Pembina District also identified and tabulated several "factors" they observed regarding the use of the CAO technology during the pilot. These factors include observed tangible and intangible benefits, impact to staff and operations, support requirements, and other concerns. The factors were tabulated separately for the operations, administration, and engineering staff. The factors supplied by the Pembina District are reproduced in Tables 12, 13, and 14.

Amoco Canada Petroleum Co. Ltd. 89331ACC0414
Drayton Valley, AB, T0E 0M0
December 5, 1989

D. L. Peterson
S14076, Calgary

Computer Assisted Operations - Pilot Project Recommendation Report; File: JDB - 092.5.14

Attached is the detailed report which culminates the Pembina Production Department's experiences and recommendations for future implementation of this technology.

The Pilot Project is considered to be on the leading edge of computer application to oil field operations. The final recommendation is to continue development of this technology at a medium priority level for use throughout Amoco Canada. We recommend medium priority since operations can function without the technology; however, the time savings, reduced deferred production, improved job satisfaction of our staff, and improved general performance associated with CAO place this technology as a definite direction to proceed for the company.

In order to further develop the technology the following must be accomplished:

- Complete the CAO Pilot prototype design for use in future field computing applications.
- Modify POWAR to add CAO functionality.
- Study the Pembina District to implement the CAO modified POWAR and institute it in the Pembina District.

It is hereby recommended that POWAR, as modified for the now proven CAO technology, be expanded to the entire Pembina District. Pembina would like to participate in the field trials of the new POWAR prior to implementation into the rest of Amoco Canada.

J. D. Bueckert

APF/cv

Figure 20: Pembina Recommendation Letter

Factor	Operations Staff
Time Savings	1. 30 minutes per day per operator.
Data Reliability	1. No mistakes allowed in dates or volumes.
Staff Viewpoint	1. Enthusiastic, even with the start-up project. 2. Job pride and ownership in "their" production data. 3. General computer literacy makes operators and foremen more valuable to themselves, engineers, and administration.
System Complexity	1. Intimidating at first, but not complex. Training was excellent.
System Reliability	1. (Initial) hardware was too limited. Micro Palm solved these problems. Software was adjusted and currently there are no problems.
District Support	1. Support was good when (the PSG District Support person) was in the office, back-up support is needed in his absence.
Calgary Support	1. Good support. 2. Only serious problems are hardware problems. 3. Started out daily communication, then weekly, and now on as-needed basis.
Networking Technology	1. Bring engineering / operations to a more cohesive team on daily production problems. 2. Faster response to production downtime will reduce deferred production.

Table 12: Pilot Factors For Operations

Factor	Administration
Time Savings	1. Print time, 15 minutes per day.
Data Reliability	1. Improved accuracy by (reducing data errors by) 50% to 70%.
Staff Viewpoint	1. Very positive. 2. Handling data only once.
System Complexity	1. Not complex at all.
System Reliability	1. NGAS changes have to be adjusted for CAO.
District Support	1. Production data support with minimal time required. 2. Training was major part, initially 10-20% of the support representatives time. Now requires about 2% of the support representatives time for ongoing support.
Calgary Support	(Same as operations comments.)
Networking Technology	1. More timely information at month end can save up to 24 hours for some areas.

Table 13: Pilot Factors For Administration

Factor	Engineering
Time Savings	1. 5 hours at month-end for manual comparison with (printed) reports.
Staff Viewpoint	1. Engineering gets more appreciation for administration and operations staff.
Networking Technology	1. Engineering will get more consistent information. 2. Networking has big potential in future. 3. Linked with (other CAO applications) will become very important and the most current database for engineers.

Table 14: Pilot Factors For Engineering

10.3 Field Acceptance

The CAO system was very well received by the field staff on the pilot. They were patient through the inevitable bugs that must be solved in a prototype system and were enthusiastic with the results and their participation.

The field participants were asked to provide written testimonials of their experiences. The field staff sometimes use the term "CAO" and "the hand helds" as synonyms since the hand held component was the only physical item introduced to their FDE system and is therefore the most visible change. The following excerpts are quoted from the written testimonial provided by the area field foreman:

> "My opinion (is) problems which were encountered during the pilot of the program were to be expected and were dealt with accordingly."
>
> "... many benefits can be realized from the use of these hand held computers (i.e., daily tank gauges and meter readings, along with other field information being done by our field operators on paper that could be handled by CAO)."
>
> "I believe these hand held computers certainly do have good cost saving and time saving capabilities, given proper training so any of our personnel can use them."

The following excerpts were quoted from the testimonials provided by the five involved operators:

> "I understand as of December 1, 1989 the ... CAO system will be discontinued due to (the conversion to) the POWAR program. It is unfortunate that there is no software to continue the CAO system with the POWAR program. It has saved valuable time".
>
> "I believe the hand held has many additional options: tank gauging, pipeline readings, ... water readings, mileage reports, etc."
>
> "simple to learn"
>
> "I found the CAO pilot to be very successful. As a result, when the project was in full swing, 3-4 hours per week were saved."
>
> "Very useful (that) the data only had to be input into the ... computer in town, down loaded into the hand held and loaded up into our PCs; a very accurate system. When every entry has to be (manually) input into the PC, there is much room for error."

11 CONCLUSIONS

Specific conclusions supported by pilot results are:

1) The theoretical CAO methods have been proven in a practical environment.

2) Electronic networking has added value to an existing field computing application and suggests improved economics for proposed systems.

3) Operator work load has been reduced by electronic transfer of data and elimination of redundant data entry.

4) CAO techniques have improved the timeliness, accessibility, and accuracy of data to field and office staff.

5) All staff have access to more consistent information.

6) Field acceptance of the new methods has been very favorable.

7) Electronic forwarding of data from the field computers has substantially reduced data entry time by clerical and administrative staff.

The CAO approach has been determined practical and beneficial to field operations in a production monitoring application. The CAO pilot has been highly successful in the applications tested.

12 FUTURE DEVELOPMENTS

> "I've never been satisfied with anything we've ever built. I've felt that dissatisfaction is the basis of progress. When we have become satisfied in business we become obsolete."
>
> J. Willard Marriott, Sr.

The CAO pilot culminates the testing of the CAO theory. However, CAO did not end with the conclusion of the pilot. This chapter discusses projects currently under development or planned for the near future. Additional brief discussion of enhanced field data communications is also provided.

POWAR

As mentioned in Section 2.2, CAO data transfer techniques will be phased into POWAR. However, POWAR already has some of the advances developed in this thesis; transaction number generation, transfer to main frame, and posting have been implemented for C-19s. POWAR has also been modified to work with the CAO Communications facility (discussed following). Although not yet approved, there are plans to develop the link between SCADA systems and POWAR for deployment by early 1991.

CAO Communications Project

CAO Communications is a project designed to handle the various communications media that are commonly found in the field. As discussed in Section 4.1, CAO applications are not burdened with the details of the communication media. It is only necessary for the CAO applications to populate and process the special classes of CAO files identified in Section 5.1.

CAO Communications is an application designed to standardize the data repositories in the main storage of each CAO station. It also manages the communication of the special CAO files between stations.

Several CAO features discussed in the thesis are included in CAO Communications: remote-controlled communications mode (Section 4), data compression and mail files (Section 4.5), and logging of communication events (Section 4.6).

The first release of CAO Communications includes handling of diskette, modem, and coaxial communications between PCs and between PCs and the main frame. Modem communications include regular telephone, aurora, and cellular capabilities. A future revision of CAO Communications will include the PC to REU communications for the SWELL project (discussed below). An investigation into the practicality of using Local Area Wireless Network (LAWN) technology for short distance field communications is also underway.

CAO Communications addresses the problem of support logistics in field computing. It uses a commercial software package for remote control of PCs by a support person. Previously, if a field user had a problem a support person would try to talk the user through the solution on the phone. If this did not work, the support person would either travel to the location or request a back up of the database be mailed in. Either method resulted in unacceptable support delays. Using CAO Communications the support person can link his PC to the troubled PC using a modem connection and take control of it until the problem is solved.

SWELL

SWELL is a production monitoring program similar to POWAR but optimized for single well batteries. (POWAR is optimized for the multi-well battery case.) SWELL is an approved project and is currently under development.

SWELL uses several advances of this thesis. SWELL will use REUs for portable data capture and decision support. SWELL provides a refresh of the main files in the REU by the CAO host which is an improvement over the pilot method. This increases the amount of data available to the operator on site.

SWELL also will have extensive C-19 processing and networking. One improvement over the pilot is SWELL's use of SDI (see Section 5.5) to import fluid transfer data from a truck weigh scale system to the CAO host. The host will then automatically generate C-19s for distribution to the operator's REUs. SWELL also adds a "C-19 verify" procedure for confirmation by the operator that the C-19 matches paper documentation left by the trucking company that hauled the transfer. C-19s which have not been verified within two days of generation are printed in an exception report to alert field staff.

SWELL will use CAO Communications for PC to REU, PC to PC, and PC to main frame communication.

Enhanced Mobile Field Communications

Mobile communications will allow greater flexibility and convenience for CAO. One method already proposed would exploit excess voice radio capacity within Amoco Canada's operations for a province wide field data network. Packet radio modems are one potential method, as are data modems designed to piggyback data transfer on voice channels.

Local Area Wireless Network (LAWN) technology with fairly long range capability (twenty miles) has recently become commercially available in Alberta. The cost is too excessive for immediate use ($4,000 per station) but is expected to drop. Data rates of 19,200 bits per second are possible.

Recent commercial advances in satellite communications allow for nearly continuous two-way data communications between mobile earth stations and a central office. Data throughputs are currently too slow for CAO use (approximately 100 bits per second) but the future potential looks very promising.[10] Similar technology has also been made available for portable satellite earth stations which have sufficient bandwidth to support voice communications.[11]

BIBLIOGRAPHY

1. G. P. Hiltz, Computer Assisted Operations - A Practical Approach to Networking Field Computers, The Petroleum Society of CIM, Paper No. 89-40-62, 1989; and The Society of Petroleum Engineers, Paper 19137, 1989.

2. B. L. Giles, and S. W. Wells, Use of Computers in the Operation of Ekofisk Complex, Society of Petroleum Engineers, Paper 14122, 1986.

3. G. R. Gill, Amoco's Field Data Entry System, The Petroleum Society of CIM, Paper No. 88-39-91, 1988.

4. D. Johnson, FDE, A Look into Pembina District's Field Data Entry System, Canadian Horizons magazine, Summer 1988.

5. IEEE Standard Dictionary of Electrical and Electronics Terms, An American National Standards, Approved July 20, 1984. ANSI/IEEE Standard 100-1984.

6. Micro Palm Technical Reference Manual, First Edition, Micro Palm Computers, Inc., Document MC10034, June 1988.

7. C. J. Glynn, Results of Remote Entry Units / Personal Computer Pilot Test, Society of Petroleum Engineers, Paper 17797, 1988.

8. J. B. Moore, Oilfield Surveillance With Personal Computers, Journal of Petroleum Technology, June 1986.

9. T. A Bettis and S. W. Shreve, The Application of Hand-Held Computers in Gathering Field Data, Society of Petroleum Engineers, Paper 16876, 1987.

10. TMI Brings Mobile Communications into the Space Age, Keep In Touch, Telesat Mobile Inc., Summer 1990.

11. FieldKIT Brings Remote Operations Closer to Home, Keep In Touch, Telesat Mobile Inc., Summer 1990.

APPENDIX A: GLOSSARY

Actual Production

The known hydrocarbon quantity taken from a reservoir through the given surface facilities; measured by sales, fluid transfers, calculated vapor losses, inventory changes, and adjusted for fluids consumed locally or injected during work over procedures.

Architecture

See "System Architecture".

Area Office

A company office housing field supervisory (foremen) personnel; may also include field administrative staff, field engineering staff, and district or field management.

Battery

A field facility where custody transfer of the basic hydrocarbon production commodities (oil and gas) usually occurs. Batteries typically have the process facilities to treat the crude hydrocarbons to remove basic sediments and water, along with other foreign or undesirable substances (e.g. H_2S) to bring the oil and gas components to pipeline sales specification.

Book Date

The book date is the date a transaction becomes effective for accounting purposes. The book date is also known as the "accounting date".

Business Affiliate

An Amoco term denoting a definable business contact with an interest in a transaction. For example, another oil company which ships crude hydrocarbon material to Amoco for treatment to sales condition.

C-19

An accounting form unique to Amoco used to track the shipment of oil and water between field facilities.

CAO Networking

CAO Networking refers to the process of exchanging information transactions through the CAO framework, to a specifically addressed station or stations.

CAO Network

The CAO network is a system of computer processors and databases, connected by the occasional use of communication channels, and capable of communicating transactions to a specifically addressed stations on the network.

Chart Date

The date that a gas or well test chart began recording process data (gas volumes) in the field.

Collection File

A special class of file unique to the CAO host. All transactions from incoming network files, SDI transfers, and keyboard edits at the host, are processed against this file. There is one collection file for each type of transaction record processed. The host continues to accumulate transactions into the file until it processes the file into transit files for network distribution; the collection file is then emptied. The collection file is never communicated directly to any other station.

Computer Assisted Operations (CAO)

A **framework** which provides for multi-directional **flow** of information necessary to sustain effective **decision support** in the work place of field employees. CAO is the computing architecture where applications running on separate field processors can communicate and share data in a practical, integrated manner. CAO provides for transaction processing, across distributed and disconnected databases, with overlapping data sets, operating asynchronously. CAO operates as a distributed database, in a computing environment where communications between processors is rare and not scheduled.

Controllable Transaction

A transaction communicated on the CAO network which must be "posted" in the receiving application before it will be included in the application analysis. The transaction resides in the main database of the recipient station, but is not included in the application analysis until posted.

Cycle Time

The clock time required for CAO network consisting of one host, and all the work stations and REUs associated with that host, to complete a system cycle. Cycle time is usually 24 hours.

Data Acquisition

The process of acquiring data which would otherwise never be available. Usually refers to field data, and is sometimes called "data capture".

Data Collision

In the CAO network there is data which is common between a work station and a host, and sometimes between several work stations and hosts. This common data can be entered or modified at any PC site. When the same data is entered or modified at different sites during the same system cycle, so the entered values disagree, a "data collision" is said to have occurred. Specifically, a data collision is the attempt to process multiple transactions of identical key fields and differing non-blank data fields in the same system cycle.

Delete Flag

A special field included in each type of communicated transaction which signals to applicable stations on the network that the transaction is to be deleted from the data set.

Derived Data

Information which can be calculated from independent data in the data set is called "derived data"; sometimes called "calculated data". In most cases, derived data is not communicated as a transaction on the network, since it is assumed the data can be easily recalculated at each CAO station from the independent data which is communicated.

Designated REU

An REU associated with a particular work station, which is used as the communications medium between that work station and its host.

Device Type Code

A single character imbedded in a Station Identity Code which identifies what type of device the station is; i.e. "H" for "Host", "W" for "Work Station", "S" for "SCADA", etc.

District Code

A single digit imbedded in a Station Identity Code which signifies which District the CAO station resides in.

District Foreman

A person who supervises the operations staff responsible for one or more areas of operation. Usually a District Foreman has several Field Foreman reporting directly to him.

District

A management jurisdiction within Amoco Canada, which directly controls the day to day operation of several area offices.

Electronic Data Transfer (EDT)

EDT is the communication of data in the broadest sense within CAO and associated special systems. EDT is the data transfer function of the CAO framework, including the transaction processing, networking, SDI, and SDE, operating throughout the CAO hierarchy.

EM-Reports

A special class of reports which must be provided, by regulation, to the Saskatchewan Government. These reports track the production, inventory, and disposition of hydrocarbon production in the province.

Energy Resources Conversation Board (ERCB)

The ERCB is the operating arm of the Alberta Government responsible for enforcing government policy and regulations in the energy industry. This includes collection of royalty revenues, resource management (maximizing long term recovery), and monitoring policy compliance.

Entry Date

The date of origin of a data record.

Entry Time

The time of day in which a data record originated.

External Data

CAO allows entry of data into any processor on the network which runs an application which captures that type of information. This is true even if the capturing processor does not require the particular data entity in question (although it is assumed to be required elsewhere in the network). Data captured at a CAO station, which has no local relevancy, but which must be communicated to the network, is called "external" data.

Field Data Entry (FDE) System

A PC based field battery accounting program, developed by Amoco Canada before the Dome - Amoco merger, for use by operator's at their field facilities. FDE was modified for the CAO framework and was the application used in the CAO pilot.

Field Foreman

A person who directly supervises the activities of operations personnel working in a specific location or locations.

Field Staff

The term "field staff" can mean any of a wide range of job functions. In CAO, "field staff" are specifically defined as operational staff of the Production Department; and including the production, maintenance, diagnostic, and construction staff responsible for the day to day operation of oil and gas production facilities and related equipment.

Fluid Transfer

A physical transportation of production fluids which requires a record for accounting purposes. Usually a fluid transfer involves trucking of production fluids from one field battery to another. Fluid transfers are sometimes called "product movements". See also "C-19" and "Truck Ticket".

Four Unit Rack

A device to multiplex up to four REUs to a CAO host. Up to ten racks may be connected allowing forty REUs to be multiplexed to a single serial port on a PC.

Free REU

An REU which is not assigned as a communication medium between a host and a work station.

Functional Location Accountability Code (FLAC)

The FLAC numbers are a type of accounting code unique to Amoco, that is used to identify certain chargeable locations or information entities for which data is to be accumulated. These locations or entities include wells, lease/units, non-interest properties, royalties, facility/plants, and administrative locations. For example, a single producing oil well has a FLAC assigned to it. The basic FLAC is a six digit code, which is paired with a 2 digit "ownership code" or "registered horizon code" to uniquely identify the location or entity. A check digit, calculated on the full eight digits (basic and ownership or reg-horizon code) is generally added. FLAC numbers are assigned to simplify computer processing of business information.

Gas Gathering System

A field facility which receives hydrocarbon gases, and does dehydration and compression processes only. No other products are removed from the incoming hydrocarbon stream.

Gas Plant

A field facility which receives hydrocarbon gases, usually from producing oil and gas wells, and processes them to extract at least one product other than water from the incoming stream. For example, a gas plant may extract ethane used in a nearby miscible flood, or it may extract sulphur for sale from sour gas production.

Hand Held Computer

A digital electronic device, similar to a personal computer, but smaller and of less weight so that it can be carried in one hand by a person to perform routine daily activities.

Hierarchy

The CAO structure where components are ranked into levels of subordination according to function and locality of domain.

Host

A computer station in the CAO hierarchy which provides primary data routing for work stations and REUs in the local CAO network. The host database is a consolidation of data from all work stations and REUs beneath it in the hierarchy.

Independent Data

Independent data is the lowest level of basic information which is generally captured at the field. Independent data items cannot be calculated from knowledge of other independent data items; sometimes called "raw data" or "source data".

Key

A key is a fundamental data item in a record structure that must be present. A key identifies the relationship of the record contents with a specific data or facility entity, time, location, person, or processing method.

Lease Fuel

Natural gas consumed on site for process energy. Sometimes called "Lease Use" or "Fuel Gas".

Mail File

A mail file is a collection of transit or network files addressed for a specific station. The mail file is typically generated using a file compression program.

Main Files

The main files are the local database files which are directly accessed by an application for its normal processing.

Meter Modeling

The means by which measured volumes are computed in arithmetic sums and differences so that the result is a valid accounting quantity; e.g. net sales of gas may equal measured sales less lease fuel buy back.

Meter Run Number

A seven digit number which uniquely identifies every gas and liquid meter used by Amoco Canada for accounting purposes; sometimes called a "measurement point". (During the CAO pilot, which was initiated before the Dome - Amoco merger of 1988, the meter run number had only 4 digits.)

Modification Date

The date on which a record in the CAO system was last modified.

Month End

A term generally denoting the end of the monthly accounting cycle, when production, sales, inventory volumes, and revenue streams are adjusted.

Natural Gas Accounting System (NGAS)

A commercially available program which provides for integration of paper charts from orifice type gas meters, along with full database storage and accounting procedures. In the CAO pilot, the NGAS computer passes gas volume information to the CAO host through an SDI.

Network Control Files

A special class of files which contain information necessary to control network data transfer, e.g. a list of Station Identity Codes, and the method of communicating to the station.

Network File

A special class of file generated by a CAO station which stores all new or modified transactions of a particular type for communication to a host for transaction processing. For example, a work station may develop a network file of newly created or modified well test records; a second network file for newly created or modified C-19 records; etc. Usually, a network file moves up the hierarchy to become an input to a transaction processing event.

Network Routing

The CAO method whereby transaction key fields are related to a list of Station Identity Codes to determine which stations in the network require notification of the transaction.

Network Transaction

A network transaction is a routine CAO transaction which contains all the data and appropriate keys for EDT processing within CAO.

Operating Run

A group of closely spaced production facilities routinely operated by a one person shift.

Operations Personnel

Amoco staff and contract employees directly charged with operating facilities and maximizing production in a clearly defined operating area. This includes production, maintenance, diagnostic, and construction personnel and their direct supervisors. Operations Personnel are sometimes called "Field Staff".

PC

A Personal Computer.

Pilot Project

An operational test of a theory, procedure, or system. In CAO, the pilot refers to the field trial where the CAO concepts were tested in actual field practice.

Posting

The act of marking a controllable transaction (such as a C-19) to be included in an application analysis. The purpose of posting is to give operators some control over data transactions received from other stations on the network. The operator must view and post each controllable transaction before it is included in the local station analysis. See also "Controllable Transaction".

Preliminary File

The data files residing in an REU which contains newly captured (or modified) data. Preliminary files are used by the REU both as a main file and as the file to be communicated to the network.

Production Data Resource (PDR)

The PDR is the highest level in the CAO hierarchy, referring to those functions residing on the centralized main frame system. Although the concept is defined within CAO, the PDR is not developed in the CAO pilot forming this thesis.

Production Operators Work station And Reporting (POWAR) System

A second generation program replacing FDE as the Amoco application of choice for field battery accounting.

Proration Factors

Government regulations and partnership agreements stipulate actual battery and well production must be balanced against theoretical production volumes, as calculated from a sampling measurement process and actual operating hours of the constituent producing wells. In multi-well batteries, deviations between theoretical and actual production are applied back to the individual producing wells on a pro rata basis. A factor for oil, water, and gas is generated for the battery and is used to weight the theoretical production for each well to arrive at its "prorated" or allocated production.

Record Matching

Record matching is a process designed to match incoming network records with temporary records contained in the station. The network records are usually paired with temporary records by relating keys in the record other than the main key. The operator must approve each match. Once matched, the temporary record is erased and replaced with the network transaction.

Remote Entry Unit (REU)

The lowest level in the CAO hierarchy, characterized by its level of function. The REU functions as a data capture unit with limited processing power. It is a highly focussed station type, designed to capture the data handled by a specific operating run or job assignment. REUs are typically hand held or lap top computers.

Response Time

The amount of processing time elapsed between operator initiation of a process and completion of a result. In CAO, applications work with the local database; queries cannot be made to the system at large. Hence response time includes access time to obtain local records and processing time to answer the query, and does not include network communication time.

S-Reports

A special class of reports which must be provided, by regulation, to the Alberta Government Energy Resources Conservation Board. These reports track the production, inventory, and disposition of hydrocarbon production in the province.

Selection Algorithm

The process by which colliding data entities are screened to select the value which has the highest likelihood of being correct. Selection criteria is predefined for each application data type. The criteria may involve creation date, sequence of arrival at host, some predetermined "ownership" of the data, a prioritized list of source reliability, etc.

Special Data Export (SDE)

The bulk transfer of operational data from the CAO network to special systems (including SCADA) is called a "special data export" (SDE).

Special Data Historical File

A term in CAO to describe the repository where special data is stored within the CAO host computer. The file is used to screen each incoming data set to remove data already received from a prior session. The goal is to reduce redundant handling of the same information. Since historical files are used to screen out redundant data, they are sometimes called "filter files". In the CAO pilot, there is a special data historical file to screen incoming NGAS data.

Special Data Import (SDI)

The bulk transfer of operational data from special systems (including SCADA) to the CAO network is called a "special data import" (SDI). The CAO host then processes this data to convert it to CAO framework requirements, and then includes the information in its regular transaction processing procedures.

Special System

The CAO term "special system" is reserved for non CAO programs or computer systems, with which it is desirable for the CAO network to transfer data. Examples include specialized accounting and technical systems. See Special Data Import and Special Data Export.

Static Data

Information describing a facility, operating run, key fields, or other relatively permanent arrangement is called "static data". It is sometimes called "configuration data". In most cases, static data is not communicated over the CAO network due to the infrequency with which it is altered.

Station Identity Code (SIC)

A four character code used to uniquely identify each station in a CAO network.

Station

A uniquely identifiable processor which oversees a known functional domain in the CAO network. It is possible, although not common, for more than one virtual station to reside on the same processor.

Supervisory Control And Data Acquisition (SCADA)

A system using telemetry and digital devices to provide data gathering and selective control of remotely located apparatus, using multiplexing techniques over a relatively small number of interconnecting channels. SCADA is sometimes called "automation" or "telecontrol" in the oil industry.

System Architecture

The structure and relationship among the components of the CAO system.

System Cycle

Operation of the CAO network can typically be characterized as a cycle which includes: 1) Each station reports its new or modified data to the network by forwarding it to the host. 2) The CAO host processes all new and modified data and addresses updates required at each station. 3) Each station receives and assimilates the new and modified data. The completion of the third step generally signals the end of one system cycle and the beginning of another. A system cycle usually corresponds to a daily accounting cycle.

Telemetry

Telemetry is the transmission of measurable quantities using telecommunication techniques. Typical telemeter types include current, frequency, pulse, ratio (phase) and voltage as a means of translating the measured value to a telemeter quantity.

Temporary Record

Temporary records are used when operators wish to enter information to his station for immediate processing, without waiting for the network transaction to arrive at a later time. Temporary records are used for entry of data known to be redundant to a more accurate data source elsewhere in the system, but which cannot determine absolutely the unique transaction key which will be applied at the other CAO station. Thus the local station can use the data immediately without waiting for network communication. Temporary records are **never** communicated to the network.

Theoretical Production

The expected production from a well or facility based on sampled measurements, known adjustments, and actual operating times.

Timing Exception

A timing exception is the attempt to process a transaction against a data set which contains a matching transaction (by identifier keys), but dated more recent than the incoming transaction.

Transaction File

A file used within the CAO system to store in each station a record of all transaction records passing to or from a station on the CAO network. In particular, data collisions are recorded so the two colliding records and the result (after algorithmic selection) are stored. It could also be called a "Transaction Log File".

Transaction Processing

In general, transaction processing refers to the working of transactions through a definable process in a centralized database. In CAO, users deal directly with a local database which later communicates all new or changed data to a network. CAO transaction processing includes the necessary data transfer control, collision checking, and updating procedures to reconcile data transactions on the distributed, disconnected databases of the CAO stations.

Transaction Record

A data record which contains the smallest definable piece of information still relevant to a business need, which is worked through the system to the appropriate user station(s) and application(s).

Transaction

An action taken against a data set. In CAO, a "transaction" implies the existence of a specific transaction record which is worked against the data set.

Transit File

A temporary file, addressed for a specific station, which is communicated across the network to the station where its contents are absorbed and the file is eliminated. A transit file is generated by a host as a function of its batch transaction processing, and is communicated to the network in the predefined means applicable for the addressed station. A transit file is an output from a transaction processing event, and moves across or down the hierarchy to an addressed station. The host creates one transit file for each type of transaction record which must be communicated to a specific station. For example, the host will create a transit file for well test transaction records, another for C-19 transaction records, etc., for each station on the network.

Truck Ticket

A paper form supplied by a trucking service company to account for fluid transfers between two locations. The ticket usually contains the name of the trucking firm, the date, the quantity of fluid (oil and water), and the source and destination identifiers. The truck ticket also serves as an instrument of conformance with the Hazardous Goods Act by documenting fluid type, and who to contact, should a mishap occur. Truck tickets are also the manifest for the trucking company, authorizing them to haul fluid belonging to Amoco or a Business Affiliate.

Well Test

A sampled measurement of a well's production volumes over a given time period. These sampled rates are assumed constant until the next sampled test in what is known as the "test to test" production calculation.

Work Station

The next to lowest level in the CAO hierarchy of stations. A work station is intended for local data storage and application processing of the field relevant programs.

Index