AUTOMATED MAINFRAME DATA COLLECTION IN A NETWORK ENVIRONMENT

David L. Gross
Computer Engineer
Analex Space Systems, Inc.
NASA Kennedy Space Center
Post Office Box 21206
Kennedy Space Center, FL 32815-0206
Phone: 407/861-5716
Fax: 407/861-5774

ABSTRACT

The progress and direction of the computer industry have resulted in widespread use of dissimilar and incompatible mainframe data systems. Data collection from these multiple systems is a labor intensive task. In the past, data collection has been restricted to the efforts of personnel specially trained on each system. Information is one of the most important resources an organization has. Any improvement in an organization's ability to access and manage that information provides a competitive advantage. This problem of data collection is compounded at NASA sites by multi-center and contractor operations. The Centralized Automated Data Retrieval System (CADRS) is designed to provide a common interface that would permit data access, query, and retrieval from multiple contractor and NASA systems. The methods developed for CADRS have a strong commercial potential in that they would be applicable for any industry that needs inter-department, inter-company, or inter-agency data communications. The widespread use of multi-system data networks, that combine older legacy systems with newer decentralized networks, has made data retrieval a critical problem for information dependent industries. Implementing the technology discussed in this paper would reduce operational expenses and improve data collection on these composite data systems.

INTRODUCTION

The need to access and retrieve data from mainframe systems is a widespread labor intensive activity. A number of commercial products based on the client/server concept are available to solve this problem. In a client/server system the "client" portion of the applications reside on workstations or Local Area Networks (LAN) with the "server" portion running on larger machines (i.e. mainframes). Economically the cost of purchasing, installing, and maintaining such products on one or more systems can outweigh the savings in manhours. These systems do save time in data retrieval and system access but they require a significant initial investment in additional training, equipment, and software development tools. The cost and time required for data retrievals increase geometrically when multiple, usually dissimilar systems are integrated. Tying different systems together means connecting incompatible architectures, protocols and languages. This paper discusses a composite system that can perform many of the same retrieval functions of a client/server system but without the technical restrictions and financial overhead involved.

Kostoff, R. N. "Co-Word Analysis for the Evaluation of R&D," in *Assessing R&D Impacts: Method and Practice*, B. Bozeman and J. Melkers, eds. Kluwer Academic Publishers, Norwell, MA, 1993a.

------. "Database Tomography: Origins and Applications," *Competitive Intelligence Review*, Special Issue on Technology, 1993b, to be published.

------. "Database Tomography for Technical Intelligence," *Competitive Intelligence Review*, Volume 4, Number 1, Spring 1993c.

------. "Database Tomography: Origins And Applications," *Competitive Intelligence Review*, Special Issue on Technology, Volume 5, Number 1, Spring 1994, Wiley & Sons.

NAS. "Reorientation of the Research Capability of the Former Soviet Union," Results of a Workshop on 3 March 1992, NAS, NAE, IOM, National Academy Press, Washington, D.C., 1992.

AUTHOR IDENTIFICATION

Ronald Neil Kostoff received a Ph.D. in Aerospace and Mechanical Sciences from Princeton University in 1967. At Bell Labs, he performed technical studies in support of the Office of Manned Space Flight, and economic and financial studies in support of AT&T Headquarters. At the U.S. Department of Energy, he managed the Nuclear Applied Technology Development Division, the Fusion Systems Studies Program, and the Advanced Technology Program. At the Office of Naval Research, he is Director of Technical Assessment, and his present interests revolve around improved methods to assess the impact of research.

Henry Joseph Eberhart holds a Masters degree in Mechanical Engineering from the University of Southern California and a Bachelors degree in Engineering from the University of California, Los Angeles. His entire career has been with the Navy at China Lake, California and has involved materials, stress analysis, and test and evaluation of air-to-surface missile systems. He is presently engaged with the planning functions at what is now the Naval Air Warfare Center Weapons Division China Lake.

BACKGROUND

In 1990 NASA funded a project to improve the data retrieval and dissemination methods used by the Safety, Reliability & Quality Assurance (SR&QA) Directorate. The methods currently being used require highly skilled data researchers to access and query over 27 different mainframe systems. Data requests could take from a few minutes to a few days to complete depending on the number and types of systems involved. The goal of the project was to develop a more time-efficient method for performing these retrievals. Several commercial client/server systems were evaluated, but from a technical or cost perspective, they were unacceptable. The decision was made to develop a new method of data retrieval.

The Centralized Automated Data Retrieval System (CADRS) is a result of this project. CADRS is a network based system that automatically handles all system accesses, queries, data conversions, and transmittals from the mainframe to the PC environment. This implementation required the development of three sub-systems. These sub-systems are the Central Document Database (CDD), Automated Reporting System (ARS), and Forms Query (FQ).

The three subsystems work in tandem to fulfill all of the data handling requirements of the SRM&QA group. The Central Document Database acts as the primary user interface to all general data reports and supporting technical documentation. The updating of this information as well as single user event driven reports are handled by the Automated Reporting System (ARS). The Forms Query system performs all ad hoc (one time only) data searches and report generations.

CENTRAL DOCUMENT DATABASE

The CDD is a repository for all technical documents and reports that require easy access and full search capability. It provides the advanced document handling techniques and specialized search techniques required by the user community. This is a dynamic system that contains a library of technical documents and host data reports that can be accessed through a Local Area Network (see Figure 1). The system is designed to access documents and delimited data files of various sizes and types that are stored at different physical locations and provides a single interface for viewing and searching of this data.

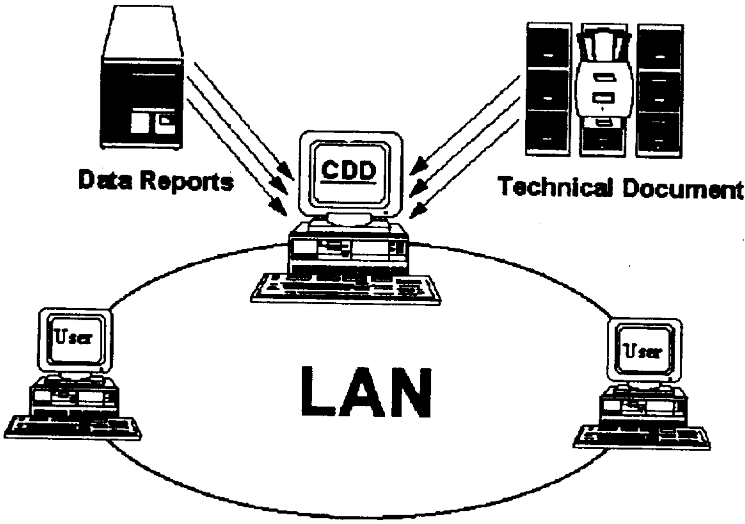


Figure 1 Data Flow of the Central Document Database

A basic menu system is used to call up and display all of the available documents and reports. The system uses a number of filters that enable it to access documents in common word processors and mainframe printer formats. These are simple filters designed to mask the specialized command codes used by the different application programs that produced the document. As a result a document in almost any format, (e.g. Word Perfect, Displaywrite, or mainframe redirected printer output), can be accessed by the system and displayed to the user.

The CDD has the capability to perform several different types and levels of data searches. The simplest search is a basic Boolean key-word search. This type of search is a useful and fairly common type of search that can locate a specific string using standard AND/OR logic. The program provides an improvement to this type of search by expanding the Boolean logic to include any acronyms and abbreviations of the user's search request from its built-in knowledge database.

The most complex search the program performs is based on a natural language parsing and weighting network. This network can identify and rank the key areas of the document that are most likely to contain the requested information. The program syntactically breaks down the user's data request and converts it to a network of related search words (see Figure 2). The document(s) being searched are then compared word by word to this network. A weight value is assigned to each word or phrase in the network. The sum of the weights for each word found in a section determines the overall value for that section. A list of pointers, to the sections of the document that had the highest values, is the final result of the filtering. The software will immediately display the area of the document that had the highest weighting. If the user does not find the desired information, he can move to the next highest weighted area.

Technical documents accessed through the CDD are initially stored by direct scanning using optical character recognition software or by downloading an existing online version. Most online documents are "dynamic" in that they are constantly being updated and revised. To ensure that the most up-to-date version of a document is available, the CDD interfaces with and uses the services of, the Automated Reporting System (ARS). The ARS automatically transmits the latest revisions of the documents to the CDD. This ensures that the CDD has the latest version of all stored technical documents as well as the latest data reports.

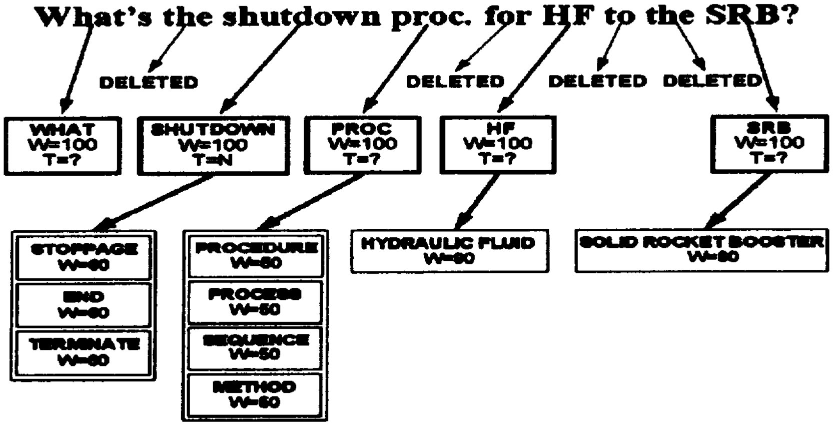


Figure 2 Development of the Weighting Network

AUTOMATED REPORTING SYSTEM

The ARS is an autonomous mainframe based set of tools and techniques designed to automatically query and transmit time or event driven data requests. The system runs on several major NASA host systems. It uses a collection of programs, queries, and scripts to collect data and transmit it to the user's Local Area Network (LAN). The host programs act as "initiators" and run as scheduled batch operations. These programs are automatically initiated on a daily, weekly, or "per shuttle flow" basis during non-peak periods of the day. This minimizes the system's impact on overtaxed mainframe resources. Below is an example of an "initiator" program written in the REXX (Restructured Extended Executor) interpreter language.

```
'SQLINIT D(KSC1H0P3'
'CP LINK SYSP01 19A RR'        /* PRODUCTION COMMON DISK    */
'ACC 19A F'
'CP LINK SQLDBA 195 RR'         /* SQL/DS                  */
'ACC 195 G'
'CP LINK ISPVM 192 193 RR'        /* ISPF & ISPF/PDF          */
'ACC 193 H'
'CP LINK MAINT 19D RR'         /* HELP                    */
'ACC 19D K'
'CP LINK MAINT 303 RR'         /* GDDM - GDDM/PGF - GDDM/GKS */
'CP LINK MAINT 303 RR'         /* GDDM - GDDM/PGF - GDDM/GKS */
'ACC 303 L'
'CP LINK SYSADMIN 399 RR'       /* PROFS                   */
'ACC 399 M'
'CP LINK MAINT 347 RR'         /* QMF                     */
'ACC 347 N'
THISDATE = DATE('S')
W = DATE('B')//7
Y = SUBSTR(THISDATE,1,4)
M = SUBSTR(THISDATE,5,2)
D = SUBSTR(THISDATE,7,2)
NDAY = '31 28 31 30 31 30 31 30 31 30 31'
IF W = 0 THEN,                 /* CURRENT DAY IS MONDAY */
  COUNT = 3
ELSE
  COUNT = 1
DO A = 1 TO COUNT
  D = D - 1
  IF D = 0 THEN DO
    M = M - 1
    IF M = 0 THEN DO
      M = '12'
      Y = Y - 1
    END
    D = WORD(NDAY,M)
  END
END
YESTERDAY = Y||'-'||M||'-'||D
SAY YESTERDAY
'REPORTER AR005 (US <CTRID>/<CTRID> DISK AR005 TEXT A PARML ' YESTERDAY
'SF AR005 TEXT A TO CDD1 AT RQVMKSC'
```

This program is designed to be invoked by VMSchedual running on a IBM 3090 every weekday before first shift. The program performs all required links, identifies the previous weekday date and passes the date as a parameter to a DBreporter query. The data returned from the query are transmitted to the CDD address on the local network.

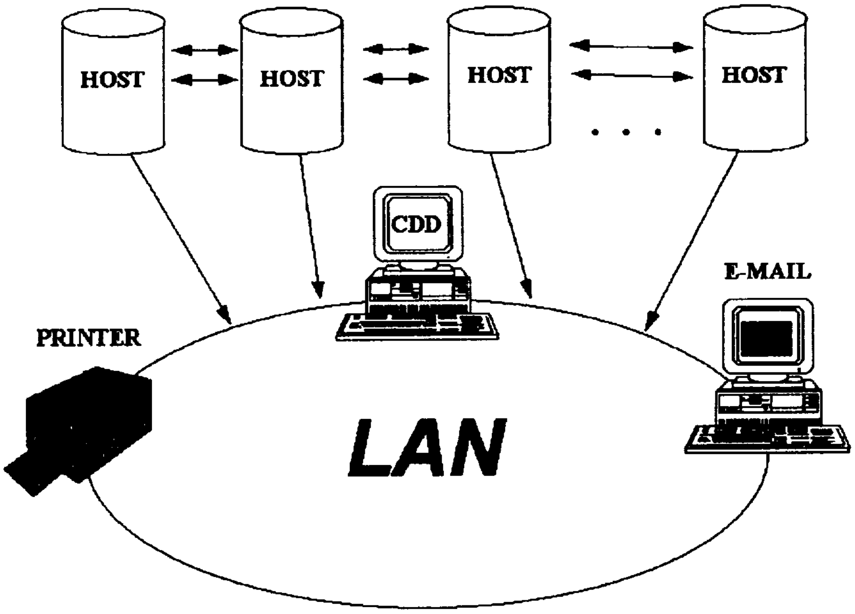
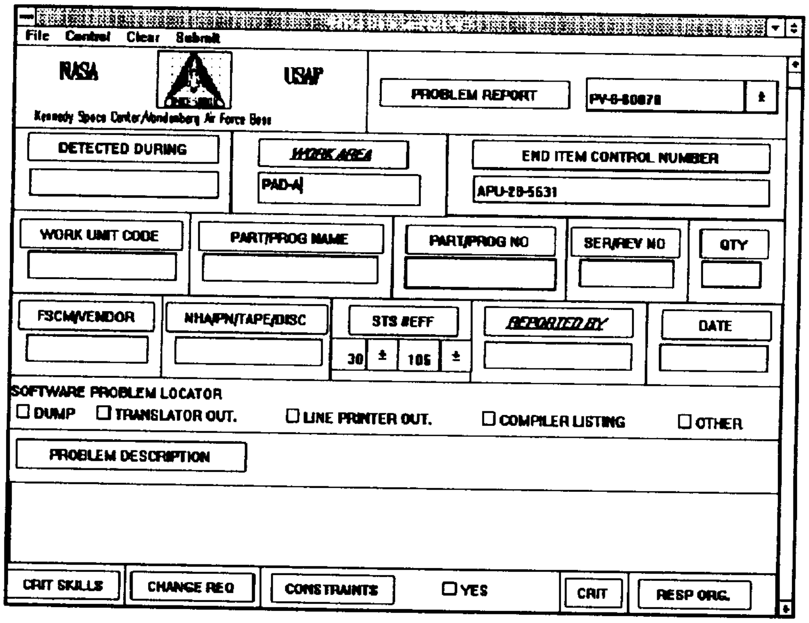


Figure 3 Data Transfer Paths of the ARS

Procedures are initiated and run against host database systems in a host-compatible language (SQL, LOUIS, Dbreporter). The resulting data is formatted into reports and transmitted through electronic routing paths to the user's local e_mail address, office printers, and the CDD (see Figure 3). The user's requirements determines the actual location where the data is to be transmitted. The normal policy is to have data reports intended for single users transmitted via e-mail or sent to a local printer. All other reports are sent into the CDD for general access.

FORMS QUERY

The Forms Query (FQ) system is a user interface linked through the CADRS server to several host "open" queries. Open queries are standardized common data queries with the actual parameter values missing. The user interface generates a file of these missing values which it transmits to the host system (see Figure 4). The basic user interface has been built around the concept of "forms." This is the use of a graphical representation of existing NASA forms for querying and displaying data (see Figure 5). All existing "Form" screens access a legal value dictionary and field identification system. This provides a context sensitive help and data identification system for the display. This also serves to restrict requests to actual legal values and data ranges. The interface program and host communication procedures were developed using C, Pascal, and Enfin development software. These access procedures support both Token Ring and synchronous communication lines.



PROBLEM REPORT: PV-6-60078

WORK AREA: PAD-A

END ITEM CONTROL NUMBER: APU-20-5631

STS #EFF: 30 ± 105 ±

Figure 4 Example Form in the Forms Query System

Once the user has entered all his data requirements into the form it is submitted to the CADRS server. The server converts the data into a set of variable values for a related open query. These values, along with the query template identification and user identification are transmitted to the indicated host system. An "initiator" program on the host system checks for this information on a periodic basis. When the program finds a parameter file waiting it submits the indicated open query with the data passed as parameters. The results of the query are transmitted down to the CADRS server where they can be accessed by the user.

The multi-tasking ability of Windows 3.1 allows the user to continue with other tasks while the query is being processed. When the data has been received the user can access it using the same forms the initial query request was made from. A number of additional utilities have been included in the system. These include a limited charting capability and data export facilities. Built-in data conversion functions from the ENFIN software libraries have been incorporated into the Query Forms system menus. This allows direct data conversion between common mainframe host data formats and workstation systems. Currently mainframe data can be converted into dbase, RBase, Oracle, Excel, SQL Server, and delimited ASCII. This allows the data to be imported into popular graphing and charting programs.

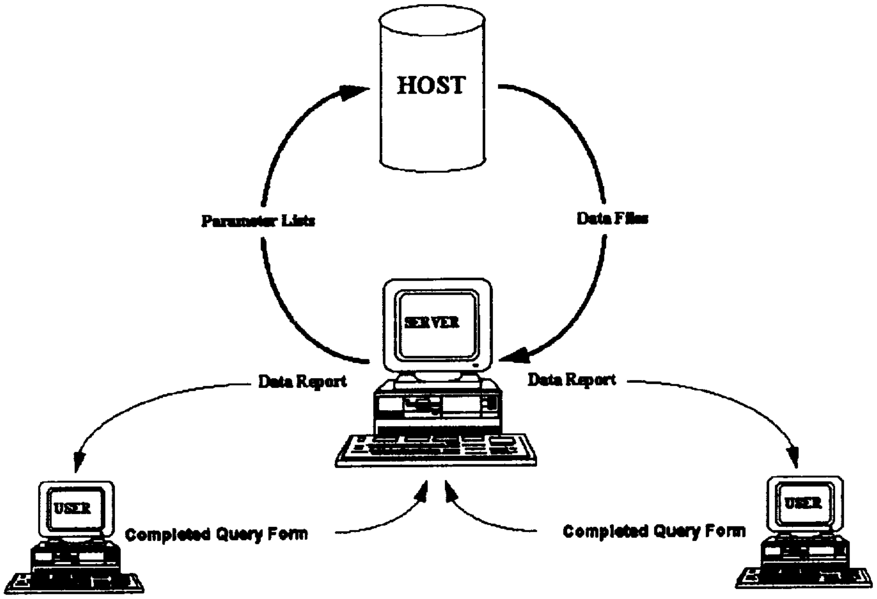


Figure 5 Data Interface of the Forms Query System

IMPLEMENTATION

The Central Document Database is installed on a Compaq SystemPro server. A one Gigabyte optical disk drive is used for local data and document storage. Communication links to mainframe systems are made through a Token Ring network and asynchronous data lines. The CDD server is connected to user workstations through Ethernet and Microsoft's Lan Manager software.

The Automated Reporting System activates itself every 24 hours during a non-peak time periods. Initiation is performed by using existing mainframe scheduling products (i.e. VMSchedual in IBM's VM system). A log command file is processed that identifies queries, data format procedures, and electronic addresses. On the local area network the CDD intercepts all data transmitted from ARS to its address. The CDD incorporates these data reports into its own database. Transmittals to individual users and network printers are identified by their own network address.

The Forms Query system resides both on the workstation as a user interface application and as open queries on the different mainframe systems. An executable procedure residing on each of the host systems checks at frequent time periods (one hour) for any parameter files sent from the interface application. These files identify the query, user and all variable values needed. The queries are run with the indicated variable values from the parameter file. Any data that are returned from the query is transmitted back to the local area network tagged with the user's ID.

SUMMARY

The CADRS system provides a unique solution to the problems of dissimilar and incompatible host systems, was compounded at NASA sites by multi-center and contractor operations. Currently, there are 27 different mainframe systems in widespread use by the space program. Included in this number are NASA specific systems as well as in-house contractor systems. Although, the situation at NASA is unusual it is by no means unique. Commercial industry with multiple legacy systems would find CADRS to be a viable option for data retrieval and dissemination. The system provides a low cost alternative to client/server systems when information retrieval is the primary consideration.

The three sub-systems of CADRS can be operated as a stand-alone system to provide improved data access. The CDD can be used on stand alone workstations to handle technical documents and manuals. Its ability to perform intelligent searches on large documents makes it well suited for reference systems. The ARS system provides techniques to automate standard data retrieval processes. This provides man-hour savings as well as shifting resource intensive tasks to non-peak periods. The Forms Query system provides a low cost graphical interface for performing common queries. These forms allow non-trained personnel to perform a greater percentage of the required data retrievals. Whether using all or a part of CADRS the benefits of the technology are obvious.

REFERENCES

[1] Gross, D., "Methods and Means Used in Programming Intelligent Searches of Technical Documents", NASA Conference Publication 3189, Volume 2, pp 55-62, 1992.

[2] Butler, J., "Middleware Needed to Plug Client/Server Holes", Software Magazine, July 1993 v13 n1O pg 55-57.

[3] Luger, G. and Stubblefield, W., "Artificial Intelligence and the Design of Expert Systems", Benjamin/Cummings Publishing Company, Inc. 1989.

[4] Minsky, M., "Semantic Information Processing", The Massachusetts Institute of Technology Press, 1968.

[5] Whittington, R.P., "Database Systems Engineering", Oxford University Press, 1988.

S34-32
2507
p.1

N94-32444

BEGINNING THE 21ST CENTURY
WITH ADVANCED AUTOMATIC PARTS IDENTIFICATION (API)

Fred Schramm
Technology Utilization Office
Marshall Space Flight Center, AL 35812

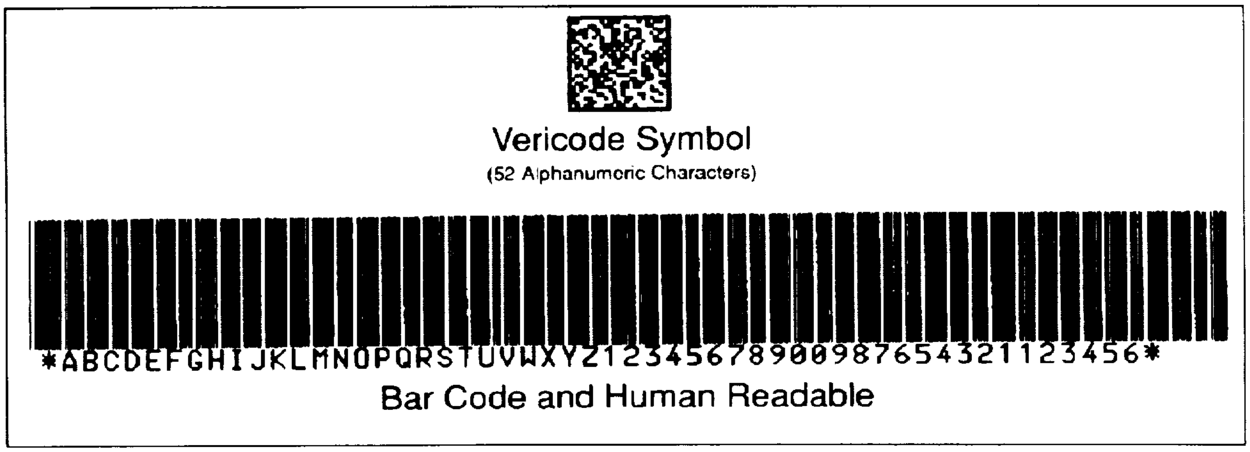
Don Roxby
Rockwell International
555 Discovery Drive
Huntsville, AL 35806

ABSTRACT

Under the direction of the NASA George C. Marshall Space Flight Center, Huntsville, Alabama, the development and commercialization of an advanced Automated Parts Identification (API) system is being undertaken by Rockwell International Corporation. The new API system is based on a variable sized, machine-readable, two-dimensional matrix symbol that can be applied directly onto most metallic and nonmetallic materials using safe, permanent marking methods. Its checkerboard-like structure is the most space efficient of all symbologies. This high data-density symbology can be applied to products of different material sizes and geometries using application-dependent, computer-driven marking devices. The high fidelity markings produced by these devices can then be captured using a specially designed camera linked to any IBM-compatible computer. Application of compressed symbology technology will reduce costs and improve quality, productivity, and processes in a wide variety of federal and commercial applications.

Existing Automated Identification Systems

There are thousands of applications for automatic identification. Although many technologies are available, most currently use bar code systems. Bar codes are one-dimensional systems and are generally attached to products using paper labels or tags, or by incorporating the code onto the product wrapper. This indirect marking approach, while suitable for retail sales, distribution, and other applications that are not paperless, is inadequate for marking products subject to harsh environments and handling. Bar coded paper labels, for example, are not tolerant of heat, cold, rain, wind, abrasion, chemicals, and other unfriendly conditions many products encounter during their life cycles. In addition to the limitations of typical bar code label material, the basic bar code design—long code length, fixed size and lack of error correction—has its own set of limitations when the decoding system attempts to deal with a variety of substrates.


Vericode Symbol
(52 Alphanumeric Characters)
ABCDEFGHIJKLMNOPQRSTUVWXYZ12345678900987654321123456
Bar Code and Human Readable

A comparison of VERICODE® and bar code symbologies using the same 52-character string.