ABSTRACT

For nearly a decade U.S. Army Combat Systems Test Activity (USACSTA) has been using computer based telemetry systems to acquire data in a variety of ground vehicles and weapons systems. Three years ago the Advanced Acquisition and Analysis Through Year 2000 (A³-2000) project was initiated with the goals of: providing instrumentation systems able to meet the demands of ever more complex weapon systems, increasing workload capacity with no increase in the number or quality of personnel, and reducing the total test cycle time for the customer. The methods selected to implement these goals were: increased reliance on state-of-the-art computer technology, the use of standards for operating systems, programming languages, networking, and interfaces, the use of expert/knowledge based systems, and integration of all organizational computer resources. This paper will describe the progress which has been made in all of these areas.

INTRODUCTION

Over ten years ago, U.S. Army Combat Systems Test Activity (USACSTA) initiated a program to modernize its data acquisition and analysis capabilities. The primary goals of the initiative were to:

a) Provide a data acquisition system which could meet the demands of increasingly complex and technologically sophisticated test items.

b) Provide real-time information to test operating personnel suitable for test execution decision making.

c) Reduce the requirement for highly skilled technicians to operate the data acquisition system.

d) Reduce the time between completion of testing and preparation of the final report.
The concept chosen to accomplish these goals was to place a minicomputer at the test site to control the data acquisition process and provide on-site analysis capability. The minicomputer and associated data acquisition and analysis system are called a Test Site Terminal (TST). The TST addressed the goals by:

a) Placing a major portion of system operation in the hands of the computer thus freeing the operator from mundane details to provide more time for quality assurance.

b) Providing instant access to data to reduce post-test processing and eliminate costly retesting.

The USACSTA data acquisition system proved to be very successful, as indicated by its growth. In 1986, USACSTA had 28 minicomputer data acquisition and processing systems of one type or another.

In the 14 years since the USACSTA’s goals were first formulated a great deal of progress has been made. Test requirements that would have been impossible to address in 1974 are now handled routinely. However, the goals expressed earlier remain as valid today as they were in 1974.

The computational platform used in USACSTA’s original system was a 16 bit general purpose minicomputer with a real-time operating system. Although this system was state-of-the-art for the 1970s, it is no longer technologically current. More importantly, these computers have run out of computational horsepower. Applications that are needed immediately are not possible because of computer system limitations. So if the original modernization goals are to be continued, a new computer system with sufficient power and flexibility to meet the demands of the 1990s had to be selected and integrated into USACSTA’s data acquisition and analysis base.

DESIRED COMPUTER CHARACTERISTICS

All of USACSTA’s original Telemetry TSTs are built around an older series of minicomputers. The reasons for selecting a single computer system type are:

a) To provide for hardware interchangeability.

b) To provide a large skill base in the operation of the systems.

c) To provide software commonality.
However, there are some problems with the single source computer solution:

a) The operating system used is peculiar to the computer system.

b) There is often a mismatch between the computer performance and the job requirements.

c) There is no guarantee that the selected computer system will provide a growth path for migration in future years.

Taking these factors into consideration and examining the data acquisition and analysis needs of USACSTA led to the following computer system criteria:

a) Operating system vendor independence. Much of the nonportability of USACSTA’s older software is related to the use of proprietary operating system and library routines. A nonproprietary or a nonvendor specific operating system eliminates some of this problem. The UNIX™ operating system most closely approximates this requirement. According to Young and Manual (1987) the UNIX share of the operating system market consists of 83 percent of technical work stations 75 percent of superminicomputers, and 6 percent of all computers. UNIX is a proprietary operating system, but it is licensed to a large number of vendors and has sufficient market place inertia that it is not subject to control by a single vendor. This is demonstrated by two recent movements: (a) a ruling by General Services Administration that the Air Force specification of conformation to UNIX did not violate federal competition rules (McCarthy (1987)), and (b) the ongoing effort by the Institute of Electrical and Electronic Engineers (IEEE) to develop a vendor independent standard (Young and Manual (1987)). UNIX is quickly becoming a truly vendor independent standard.

b) Real-time operating system. To acquire data at the rates required by some USACSTA applications necessitates an operating system with real-time capability to avoid gaps in the data. UNIX is not a real-time operating system; however, several vendors provide UNIX with extensions which provide real-time capability. Unfortunately, these extensions are not portable from vendor to vendor. But this is a better situation than having a proprietary operating system. In addition, while the real-time extensions are vital, they are not used extensively.

c) Standard programming language. A majority of the software developed by USACSTA is written in some version of FORTRAN. Currently, FORTRAN77 is a standard (American National Standards Institute) programming language available on many computers. It provides a means of developing software which can be used over a
wide spectrum of machines and vendors. The availability of other standard languages is also a consideration, but due to its strong input/output features and its engineering/scientific emphasis, FORTRAN77 is the most important language requirement.

d) High-speed data acquisition. Data acquisition rates of 150,000 samples per second to disk and 270,000 samples per second to memory are routinely performed using the current systems.

e) Input/output interfaces. Several different interface types are necessary to connect and operate the many different instruments used by USACSTA.

1. High-speed parallel. Sixteen bit input/output with handshaking for high-speed control and data acquisition,


f) Improved computational performance. Testing is sometimes slowed or delayed by the computational speed of the existing system. Increased computational speed would eliminate delays and provide the capability to perform additional functions.

g) Digital signal processing. Many tests require analysis using digital signal processing techniques. Currently, these requirements are handled using an attached array processor. The array processor performs digital signal processing much faster than the computer can alone. It is anticipated that digital signal processing will become even more important in the future, An integral array processor is preferable to an attached processor.

h) Thirty-two bit processor. The processor address capability is related to the processor word size. The older 16 bit word size severely limits program size. The 32 bit word size should provide adequate address space for future growth.

i) Wide range of machines. The older system is built around a single computer. There is no range of machines around which to build systems to meet a variety of performance requirements. A range of machines from small desk tops to large multiprocessor configuration is desirable.
j) Multivendor networking capability. Interconnecting computers from various vendors is essential in many instances to facilitate data transfer or increase computational performance. The availability of a standard networking scheme guarantees ease of connectivity.

CURRENT SITUATION

Over the past three years many actions have taken place under the Advanced Acquisition and Analysis Through the Year 2000 (A³-2000) umbrella. Efforts have been directed toward establishing vendor independent data acquisition and processing systems. During this time 12 UNIX based Vehicle Performance Recorder Data Retrieval Systems, 4 Telemetry Test Site Terminals (TTSTs), 3 Digital Doppler Test Site Terminals, 9 Ballistic Test Site Terminals plus numerous analyst’s work stations have been procured. Concentrated effort has been expended by both government and contract personnel to port all existing code to the UNIX environment. To date, approximately 85,000 lines of FORTRAN source code have been so ported. To aid the porting process, and to prevent any future revisiting of hardware control over software implementations, a set of portability guidelines has been developed which is applied to software conversions as well as new software development. It has been demonstrated that portable source code can be written for a multivendor environment. Through the use of high performance computers it is possible to write portable source code and still maintain high performance.

HARDWARE DESIGN

A block diagram of the basic USACSTA PCM Telemetry Test Site Terminal is shown in Figure 1. This system is composed of a vehicle mounted subsystem; the signal conditioner, PCM encoder, and transmitter; the PCM ground station, and a real-time UNIX computer. The signal conditioning provides the interface between various types of transducers and the PCM encoder. Transducer types include accelerometers (bridge, piezoresistive, or piezoelectric), strain gages, pulse generators, thermocouples, and others. The encoder multiplexes and digitizes analog channels, multiplexes digital data channels, generates synchronization patterns, and serializes the data into a bit stream. The ground station restructures the data into parallel form and presents the data together with timing information for recording and display on the host computer. The telemetry data acquisition system is composed of a mix of commercially purchased units and in-house developed items.

Data from the ground station is transferred through a 16 bit parallel input/output port to the host computer. The input port is capable of 400,000 words per second in handshaking mode. The output port is capable of 175,000 samples per second in handshaking mode.
The resulting data is stored in a direct access data file as 16 bit integers. Quick look or data analysis programs can be run on the data at this point.

SOFTWARE DESIGN

The design objectives were to create a standard file structure, taking full advantage of the UNIX hierarchical structure, and to keep the user interface essentially the same as the original software.

The file structure of the basic Telemetry Test Site Terminal (TTST) employs the normal UNIX hierarchical file system to ensure portability to other systems. A diagram of the basic TTST file structure is shown in Figure 2.

PROGRAMMING STANDARDS AND GUIDELINES

The CSTA data acquisition system employs industry standards in every area possible. Beyond these standards, however, internal standards have been implemented to further enhance portability and re-useability. The flexibility inherent in many higher level languages provides a programmer with the means to construct programs, routines, and algorithms consisting of many styles and forms. Although these may satisfy the problem at hand they invariably reflect the personality of the designer. When such routines must be integrated into an existing software library for use and maintenance by personnel other than the author problems arise. Therefore the following policy, guidelines, and standards are set forth.

a) In order to facilitate algorithm design, to increase maintainability, and to reduce the development time, a Top-Down-Structure policy has been implemented.

b) The following guidelines should be strictly adhered to:

1. Prior to the creation of a new routine or algorithm, existing libraries should be searched for desired functions. This procedure will reduce replication of effort.

2. Routines should be designed in a modular fashion. In addition software should be designed in a cosmopolitan manner, i.e., thought should go beyond the specific application so as possibly to end up with a multi-use routine.

3. In house programming standards must be utilized.
c) The following constitute the minimum programming standards for higher level languages:

1. Programs and separately compiled routines, modules, or functions must each contain a header block which delineates revision level, date, author, pertinent history, and a brief description of function.

2. Programs must provide a description and definition of input/output variables.

3. Separately compiled routines and functions must provide a definition of all parameters and/or common variables used.

4. Variable declaration will be primarily ordered by byte size (descending) and further arranged in alphabetical order.

5. Names and/or structures of required files must be listed in a program or routine’s header block.

6. Utilization of the ‘GO TO’ statement should be restricted, implementing the ‘IF...THEN ELSE’ construction whenever possible.

7. Use of the ‘arithmetic if” construction is not recommend.

8. Executable statement numbers should begin at 100 and increase monotonically over the extent of a program or subroutine. A uniform increment of 10 is recommended. All statement numbers should begin in column 1.

9. Judiciously placed comment statements should be included as means of built in documentation.

10. Data typing should be restricted to FORTRAN 77’s default values whenever possible (I--N integer, everything else real).

11. Subroutines should appear in alphabetical order subsequent to the driver routine or main program.

PORTABILITY ISSUES AND GUIDELINES

It is important that the programmer knows the standard FORTRAN 77 language features and is aware of the extensions each manufacturer has implemented. Compiler documentation does not always call attention to the extensions in the main body of the
compiler manual, but rather in an appendix. Some compilers provide an option which checks for the use of extensions and provides warning messages when extensions are encountered. This feature should be used if possible.

Although the FORTRAN 77 standard provides great flexibility and utility, most manufacturers provide extensions to the language. The extensions often provide backward compatibility for older compilers or provide enhancements that users have requested. While these extensions often make the programmer’s job easier or enhance software performance, in general the extensions are not portable. If it is necessary to use the extension, then the extension should be placed in a short subroutine or function which is called by the user.

Manufacturers normally supply library routines which perform a wide range of auxiliary functions. Essential to USACSTA software operation is access to operating features such as: passing run string parameters, scheduling programs from a program, and reading system time. The recommended approach is to place the proprietary routines in a subroutine which uses the same calling sequence for any machine.

**NETWORKING**

A multivendor local area network (LAN) consisting of systems from several vendors and utilizing 3 operating systems (UNIX, DOS and VMS) has been established. This LAN has been implemented in a small area as a pilot project to integrate engineering and office functions. The VMS portion of the network is included because of a particularly nonportable set of software dealing with the processing of radar data. The DOS portion of the network, which makes use of a large installed base of machines, provides workstations for the scientific and engineering staff as well as office automation features.

**FUTURE INITIATIVES**

The major thrusts so far with A-2000 have been directed toward operation of test systems in a manner nearly identical to the operation of older data acquisition systems. Efforts so far have provided incremental improvements over the older systems in terms of processing speed, but so far no revolutionary changes have been experienced. The next phase of this effort, the introduction of expert systems, is expected to provide dramatic impact.

The large diversity of testing that is performed at USACSTA requires that instrumentation systems be designed to cover a broad spectrum of applications. This flexibility of configuration in turn imparts a level of complexity in readying an instrumentation system for a particular application. Currently, system configuration, which in general is implemented through a series of computer files, is the responsibility of experienced, well
trained engineers and technicians whose breadth of knowledge of test item, transducers, and instrumentation system allows them to properly prepare for any given test. Work has been initiated, both in-house and through contract, to capture the expertise of these test and evaluation experts into a formalized expert system.

Another area that is an anticipated area of exploitation for an expert system is that of data quality control. After data records have been captured they are generally examined for “correctness”. Experienced engineers and technicians subject data to various tests (sometimes not realizing that they are doing so) and make real time decisions relating to data quality and whether or not a test should proceed. By capturing the knowledge of these personnel into an expert system formalism their expertise can be leveraged through wide application throughout USACSTA.

The key to making the expert system function is an on-line data base system, which in turn will depend upon a high performance network and a central computing facility supporting high speed data base access. USACSTA is currently purchasing a high speed, UNIX based computer for a central computing facility and plans are being formulated for a USACSTA wide network that encompasses test site instrumentation systems, engineering work stations and the central facility. The current pilot network is being used to test networking concepts and the network media (principally broadband cable) is being readied for use.

CONCLUSION

USACSTA is in the midst of major changes in the way data are acquired and processed. These changes so far have provided incremental improvements, but much greater benefits are anticipated. The change from “old, worn, comfortable” computer systems to systems with a much more complicated operating system is proving to be somewhat painful, but these changes are deemed to be necessary to meet future testing needs.

REFERENCES


Figure 1. Block Diagram of PCM Telemetry Test Site Terminal

Figure 2. Basic TTST File Structure