SYSTEMS AND METHODS TO REDUCE DATA PROCESSING TURNAROUND TIME

Robert Crolene
Code 3442
Pacific Missile Test Center
Pt. Mugu, CA 93042

ABSTRACT

Weapon system complexity and its data expression have become a central issue for the Range Directorate at the Pacific Missile Test Center (PMTC). Increasing data complexity and data product turnaround requirements have created a technological push-pull on traditional data processing methods. Several possible responses are discussed which include distributed front and back end processing relative to the large mainframes, and increasing use of artificial intelligence techniques in the data reduction area. These methods are going through progressive steps of implementation at PMTC with some notable success.

INTRODUCTION

A top-level view of the test and evaluation process from the data perspective is depicted in Figure 1. Within this schema in realtime, the data is captured, the test is monitored and controlled, and systems oriented impressions are gained. True evaluation, however, must take place down the postflight timeline as indicated on the figure, and the postflight domain will be the focus of this paper. After the data is captured it must be processed and qualified and passed on to the project's database management system (DBMS). Ideally, thereafter, the information is networked out to subscribers and members including systems engineers and analysts. The data is then analyzed in the wider context of past operations and simulations, and the operationally tested systems are evaluated. This then completes the evaluation component of the test and evaluation couple.

By far the most significant bottleneck to the data flow described above occurs within the "Process and Qualify Data" module in Figure 1. Within this module the raw packed test data must be converted to compatible sequential records and scaled, processed and displayed according to data type and project specifications, and formally qualified (e.g., a quality measure may be affixed to each logical record according to some convention).

This cluster of traditional data processing tasks finds itself today in an environment of increasing pressure from two directions. First, there is currently a profound escalation in telemetry data rates and complexity (see Figure 2) which the present front-end decommutation and formatting systems have great difficulty coping with on a timely basis. Secondly, the data product service requirements are increasing in terms of more stringent turnaround ("overnight" turnaround for the best and final data package is typical) and more thorough and objective data quality assurance. The net result of these forces is to create a technological push-pull on the traditional data processing and qualification methodologies. Several working solutions to this situation have been under development at PMTC and will be discussed below.

THE COMPLEXITY ISSUE AND ITS IMPACT ON FRONT-END PROCESSING

Figure 2 depicts the exponential growth of telemetry data rates in modern weapon systems with time. Only a few examples are given, but they well represent a generational tendency towards not only higher data rates but also larger frame sizes, and increasing variability in most everything including commutation logic and frame, group, and word formats. With additional project-specific ad hoc features thrown in such as track interleaving and unique recording media, the state of the art of telemetry data transmission is racing well beyond traditional front-end processing capabilities. The tendency is towards a situation of nearly free representation of huge densely-packed streams of information, which, of course, presents significant problems downstream where all this encoded information must be recovered.

A large portion of the postflight information recovery burden has been placed upon the general purpose mainframe computers in terms of reformatting the partially processed telemetry streams. A specific example of mainframe processing is discussed here in which single tracks of 1553 avionics data are split into multiple tracks to reduce recording bandwidth. The resulting confounding of information predicates the use of buffering and nested logic to generate fixed subframes. After a cycle of software development and optimization of reformat code converging to a state of cp-boundedness (where efficiency is now limited only by the speed of the central processor), a one hour operational test today requires at least 24 hours of machine time to do the data reformatting. Figure 3 depicts this in the central portion of the top bargraph. The true measure of the cost for such a circumstance must include not only the resource cost and time delay to the project but also the more intangible price to other users and projects who are inevitably displaced and delayed.

The above situation demonstrates a need for dedicated front-end processing to relieve the mainframe from a mundane but very expensive application of its capabilities in the postflight arena. The state of development of small parallel processors utilizing data flow architecture has sufficiently matured[1,6] to the point that the use of such a device dedicated to reformatting telemetry data could hypothetically excel by orders of magnitude in performance/price the use of the mainframe trying to support that same function.

The practical application of a data flow computer to the task of reformatting high-rate telemetry data online (viz., keeping up with realtime or playback speed) has been studied with pronouncedly favorable results[5]. If such a system could be configured within PMTC's postflight environment, then the current trivialization of the mainframe's capabilities would be reduced with benefits to current and future projects. This as well as other possibilities are being evaluated in the context of an overall upgrade plan.

REDUCING THE DATA REDUCTION BOTTLENECK

After the data (telemetry, radar, or any other type) has been placed in a compatible sequential format, it must be processed and qualified according to data type and project requirements. A wide spectrum of processing options is available at this point from doing nothing at all except checking channel assignments and outputting some graphics and send-out tapes, up through the application of sophisticated time series analysis and optimization techniques (as is routinely done for radar data).

The traditional methodology for fulfilling such a set of specific requirements is a manual, sequentially latched, labor intensive system involving tape interfaces and almost exclusive use of dumps and listings for quality control. Such methods, although well suited to the constraints of the 1960's and early 1970's, do not utilize today's interactive third generation capabilities which can provide a quantum leap in performance when suitably applied.

A modernization effort to eliminate this technology gap began several years ago at PMTC and yielded a data reduction methodology with the following features:

1) Subtasks are integrated into one software module at a graphics workstation.
2) This workstation is operated by a single data quality manager (called an "analyst" from here on) who is fully accountable and responsible for his operational data.
3) The user interface consists almost entirely of diagnostic graphics and command or menu driven screens.

This approach although showing great promise even during the prototype stages, urgently called out for further development.

EVOLUTION OF THE WORKSTATION APPROACH

The increased data processing efficiency created by the functional integration discussed above, shrank the time scale in which a standard set of decisions had to be made by the analyst. Instead of having a week to mull over possible data strategies, the analyst now had to do about the same thing within an hour or so. This created a state of intellectual and even emotional overload in the analyst causing discouragement and “mental gridlock”. The solution to the dilemna was found to be the use of artificial intelligence to reduce the decision load as follows:

1) A high-level editor was implemented that simulates visual recognition of bad data regions and automatically keys downstream processing accordingly.
2) A simple process monitor was implemented that keeps track of history and goals and advises the analyst accordingly.

Neither one of these upgrades at the time were developed using expert system architecture (reference 4 for example), but were nevertheless very effective in making the interactive systems tolerable and even friendly.

Two more expert systems (this time with “sound” architecture although done in Fortran) were added to the analyst’s toolset. One of them diagnoses radar data problems and recommends solutions (SHERLOK) and the other describes probabilities of success and pitfalls in alternate strategies (PROJEX). Neither of these have been heavily used, mainly because they are not tightly coupled into the working environment. A more generic expert system tool (or “shell”) can be utilized to integrate these and other knowledge resources into the runtime procedures wherein their presence would not be seen unless needed. This topic is further discussed below in the context of computer based training and process control.

The so-called graphics workstations on which the interactive software was implemented were not real workstations in the sense of supporting distributed processing. They were time-share graphics terminals totally dependent on the mainframe’s resource status, a dependency that very often did not insure high reliability in meeting deadlines. The next developmental step was thus defined to be the implementation of a workstation with superminicomputer capability, and to migrate the software modules and associated processing down to that local environment. This phase has begun and will be underway for some time. Small token networks of perhaps three or four workstations in each of the classified and unclassified domains will eventually serve postflight interactive data processing needs. The bulk of the non-interactive or batch data processing which requires great number crunching and I/O capabilities can only be accomplished on the mainframe just as it currently is done.

Although sophisticated processing tools such as discussed above are developed and implemented, their correct future application is always problematical. Formal training programs, job aids support, and even intelligent computer assisted instruction (ICAI), and/or a process control system (PCS) should also be part of the whole package to help assure correct long term application. Training and job aids for the interactive systems have been developed using principles of instructional systems design. The ICAI and PCS which are expert system based[7] require either the purchase or development of an expert system shell and then supplying that shell with a knowledge base unique to the task at hand[2]. The shell currently under development in the Range Instrumentation Systems Department at PMTC is being done in Fortran 77 so that it can be eventually coupled to a variety of existing software systems.

SUMMARY AND CONCLUSIONS

Escalating front-end telemetry and data product turnaround requirements must be answered by a level of technology and expertise equal to the challenge. The use of parallel processors with data flow architecture was discussed above as the beginning of a realistic response to present and future front-end requirements. The current implementation of interactive systems with artificial intelligence techniques demonstrates dramatic improvement in data product turnaround time. Emergent responses to data complexity involve not only the front-end data flow processing but also back-end distributed processing in token network workstations. A richly interconnected development of all these aspects can help support the test and evaluation requirements of the modern Navy.

ACKNOWLEDGEMENTS

Gratitude must be extended to Mike Burdullis and Tom Maulding for their creative insights into the nature of telemetry front-end processing.

REFERENCES

1) Breedlove, P.M. and Stevens, C.R., "Real-Time Telemetry of Complete 1553 Data Buses", Proceedings of the 1985 International Telemetry Conference.
2) Citrenbaum, R., "A Practical Cost-Conscious Expert System Development Methodology", Proceedings of the Second Annual Artificial Intelligence and Advanced Computer Technology Conference, (organized by Tower Conference Management Co., Wheaton, Illinois), 1986.
3) Crolene, R., Post-Flight Data Processing at PMTC - Integrated DQA Plan, PMTC Technical Note 3440-01-86, Pt. Mugu, CA, January 1966.
4) Harmon, P. and King, D., Expert Systems: Artificial Intelligence in Business, John Wiley & Sons, 1985.

5) Hightower, C.H. and Patterson, W.M., “Telemetry Processing Applications of Dataflow Techniques”, Proceedings of the 1986 International Conference on Parallel Processing (to be published).
6) Williamson, G., “User-Programmable Real-Time Processing in Modern Telemetry Systems”, Proceedings of the 1985 International Telemetry Conference.
7) Woolf, B. and McDonald, D., “Building a Computer Tutor: Design Issues”, IEEE Computer, September, 1984.

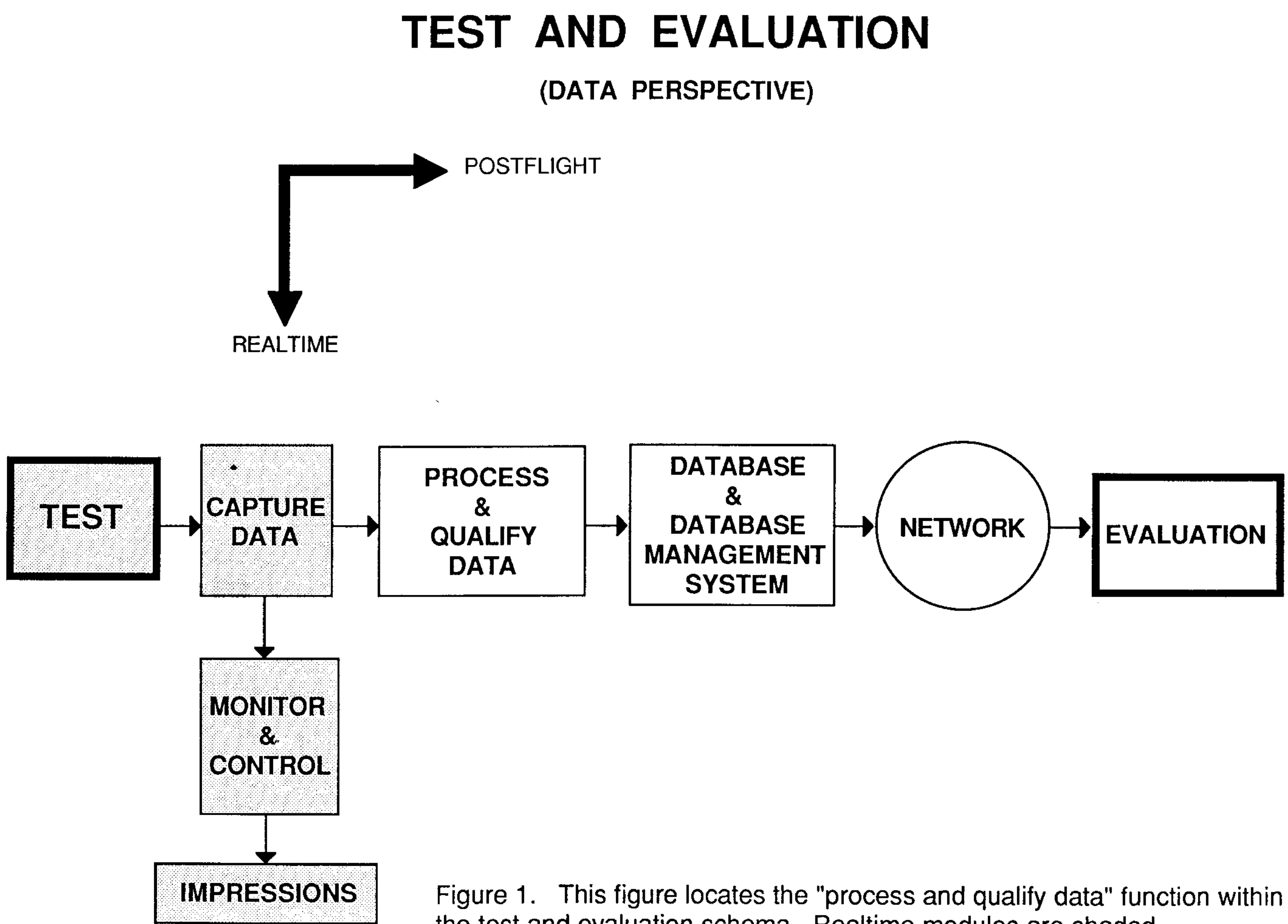

Figure 1. This figure locates the "process and qualify data" function within the test and evaluation schema. Realtime modules are shaded.

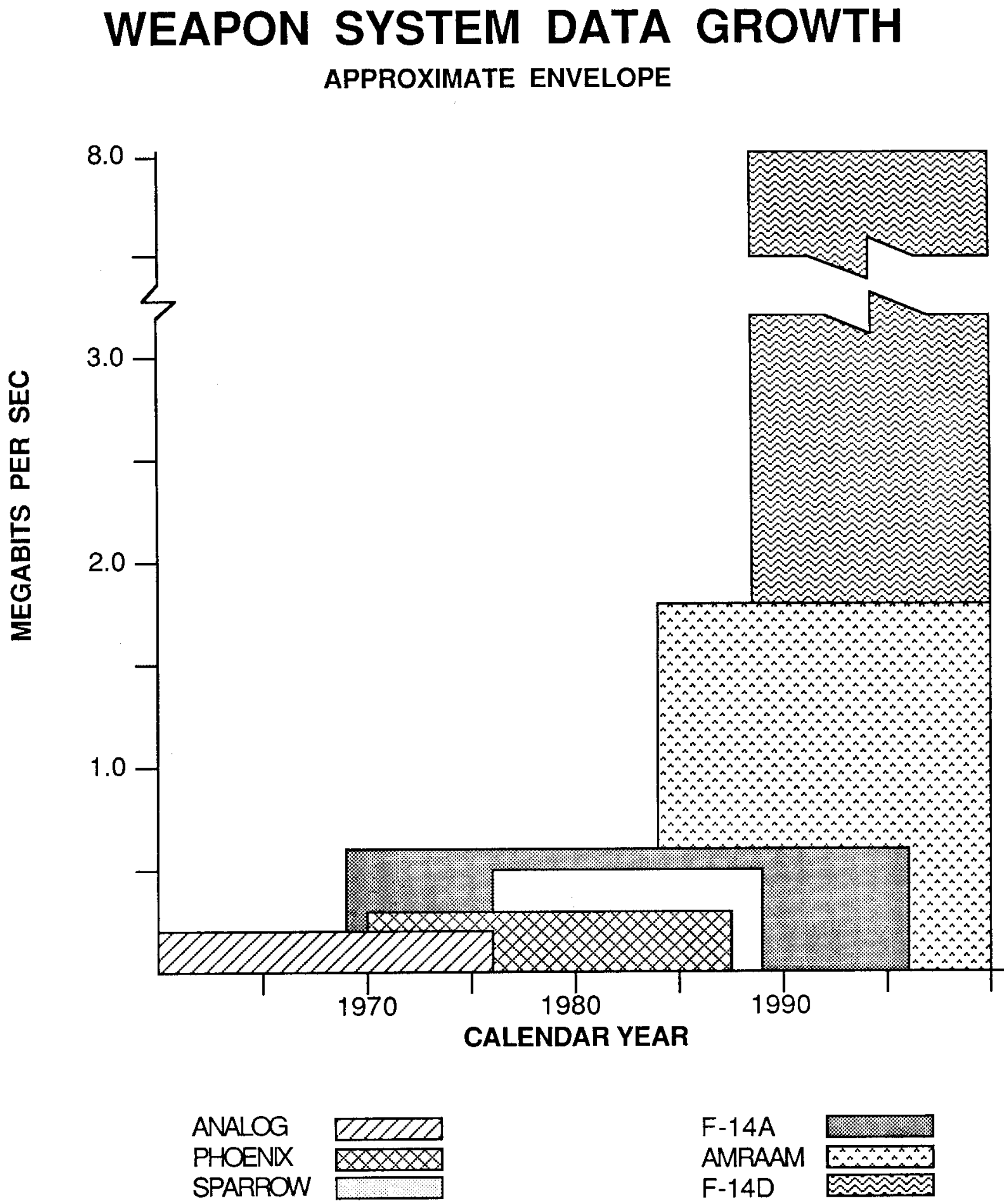

Figure 2. This figure illustrates the precipitous growth in telemetry data using a few representative projects.

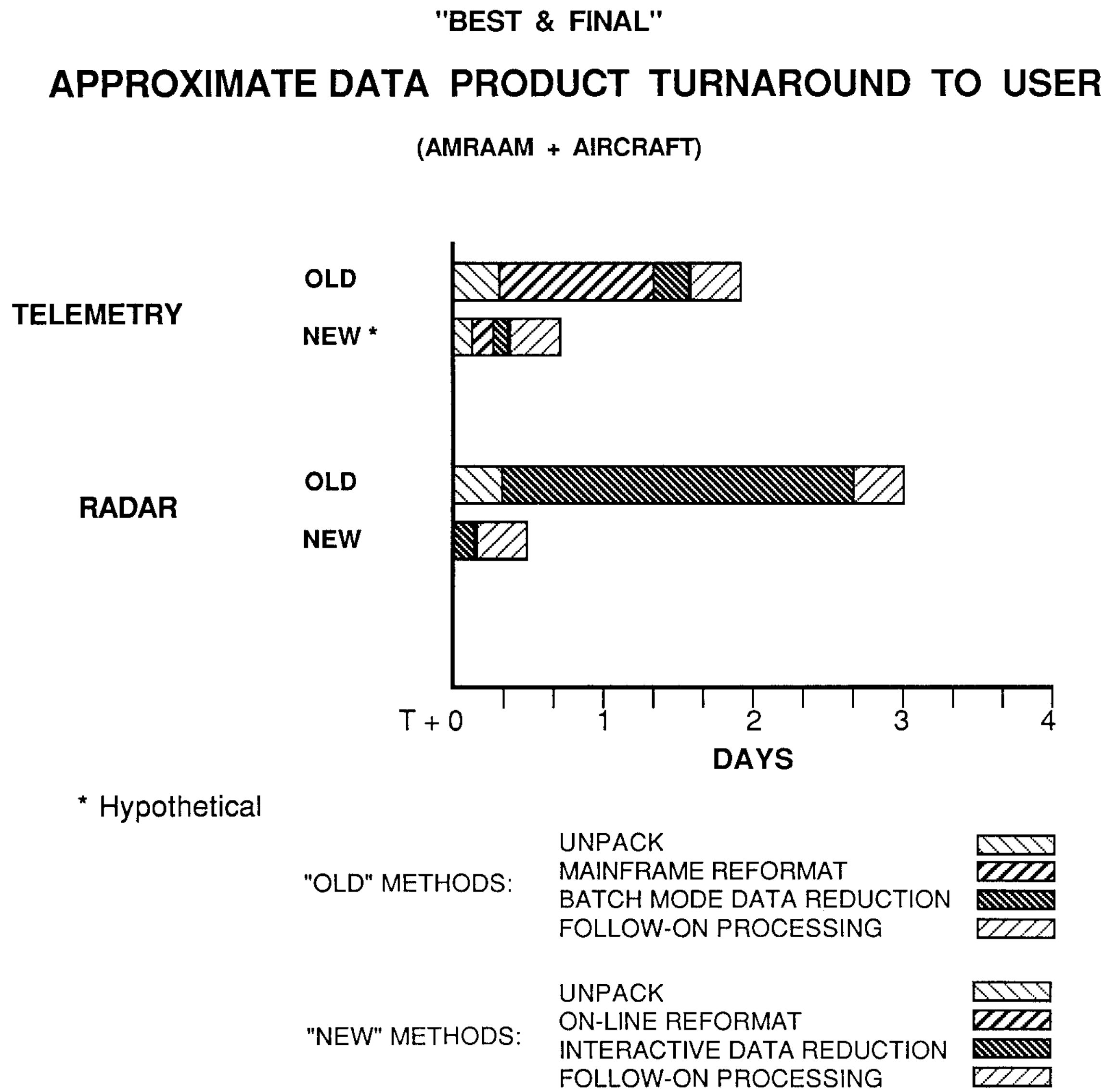

Figure 3. For a representative project this figure compares results using two methodologies--the more traditional batch mode mainframe approach vs. interactive distributed processing (certain aspects still hypothetical).