IEEE Transactions on Nuclear Science, Vol. NS-26, No. 4, August 1979

THE GSI DATA ACQUISITION SYSTEM (EDAS)

Ronald Chestnut (1),Barbara Grasmueck, Richard Hadsell (2)
Edith Krieger, Waltraud Lebershausen, Juergen Lowsky,
Winfried Plappert, Matthias Richter, Hans-Peter Rother(3),
Otto Siart, Ruth Thieme
Gesellschaft fuer Schwerionenforschung,
6100 Darmstadt, Federal Republic of Germany.

ABSTRACT

An integrated facility-wide system for nuclear physics data acquisition is described. Four PDP-11/45's connected via a SYSTEM/7 to an IBM/370-168 are used in the system. User programs specifying spectrum definition and analysis logic are written using macros and are expanded into PL11 or PL/1, depending on the subsystem into which the program is to be linked. Both subsystems also use a common command set.

I. INTRODUCTION

Foregoing the use of digital computers for nuclear data analysis is nowadays much the exception to the rule. Systems in use range from computer based multi-channel analyzers with several ADC's, to small computers equipped with special I/O devices (1-6), up to systems of coupled computers (7-10). Most of these systems are fairly static, the user either being required to write everything from I/O to analysis programs (mostly using a system of tailored subroutines) or only being able to define the course of the data analysis under certain restrictions. In most installations this is completely acceptable, since the types of experiments handled by one computer system remain constant while demands are slowly changing.

The GSI (Gesellschaft fuer Schwerionenforschung) is a research laboratory equipped with a large heavy-ion linear accelerator (the UNILAC). At present there are up to 17 high-energy (8.6 MeV/nucleon) experimental setups in addition to nine in the low-energy area. It is possible to have beam at one low-energy experiment and three high-energy experiments simultaneously. Most setups are not static but, due to the exploratory and speculative nature of present heavy ion experiments, evolve from beam time to beam time or, in the case of several general purpose stations, change completely. The types of data measurement range from simple spectrum accumulation, as would befit a multi-channel analyzer, up to multiple coincidences between multi-parameter detectors, more in the direction of high-energy physics experiments. The management of such data is one of the major tasks of the GSI computing facilities.

The Experimental Data Acquisition System (EDAS) (11) was conceived to meet the following specifications:

Flexibility, to cope with rapidly changing demands of various experiments;

Simplicity, so that the user need not be a computer expert;

Acquisition and storage of raw data, at high data rates if need be;

Analysis of data 'on-line' for technical and scientific control of the experimental parameters;

Automatic or interactive control of the experimental apparatus;

Analysis of collected data 'off-line';

Graphical representation of results (mostly spectra).

Four well-equipped PDP-11/45's connected by a fast data link (specifications 400 kByte/sec; presently working with 120 kByte/sec) to an IBM/370-168 are available to cope with the data acquisition problem. The workload can be divided between the large system and a given satellite computer in one of two ways:

In stand-alone mode the satellite accepts data from an experiment, writes them onto magnetic tape, and analyzes a portion of them during the run. Later the data tapes are fully analyzed on the large system. This method has three limitations:

low data transfer rate to magnetic tape (=< 25 kBytes/s)

restricted space for analysis programs and spectra during experiment time (=< 184 kBytes)

no complicated online computations possible (no floating-point hardware)

---

(1) now at SLAC, Stanford University, Palo Alto, Cal. 94305
(2) now at IBM Watson Research Laboratory, New York
(3) now with Amdahl Computer Corporation, Frankfurt

In coupled mode the satellite accepts data from the experiment and performs trivial analysis. The data are simultaneously transfered via the fast data link to the large computer system, where they are more fully analyzed and can be written at a high data rate onto tape or disk. Thus the analysis complexity is not limited by the satellite computer.

For any analysis more complicated than just an accumulation of singles spectra, the analysis program must be tailored explicitly to the user's problem in both the PDP-11/45 and the IBM/370-168. The work required of the user was to be kept at a minimum, and this goal led to the concept of a single command structure and program system for both computer types. The ideas in the software concept are based on experience with earlier systems at the Yale Van de Graaff and the Munich Tandem Laboratory. The hardware configuration was strongly influenced by SLAC. The two subsystems 'GSI On-Line Data Acquisition System (GOLDA)' and the 'System to Analyze Tremendous Amounts of Nuclear data (SATAN)', were functional in a preliminary form when regularly scheduled beam time was first available, at the beginning of 1976.

II. THE IDEAS BEHIND EDAS

Adaptation to the Experiment.

Due to the multifarious demands on EDAS it was not conceived as a black box containing all imaginable analysis possibilities (1,9), but rather as a system which contains all standard features and calls user-supplied subroutines in which the experimental configuration and analysis logic are specified. Each subroutine is written in a problem-oriented analysis language derived from a high-level language, supplemented by special language elements (Macros). Subroutine and Macro libraries are supplied for all standard functions. The major advantage of such a modular system is the ease with which both user segments and system extensions may be added. The system is not static; it may be easily adapted to new requirements.

Interactive Control.

The user controls EDAS via commands which he enters on a terminal in a transparent and flexible command language. A multitude of system commands are available, for example, to control data flow or generate displays or plots of accumulated spectra. User-written commands may be easily defined and integrated into the system to deal with special problems.

Asynchronous I/O and Analysis .

At high data rates or in the case of extensive analysis, it is often impossible to analyze all of the data from a given experiment on-line. In order not to impede data collection (for example, by introducing dead time) it is useful to write all data to tape, but to analyze only as much as there is time for. This requires a multi-tasking system where the analysis runs at low priority and data is accepted and written at high priority. Due to the statistical nature of most nuclear data the user retains sufficient control.

The Structure of EDAS.

Irrespective of the computer-dependent implementation, EDAS comprises the following components:

I/O of raw data with source and destination differing according to the mode of utilization
(stand-alone, coupled, or replay mode);

Data sorting according to the desired events and dispatching to the corresponding analysis routine;

Processing the data to obtain results;

User control of data flow and display functions.

The Analyzer Concept

Independent of the various physics problems with which an experiment deals, the analysis of data can be reduced to the following basic elements:

Numerical calculations using the measured parameters to generate new parameters (e.g. energy calibration, calculation of mass from time of flight and energy);

Accumulation of spectra from measured or calculated parameters;

Window checking;

Branching according to the results of window checks.

The user himself must program any numerical portions of the program, for which a high-level language is at his disposal. The three other basic functions may be standardized. In EDAS the 'Analyzer' was conceived for this purpose.

Although a spectrum may be associated with an analyzer, the latter's main function is not spectrum accumulation but checking whether a parameter or set of parameters lies between defined lower and upper limits (windows, conditions). An analyzer may have several associated windows, each of which is tested whenever a parameter or set of parameters is 'analyzed' by it. The results of these checks are stored as yes/no information in the analyzer itself and remain valid until the next event is analyzed. An analyzer is to be thought of as a fully passive element which makes no decisions concerning program execution. These decisions are made by the program itself on the basis of the information stored in the analyzer.

III. THE SYSTEM INDEPENDENT MACRO PROGRAMMING LANGUAGE FOR EDAS (SIMPLE)

The implementation of these concepts demands the use of a high-level programming language. No suitable common language exists on the two EDAS computer types (PDP-11 and IBM/370); therefore the structurally similar languages PL-11 (12) and PL/I (13) were chosen as the basis languages for user programs. Both languages offer far greater latitude than FORTRAN does.

Since large parts of an analysis programm consist of the functions described above, these can be standardized and implemented in two ways: with a subroutine library or through the creation of new language elements.

The first method would be simple but the lower execution speed is a drawback. In addition, the user would notice the differences between PL/I and PL-11 more than he would be able to take advantage of their similarities.

New language elements for special functions are most easily implemented by means of Macros, i.e. keywords possibly accompanied by an argument list. These Macros are expanded by a Macro preprocessor into legal code (PL/I or PL-11), which can then be fed into the respective compiler. Macros make possible the automatic generation of extensive blocks of code via simple statements.

The PL/I compiler is equipped with a suitable preprocessor. Since this preprocessor is practically language-independent, it is able to generate PL-11 code as well as PL/I code and originally formed the basis of the unified problem-oriented analysis language for EDAS. A SPITBOL version (14) of the preprocessor has now replaced the PL/I version, reducing compile times by a factor of about 30.

Macros have been developed for all basic analysis functions, using the analyzer concept as the foundation. Two Macro libraries and their corresponding compilers, PL/I for the SATAN subsystem and PL-11 for the GOLDA subsystem, are available on the IBM/370, where both user and systems program development take place.

Figure 1 shows an analysis program for a simple experiment, which could be used in either subsystem. The Macros $GOLDA and $SATAN designate code relevant to the corresponding subsystem only.

IV. THE IMPLEMENTATION OF THE EDAS SUBSYSTEMS

GOLDA

The Monitor RSX-11G, developed at the GSI from the DEC system RSX-11A, is the basis of GOLDA. It is a fast, priority-driven operating system under which up to 128 tasks may run. Both the Memory Management option and the use of overlays are supported. The PL-11 compiler, developed at CERN, was slightly modified and installed on the IBM/370. Systems programs for GOLDA are written in either PDP-11 Macro Assembler or PL-11. CAMAC (15) provides the interface to the experimental electronics, which enables data acquisition and experiment control to be carried out.

Together with the GSI Nuclear Electronics group the following standard data acquisition modes were developed:

> Random Scan Mode (RASMO)
>
> Several ADC's are defined as a group. An event is triggered by whichever ADC first receives data. After a variable coincidence time (from one to 100 microseconds) all ADC's as well as a pattern unit are read out via DMA into the computer. The pattern unit indicates which ADC's have received valid data.
>
> Multiscaling
>
> Some special applications require that the data acquisition be coupled to the instrumental parameters (current in a Beta-spectrometer, for example). GOLDA contains special functions to deal with this problem.
>
> Direct Memory Increment (DMI)
>
> Parallel to the above modes up to 8 singles spectra may be accumulated directly in the memory of the PDP-11/45 at a total count rate of 80kHz.

The ADC values are either hardware-accumulated or written event-by-event onto magnetic tape. The event data may also be asynchronously analyzed by a user-written analysis program, if time should remain after the I/O work. The control of data flow, analysis and graphical display of the data are accomplished by typing commands at a terminal. Some useful standard routines provided in GOLDA include a monitoring package for peak position and width. This function and also a mass calculation package are both supported by Macros and commands.

SATAN

In order to take advantage of the interactive possibilities of time-sharing, the EDAS subsystem SATAN normally runs as a user program under the TSO option of the MVS(OS/VS2) operating system on the IBM/370-168. In addition, standard analyses may be executed as batch jobs.

With few exceptions all system components of SATAN are written in PL/I, which is well adapted to the problem at hand:

> Multitasking facilities are supported, allowing asynchronous operation of analysis and I/O;
>
> The in-line code produced by the optimizing compiler is very fast;

The interlanguage facility of PL/I makes it easy to include user-written FORTRAN subroutines.

Reflecting the capabilities of the IBM/370-168, a number of functions are available in SATAN which cannot be included in GOLDA. These include, for example, various two-dimensional displays, plotting, peak fitting and interfaces for fetching spectra dumped by multi-channel analyzers. The installation of the mass storage system IBM 3850 (MSS) with a capacity of 36 Gigabytes has opened new vistas for spectrum storage and retrieval. With the help of the VSAM access method a ´spectrum bank´ has been established which allows direct and fast fetching of an almost unlimited number of spectra.

V. THE USE OF EDAS IN PRACTICE

User programs for both subsystems are developed on the IBM/370-168. First, the user writes and compiles the GOLDA analysis program. Compilation can take place either under TSO or as a batch job using the GOLDA Macro library and the PL-11 compiler. The resultant object modules are then transfered to a PDP-11 and linked into the rest of the GOLDA system. A user load-module is produced on disk which can be stored and reused. During a run in stand-alone mode, data is written onto 800 bpi magnetic tapes. After the run the tapes are reformatted, and the data blocked and compressed to 6250 bpi on the IBM/370-168.

For off-line analysis either the same or a more detailed program is retranslated, this time using the SATAN Macro library and the PL/I compiler, and linked to the SATAN subsystem. Data can be analyzed either from tape or, so as not to monopolize tape stations, from disk. A standard procedure is available to copy specified datasets from tape to temporary disk storage, where the datasets remain until the day after they have not been used. The total capacity of the temporary disks is about 100 tapes (800 bpi), which provides sufficient storage space for everyone and keeps the tape stations free for other work.

The Coupling of the Two Subsystems .

In stand-alone mode, magnetic tapes written during a GOLDA run must be reformatted before being analyzed under SATAN because of the different hardware representation of binary information on the tapes. This procedure is as automated as possible, but, nevertheless, slows down the process of data analysis. The data transfer rate, CPU and memory limits of the PDP are also noticeable. The fast data link (up to 120 kByte/s) by which the two subsystems are connected removes these limitations.

VI. THE PRESENT STATE OF EDAS

EDAS has undergone considerable development and has improved steadily since the first preliminary version through interaction with the experimenters, and has become the standard data analysis procedure at the GSI. The numerous additions and improvements have shown that EDAS is adaptable and flexible, and will be capable of satisfying the demands of the users for some time to come.

There are 30 groups using GOLDA, each with several different programs. Normally, two PDP-11´s are occupied with data taking (primary and parasitic beam experiments), the other two either being set up for the following runs or used for post-run calibrations. During the time from July 1977 to June 1978, 60 Gigabytes of data were collected in 3000 beam hours; the average data rate is thus 2,800 parameters per beam second.

About half of the 700 SATAN sessions per month are taken up with CPU-intensive list-mode data processing, the other half with display-bound spectrum handling. Total CPU usage amounts to about 75 hours per month for all SATAN users (approximately 60 users per month). In August 1978 there were 42,000 spectra in the ´spectrum bank´; these 250 Megabytes of data occupy two and a half virtual disk volumes in the mass storage system and are all accessible in less than one minute.

With the installation of a second large processor in December 1978 (an IBM 3032), we expect SATAN usage to continue climbing. The amount of data accumulated increases constantly; we are confident, however, that we will be in a position to cope with future demands placed on the system.

```
%INCLUDE $MACRO($AMAC1);/* standard prologue*/
 $LISTPROC;             /* for RASMO data   */
%INCLUDE $MACRO($AMAC2);
$GOLDA(INTEGER ETOT;);  /* definition - ETOT*/
$SATAN(DECLARE ETOT FIXED BIN;);
$AGEN(DELTA) NCND(3);   /* define DELTA     */
                        /*  (1024 channels  */
                        /* 3 conditions)    */
$AGEN(ENERGY);          /* define ENERGY    */
                        /*   (default=1024  */
                        /* ch.,0 conditions)*/
$AGEN(ETOTAL(3)) LIMITS(0,2047);
   /* 3 identical ETOTAL, 0-2047 chan.*/
$ANTRY;                 /* end definitions, */
                        /* start of analysis*/
$EVENT(PATTERN,DE,E);   /* input parameters */
                        /* PATTERN = RASMO  */
                        /*  bit pattern     */
IF $BIT(DE) THEN DO;    /* check validity   */
                        /*  of DE parameter */
   $ANAL(DELTA,DE);     /* check conditions */
                        /* accumulate DELTA */
                        /* spectrum         */
   IF $BIT(E) THEN DO;  /* check validity   */
                        /* of  E parameter  */
      $ANAL(ENERGY,E);  /* E -> ENERGY      */

     /* calculate ETOT */

      $GOLDA(E=>ETOT+DE;);
      $SATAN(ETOT=E+DE;);
      $LOOP(I,1,3);     /* loop over 3 cond.*/

    /* look at condition "I" on DELTA */

         IF $AC(DELTA,I) THEN

    /* analyze ETOT into ETOTAL(I) if ok */

            $ANAL(ETOTAL(I),ETOT);
      $ENDLOOP;
   END;
END;                    /* end 'DO' groups  */
$ENDEVT;                /* end of this event*/
$ENDANL;                /* end of subroutine*/
```

Figure 1: Analysis Program for a Simple Telescope (Macro Names start with a dollar sign)

REFERENCES

1) H. K. Carter et al., Nucl. Instr. and Meth. 139(1976)349.

2) C. Lengauer and A. Silverman, Nucl. Instr. and Meth. 140(1977)577.

3) L. Harms-Ringdahl and J. Sztarkier, Nucl. Instr. and Meth. 108(1973)557.

4) R. Mueller et al., Nucl. Instr. and Meth. 104(1972)125.

5) D. Lowenstein, Nucl. Instr. and Meth. 109(1973) 589.

6) S.L. Blatt et al., Nucl. Instr. and Meth. 111(1973)53.

7) B. Ghidini et al., Nucl. Instr. and Meth. 140(1977) 149.

8) B. Mertens and V. Kaltenbach, Nucl. Instr. and Meth. 113(1973)125.

9) M.P.H. Davies et al., Nucl. Instr. and Meth. 122(1974) 451.

10) H. Gelernter et al., Nucl. Instr. and Meth. 54(1967) 77.
J. Birnbaum and H. Gelernter, IEEE Trans. Nucl. Sci. NS-15(1969) 147.
J. Birnbaum and M.W. Sachs, Physics Today, July 1968, p. 43.

11) EDAS User's Guide, Volumes I to VI, R.P. Chestnut et al., GSI Darmstadt, October 1978.

12) R. D. Russell, PL-11: a programming language for the DEC PDP-11 computer, CERN Yellow Report,74-24(Dec. 1974)

13) OS PL/I Checkout and Optimizing Compilers: Language Reference Manual, GC33-0009-4, IBM, October 1976.

14) Robert B. K. Dewar, SPITBOL, Version 2.0, Illinois Institute of Technology, February 1971.

15) CAMAC: A modular instrumentation for data handling, EUR 4100e(1969)