DATA ANALYSIS SOFTWARE ON THE FTU Experiment and Its Recent Developments

G. Bracco, G. Buceti, A. Imparato, M. Panella, S. Podda, G. B. Righetti, O. Tudisco, V. Zanza

Associazione EURATOM-ENEA sulla Fusione CRE
Frascati, CP 65
I-00044 Frascati (Roma) ITALY

First Workshop of the IAEA Technical Committee Meeting on Data Acquisition and Management for Fusion Research
July 22-24, 1997, IPP Garching, Germany
MOTIVATION

Data analysis is the main activity of the experimental physicists working on a large fusion device producing a large amount of data during its operation.

Most of the data analysis on the FTU (Frascati Tokamak Upgrade, high magnetic field, medium size tokamak) experiment is presently performed using a general purpose data analysis and display program, named SHOW.

SHOW has been developed over the years to answer to the needs of the experimentalists working on various fusion devices (FT, FTU, a version also at JET, in connection with the NPA diagnostics).

The present SHOW implementation reflects the hardware where it has been developed, an IBM mainframe running MVS/ESA.

Due to the trend to migrate to a distributed system of workstations a preliminary version of SHOW has been recently ported to UNIX (Dec Alpha).
MOTIVATION cont.

The migration to UNIX has triggered the preparation of an extended documentation of the program and a review of its structure.

This presentation results from this activity and concentrates on two main items:

• The summary of what an experimentalist requires from a data display and analysis software.

• The importance of the flexibility of the software to follow the changing requirements of the data analysis activity.
OUTLINE

- Computation hardware in ENEA Frascati, related to FTU data analysis.
- FTU databases.
- SHOW program facilities.
- Program structure and system constraints.
- Examples of the SHOW facilities.
- Monitoring the program utilisation.
- Migration to UNIX.
- Conclusions.
HARDWARE for DATA ANALYSIS

Central computer centre

- **IBM 9672-R21** MVS/ESA, 256 MB (120 MB for scientific users) 16 Gb + 12 Gb (FTU databases), Automated Tape Library (Interactive and batch data analysis).

- **Dec Alpha 2100 5/250** Dec UNIX 4.0B 348 MB 18 Gb (interactive data analysis is starting now)

- **IBM SP2** 16 knots x 128 Mb 2 Gb each (mostly plasma physics theory).

Fusion department

- **VAX 4000/100** VMS 6.1 64 Mb (FTU data acquisition, CAMAC).

- **3 Dec Alpha** Dec UNIX 3.2 3000/800 256 Mb (FTU control system).

>>>> M. Panella poster 23/7/1997 <<<<
"A commercial real-time manufacturing integration platform for the new control system on FTU"

- **IBM Risc 6000/370** 128 Mb, 3 Gb (mostly plasma physics theory).

- **Desktop Macintoshes** PowerPC, 68030, 68040 (Office applications, terminal emulation, scientific computing)
FTU databases

The Pulse databases are originated from the control & diagnostic systems and have two keys, the pulse number and the channel name.

The Global databases contain the relevant information extracted from many records of the pulse databases and consist essentially of tables of data. They can also be used as indexes of the pulse databases. Presently the FTU global databases are contained in text files as tables of data.

All FTU pulse databases are located on the disk system of the IBM mainframe, where an Automated Tape Library is available.

A client/server software has been developed to access the pulse databases from UNIX workstations connected on the local network.

Three types of pulse databases are available for FTU:

DAS database, PED database and the Elaborated Data facility.
FTU databases cont.

1) DAS (Data Acquisition System)

800 channels/pulse

10 Mb/pulse

directly originated from the control & diagnostic systems: the content of each record does not change after the initial creation.

Direct-access unformatted datasets, each channel has a header containing the complete information required to reconstruct the data relevant for the experimentalist.

The name of the channel consists of the "family name" (diagnostic) & the data name:

NPA_9E.C01 Count rate of the first channel of the NPA versus time

It can also contain a two dimensional data structure

PHA_SX.MCA2 Soft-X rays spectra versus energy and time
FTU databases, cont.

2) PED (Post-pulse Elaboration Database)

170 channels/pulse

1 MB/pulse

generated by the codes running on the IBM mainframe: the content of each record must be easily updated.

channels are members of a partitioned dataset identified by the pulse number.

The name of the channel begins with $:

$E\text{QERAX}$ Major radius of the plasma magnetic axis versus time

It can also contain a two (or more) dimensional data structure:

$E\text{CMSVF}$ Electron Cyclotron Emission spectra as a function of frequency and time

or a family of data with the same abscissa values:

$L\text{ANGM1}$ Edge Density, Temperature, Potential from the Langmuir probe number 1
FTU databases cont.

A third kind of database is the Elaborated Data facility. In this case the channel name identifies the recipe to evaluate the returned data.

The database consists of a collection of channel definitions, stored as members of a partitioned dataset.

The definition relies on a simple interpreter language that can access the DAS and PED databases and also suitably formatted text files. In addition arguments can be provided by the user when the channel is invoked.

The channel name begins with %E.

700 channels are available

An example:

%E.QCYL Cylindrical q value, with fixed plasma geometry, versus time

definition: A='ZZZZED.IPL' B='EQLB00.MAGNET' C='(ABS(1.05E02*B/A))' X_VALUE='X' Y_VALUE='C' Y_LABEL='Cylindrical_q'
FTU databases cont.

The advantages of the Elaborated Data facility are:

• Reduction of the required disk space, avoiding the duplication of very similar data in PED databases.

• Prompt definition of new channels according to the experimentalist needs, immediately available for all the pulses for which the related DAS and PED data are available.

• User friendly access to the software packages of various diagnostic systems that are incorporated in the SHOW program. See later.

Some of their features:

Being a text file, the %E channel definition can contain an easy accessible documentation for the returned data, as comment cards.

The same %E channel can have different definition for different pulse ranges.

The possibility to create 3D data structure from several one-dimensional DAS channels having the same abscissa values.
SHOW program facilities

- Database read-out.

- 2D Plots.

- Command language utility.

- Computation utility. A simple interpreter package provides the possibility to perform algebraic computation between data and includes the usual functions and operators (new ones can be easily added).

- VERSUS spread-sheet (plots, fits, computation)

- 3D and contour plots

- Time series analysis

- I/O to files
  The SHOW program can read/write text files to export the data to other software utilities (e.g., SAS). Records in the PED database can be written too.

- PDF/ISPF environment can be invoked without exiting from the program
SHOW program structure

Hardware and software constraints:

Users access the data from 3270 terminals (or emulators) managed using the GDDM package; interactive session are run under TSO and ISPF/PDF facilities; the programming language is FORTRAN VS 2.5 (IBM FORTRAN 77).

Program architecture:

All the data read by the user are stored in a dynamic common area, called **SDA** (SHOW DATA AREA); max. number of channel data is 20, max. size of a single channel data is 65536, total SDA size is 163820x4 bytes.

2-D & 3-D plots are generated by a plot package written mostly in ENEA Frascati, ported to several systems from old HP computers to Norsk Data systems, VAX-VMS, IBM-MVS & VM.., recently Digital UNIX; a single routine for the interface with the hardware makes the porting rather easy.

The interactive user interface is based on panels and is optimised for a 32x80 character screen but also other screen size are supported. The functional keys **Fn** are used to produce the plots, to scroll the data list, to access the on-line help, to navigate between different panels.
SHOW program structure cont.

The batch user can access most of the program facilities, using the same command language that is used to automate a sequence of interactive commands. The batch mode is typically used for the analysis of a large number of plasma pulses, that requires to plot the same data for every pulse of the series or to build a table of values at given times for the same series.

- The program includes also some of the codes of most of the FTU diagnostic systems, developed and maintained under the responsibility of each diagnostic group.

These diagnostic packages provide data to the SHOW program using a standard interface structure, for which a symbolic calling sequence is available.

Presently 28 different packages and utilities can be accessed. This feature permits to easily add new facilities to the program.

Note: it may be that this last approach is suitable only for a medium size experimental group (30 regular users of the program in our case) at least in the present implementation.
SHOW program structure cont.

Example of a diagnostic specific package:

The BF3 neutron counters

The interface routine is BF3GET

SUBROUTINE
  BF3GET(NSHOT,IWHAT,VAL,X,Y,N,XL,YL,XU,YU,IRC)

IWHAT=9 means that the returned data values contain the peak ion temperature, versus time, for a given temperature profile.

VAL(3)=2 parameter defining the peaking factor of the ion temperature profile.

The symbolic calling sequence: the user access the data with the channel name: &BF3NEU0009(,,2)

To avoid the user to remember this sequence an Elaborated Data channel has been created: %E.BF3TI

%E.BF3TI * comments and description
  A=&BF3NEU0009(*,*,*)
  X_VALUE='X'
  Y_VALUE='A'

and the user invokes it as %E.BF3TI(,,2)
SHOW program structure cont.

The main advantages of the inclusion of the diagnostic packages in the SHOW program:

• a reduction of the size of the PED databases.

• the possibility for the user to change some of the control parameters for the interpretation of the data resulting in a more complete analysis.

• the diagnostician can immediately compare the results of its diagnostics with all the others available in the databases, and does not need to develop his own plotting program.

The disadvantages:

• the possibility of unwanted interactions between different sections of the code, due to errors or misunderstandings by a relative large number of different programmers.

• the need to re-link the entire program whenever any of the routines is changed. This last problem could be avoided in principle by the use of a run-time library, but no attempt has been done for the moment.
SHOW database-readout panel

Features:
the same sequence of channel data can be read for a
different pulse, the same for a pulse sequence; the full list of
the available channel data can be obtained and a wild card
search is available;
SHOW 2D plot page

Features:
overlay, x & y error bars, log scales, data normalisation, various lines and marks, histogram like display, the cursor changes the grid limits and measures values or distances, polynomial fitting, the abscissa range can be automatically evaluated in relation of the abscissa values of one of the channel data...
SHOW Command file facility

The data read-out and plotting facilities can be invoked using a command file.

The command file that generates the present configuration can be created by SHOW itself.

Input parameters can be passes to the command file.
VERSUS spread-sheet plot page

Hugill diagram of FTU ohmic plasma

Some built-in commands:

range XM 2 2 only deuterium data
scan btax (3.5 4.5 6.5) markers identify $B_T$ values
wc shot (5000 5500) time (0.3 0.7) generates a command file to be used to access the pulse database to prepare a table of other plasma quantities.
SHOW 3D plot page

Electron Cyclotron Emission spectra versus time and frequency
SHOW time series plot page

Fourier analysis: auto correlation function of one of the signals from Mirnov coils

Box car analysis technique is also available
Monitoring program utilisation

The SHOW program writes a log file and its content covering more than 3 years have been analysed.

The program has been invoked 33500 times by 71 users in the observed period, mostly in interactive session (batch session are 9% of the total).

30 users have called the program more than 200 times each, while 11 batch users more than 20 times each.

There have been 960 different versions of the program, and this large number of versions is due to the fact that SHOW includes several diagnostic packages, that are frequently updated.

During the tokamak operation period the average number of daily invocation of the program has been 40±16, while during shut-down periods it decreases to 10±11.

The maximum number of contemporary users is in the range 15±17.

The average user produces 9 plots with sessions lasting 26 minutes (the copy of the plots on papers requires to exit from the program, at least on the 3270 terminals).
Migration to UNIX

A preliminary version of SHOW has been ported to UNIX (without the diagnostic specific packages).

The porting of the FORTRAN code has been straightforward, the only problem being the implementation of the user interface.

To minimise the effort it has been decided to maintain the same structure of the user interface, substituting the GDDM packages with freely available libraries.

- **panels**: the needed sub-set of GDDM routines has been emulated using **curses library**.

- **plot sections**: the interface with two different packages have been prepared:
  1) with **CERN HIGZ** package
  2) with **PGPLOT** package (T.J. Pearson, California Institute of Technology)

The advantage of this approach has been the rapidity of the porting process, but the obvious drawback is the under utilisation of the facilities provided by the UNIX environment.

The user interface design will be re-examined in the future.
Migration to UNIX cont.
Conclusions

SHOW program has satisfied the request of a rather large number of experimental physicists working in the nuclear fusion domain due to:

• the flexibility of the program:
  Elaborated Data facility, the inclusion of the diagnostic packages, an easily updatable interpreter, a simple command language.

• the optimisation of the user interface, in the limits of the available hardware, so that the minimum of redundant actions have to be performed by the user.

Some of the solutions adopted, at least in the present implementation, may be only suitable for a medium size experimental group.

The porting of the program to a more modern environment is in progress.

The target: migration to UNIX during spring 1998, a preliminary version works already.