The ELETTRA Control System Database

C. Bortolotto, P. Michelini
Sincrotrone Trieste
Padriciano 99
34012 Trieste - Italy

Abstract

The use of an RDBMS (Relational Data Base Management System) and the development of applications for storing ELETTRA operational parameters and the management of Control System configuration data are discussed. A description of the software architecture is given with regard to the design and implementation of data structure, data entry and data retrieval for machine control operations and the interface with the Data Acquisition System.

1. INTRODUCTION

The ELETTRA Control System [1], which controls and monitors about 8000 I/O points, is based on three computer levels connected by two types of networks. The *presentation level* consists of control room high performance workstations, while the *processing level* and the *equipment interface level* are made of VME based microprocessor systems which run the OS-9 operating system and are distributed all over the machine. These last two computer layers, the so called Local Process Computers (LPCs) and Equipment Interface Units (EIUs), exchange information via the MIL-1553B highway using a simple and reliable Command/Response protocol.

In order to manage the large amount of configuration parameters, device and I/O point characteristics and to allow the saving and restoring of the machine operational conditions, the use of a database program is mandatory: the Oracle RDBMS [2] has been chosen for Elettra and two specials applications have been developed, the *Configurator* and the *Machine File*.

2. DATABASE FEATURES

All database internal data are grouped into tables using special common data values, called *link-data*, to establish relationships among them.

Each database application can be implemented through SQL*Forms applications and Pro*C procedures.

SQL*Forms applications consist of a set of computer screen "fill-in-the-blanks" templates using a 'Motif-like' user interface. They principally allow a guided data entry and a fast information retrieval. Any data manipulation needs no detailed knowledge about the relational database and data structure configuration. The use of these facilities increases the reliability of the database internal data. In fact, constraint on-line verifications on new data and link-data presence checks are made during the insertion procedure while the link-data automatic refresh is provided during the update procedure. Moreover, data organization and maintenance are completely transparent to the application users which can concentrate only on the data processing.

Pro*C procedures, C language procedures including SQL (Structured Query Language) statements, are generally used to access and manipulate a large amount of internal data or to implement more complex actions.

3. THE CONFIGURATOR

This database application manages the information associated to each control system I/O point and to the LPC and EIU configuration.

External Pro*C procedures are used to extract suitable sub-sets of Oracle data and to copy them into external files creating the so called *working database*. Part of this data is processed by the presentation level programs in run-time, part downloaded and finally handled by the real time applications. This facility runs either automatically from the SQL*Forms application or externally, whenever needed.

3.1 The Data

Each Elettra control system I/O point is completely identified by its logical name which is composed by a family, member, action and mode string [3]. The correspondence between this string and the physical parameters (VME-bus address, channel offset, calibration parameters etc.) are kept into the database.

In the same way, the board hardware parameters, the board location, the physical and logical network addresses etc., form the database tables describing the LPC and EIU configuration.

Fig. 1. show the main database table features and the relationships among them.

3.2 The Configurator Application

The data entry is organized in hierarchical steps to guarantee consistency and to assure the relations among the database tables. At each step, new data is inserted in the corresponding table and the link-data is selected from lists extracted from the previously filled tables. For example, when we enter the controlled point description, first we have to provide the corresponding hardware and functional description, then collect all this information into the controlled point descriptor table by means of the specific link-data.

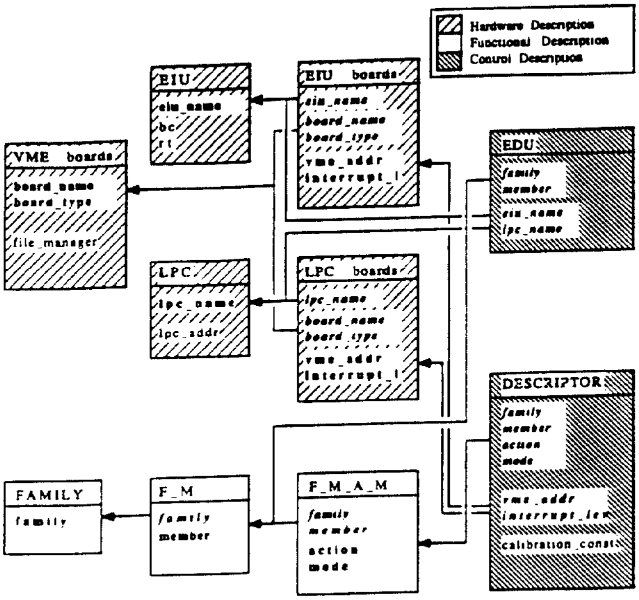

Figure 1. Configurator tables and link-data.

The steps, as shown in Fig. 2., can be grouped in the following logical layers:

Hardware Configuration : list of the VME boards, list of the EIU and the LPC logical names, link between each LPC and its VME boards, link between each EIU and its VME boards;

Functional Configuration : list of the FAMILIES, list of the MEMBERs associated to each FAMILY and list of ACTION MODEs associated to each couple of FAMILY MEMBER;

Control Configuration : link between each FAMILY MEMBER and its associated couple of EIU and LPC, link between each FAMILY MEMBER ACTION MODE and its hardware description.

3.3 The Working Database

The working database consists of the *descriptors* and the *controlled point configuration file* .

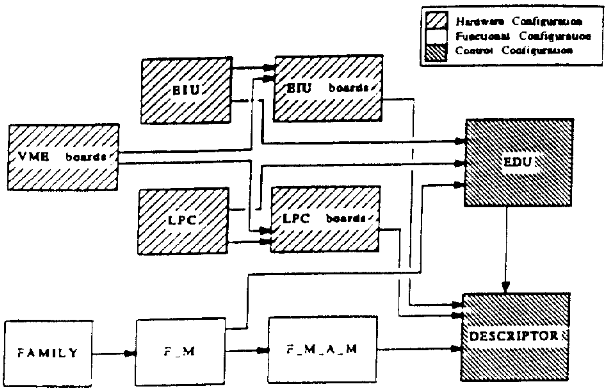

Figure 2. Configurator steps.

Each *descriptor* is a binary file representing the OS-9 device descriptor connected to a specific family, member, action, mode string. It contains all the parameters to operate that particular I/O point: interrupt level, hardware board address, calibration constants, etc.

The ELETTRA *controlled point configuration file* is an ASCII text file containing the list of the machine controlled points and, for each of them, the authorized operations, attributes and constraints. It has a particular format in order to be processed in run time by the Remote Procedure Calls (RPCs) accessing the field values.

4. THE MACHINE FILE

The *Machine File* application [4] is mainly used to save and restore the status of the ELETTRA machine (Linac, Transfer Line, Injection and Storage Ring). This is accomplished through LPC RPC [5] servers, which allow fast reading and setting for a particular machine configuration. Successive manipulations, like editing and add/merge facilities are available on the previously saved machine files. Delete, display, compare between two machine files and report generation have been also implemented.

All these operations can be run within the Machine File application by a sequence of forms integrated with Pro*C procedures. More specific actions have also been implemented writing ad-hoc Pro*C procedures.

4.1 The Data

The Machine File data are collected in several data blocks described in Tab. 1. General information like date, operator on shift, beam energy, optics identifier, mode/operation and comment about file contents, constitute the machine file header.

One database table is associated to the machine file header and there are as many associated tables as the data blocks are. A progressive number called Run Id uniquely identifies a machine file. Starting with a single record representing the machine file header we use the Run Id to retrieve every data block specific information, structured in one record for each data block element. A given Run Id value, belonging to different table records, specifies the "information of the same machine file", while it identifies "element of the same data block" internally to the same table.

4.2 The Machine File Application

When the application starts, a main menu shows the following choices :

Save : executes an external Pro*C procedure which, by RPCs, reads the element parameters of the selected data blocks and saves them, together with the header, in a database machine file;

Restore : recovers the machine operational condition stored in a database machine file. This can be the result of a previous saving or obtained from file manipulations (editing or add/merge). An external Pro*C procedure extracts the parameters to set and pass them to the specific server procedures;

Table 1.
The Oracle Machine File data design.

HEADER	DATA BLOCKS	PARAMETERS	No ELEM.
Date and time Operator on shift Beam energy Optics identifier Mode or operation Text comment	Linac		
	TL Bends Quads	setting reading status	23
	TL Correctors	setting reading status	34
	Injection	delay voltage setting	9
	SR Bends Quads Sextupoles	setting reading status	42
	SR Correctors	setting reading status	164
	SR Radio Frequency		
	SR Insertion Devices		
	SR Orbit	x y	96

Delete : erases a specific machine file;

Display : lists the parameters of all elements in each data block;

Edit : makes a copy of the specified machine file and updates the manipulated one;

Add/Merge : generates a new machine file as result of add/merge operations between two selected machine files;

Compare : checks the differences between two different machine files;

Reports : creates an external file containing all the element parameters.

5. CONCLUSIONS

The use of an RDBMS, the development of the Machine File application to save/restore the ELETTRA operational parameters and the Configurator application for the management of the Control System data configuration have been presented. The design description, the database table implementation, the software architecture and its interface with the control system have also been summarized.

SQL*Forms applications have been principally used in order to allow a guided data entry and fast information retrieval with no specific knowledge of the relational database. Moreover the data organization and maintenance are completely transparent to the application users.

Finally Pro*C procedures are used to access and manipulate a big amount of internal data or to implement more complex actions.

REFERENCES

[1] D.Bulfone, "Status and Prospects of the ELETTRA Control System", in the proceedings of the ICALEPCS, Berlin, Germany, 1993; to be published on Nucl. Instr. and Meth. in Phys. Res. A.

[2] "ORACLE RDBMS Database Administrator Guide", Version 6.0, Revised October 1990, ORACLE Corporation.

[3] M.C. Crowley-Milling, "Naming Conventions for the ELETTRA Components", Sincrotrone Trieste Technical Note ST/M - TN - 88/13.

[4] C. Bortolotto, "The ORACLE Machine File - User's Guide", Sincrotrone Trieste Technical Note ST/M - TN - 94/3.

[5] P. Andersen *et al.*, "User Guide to the Network Compiler Remote Procedure Call", LEP Controls Note 97, CERN 1989.

[6] D. Bulfone *et al.*, "The Design of the Energy Ramping System for ELETTRA", in these proceedings.