EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN - PS DIVISION

24

CERN LIBRARIES, GENEVA

CM-P00059547

CERN/PS 93-53 (CO)

829348

9

**THE OPERATOR INTERFACE TO THE EQUIPMENT OF THE CERN PS
ACCELERATORS**

J. Cuperus, F. di Maio, C.-H. Sicard

International Conference on Accelerator and Large Experimental Physics Control
Systems
Berlin, Germany, October 18-22, 1993

Geneva, Switzerland
15/11/93

THE OPERATOR INTERFACE TO THE EQUIPMENT OF THE CERN PS ACCELERATORS

J. Cuperus, F. Di Maio, C.H. Sicard
PS Division, CERN, CH1211 Geneva, Switzerland

A well defined equipment interface has been in use for a long time in the PS complex. It is based on standard calling sequences to a set of object-oriented software modules, which fit the diverse equipment in homogeneous classes. In the context of the new control system, with the introduction of UNIX, MOTIF, and an on-line database management system, this concept of standardization was extended to the workstation user interface. A single framework guides the operator through menus and options and provides specific control widgets and displays according to configuration data found in the database. This has proved very valuable for producing new control interfaces for whole accelerators with minimal programming effort.

This paper describes the current software architecture and also reviews the generic applications and commercial products for operator interaction which make use of that architecture.

1. Introduction

In the rejuvenated CERN PS control system [1], commercial workstations are replacing home-designed consoles and a new operator interface has been made for them. An important part of this interface is control of accelerator equipment, and this is implemented with software modules based on the concept of general purpose data-driven programs.

2. Functional Requirements

The main functions for accelerator control are:

(1) Individual control for any piece of equipment by means of standard tools; global commands, like "Stand-by" or "Send back Reference" for any subset of the equipment.

(2) Presentation of the equipment data into user-configurable tables. The data can be updated every machine cycle (1.2s) and the equipment data is interpreted, when necessary, by means of labels and colors.

(3) Availability of a reference set of the values for global commands and for comparisons.

(4) Transparent access to the equipment in the front-end computers (both old and new control systems). The communications and the network topology are transparent to the application programs.

3. Design Requirements

For efficiency and easy maintenance, the following constraints have been adopted:

(1) The software is data-driven: no description of the equipment or of its parameters are built into the software. The installation of a new piece of equipment or the modifications of its description should not require any program modification.

(2) All data are centrally managed and maintained by means of the Oracle DBMS.

(3) The network traffic is optimized by grouping transactions.

(4) Dedicated programs are implemented by end-users for all specific applications. All the necessary functions (e.g. equipment access) are provided by means of system libraries.

4. Software Architecture

The software is composed of 3 major modules: the equipment access, the operator interface generation and the database management. Fig. 1 illustrates the data flows between these modules, the users and the equipment.

4.1. The Equipment Access

The equipment access implements all the communication functions and provides a uniform interface to very diverse equipment [2].

The call parameters are: the equipment identifier with a class name and an element number (e.g. POW 3001); the action parameter ("property"), belonging to a fixed set (e.g. CCV, AQN, ...); the accelerator cycle type in the super-cycle; and the data exchanged, which is an array of simple type values (float, int, double...). An array call is

also possible for an array of element numbers in the same class, through the same action.

The equipment interface in the workstations provides the resolution from names to equipment identifiers and the distribution of array calls to the relevant front-end computers. Communications are implemented by means of RPC (remote procedure call) packages based on TCP or UDP.

The equipment access also acts as a "software bus" which integrates all the equipment-related software packages. Reference values, which are handled by a dedicated software package using Oracle, are accessible through special action parameters (e.g. CCV_DBREF). Synchronous acquisitions (i.e. synchronized on machine cycles), which are distributed from front-end computers into workstations' shared-memory segments, are also made available through special action parameters (e.g. AQN_MDR).

Fig. 1. Data Flow Diagram

4.2. The Operator Interface Generation

The "operator interface generation" module handles the requests from the operator, creates the user-interface, and manages the dialog between the operator and the equipment.

The operator interface is built in an automatic manner using two data stores: the operation configuration and the parameter description.

The operation configuration is maintained by the operation specialists. It contains the "working-sets" which describe the list of elements of a sub-system (like "Linac RF" or "Steering in the LTB line") and

the list of the parameters to display and control for each equipment class.

The parameter description is maintained by the equipment specialists and includes, for each equipment class, the name, data type, array dimension, patterns, labels, etc...

4.3. The Database Management

The database management module provides edition and distribution functions for the data of the other two modules.

All data are stored into Oracle tables. The database is centrally managed but equipment specialists and operation specialists are provided with dedicated edition environments with menus and forms.

The equipment and parameter descriptions, as well as the operation configurations, are distributed as read-only copies to redundant file servers and are made available by means of an efficient UNIX single key database system (dbm).

5. The Operator Interface

Application programs can be special purpose, for special instruments or high level control, or they can be generic. A large part of the operator interface is built with generic application programs and commercial tools which will be described below [3].

Fig. 2. Sample Operator Interface

5.1. Parameters displays

Parameter displays (rightmost window in Fig. 2.) are tables for the selected working-set. The usual layout includes for each element: its name, status, control value, acquisition value, units and reference value. The status data is interpreted with labels and colors. The colors encode a

general status information (enabled or disabled) as well as warning or error indications.

The data is updated on demand, time scheduled (every 4s), or every machine cycle.

5.2. Knobs

Knobs (bottom left window in Fig .2.) are interaction boxes providing the user with the visualization of an equipment element and means for changing it. All setting commands are built into pull-down menus and control values are managed with a dedicated widget supporting mouse or keyboard interaction.

5.3. Reference Values and Consistency Checks

Each control value of the equipment is duplicated in an Oracle table. These "reference values" can be updated with the current values or sent back to the equipment (either individually or in blocks). The color of control and acquisition values (green, yellow or red) reflects whether or not the value is consistent with the reference value within the required tolerances.

5.4. Equipment Logs

Equipment logs are produced in plain text or ANSI format for file archives or for printing (monochrome or color). They indicate values and settings of sets of equipment elements with a warning or error indication (this includes reference and acquisition checks).

5.5. Generic Measurement Programs

Generic measurement programs provide a uniform operator interface to various instances of the same class of instrument. These programs use the working-set structures in order to build in an automatic manner the measurement display, the instrument control and the connections to other measurement or correction elements. The following programs have been implemented: profiles with sem-grids, intensity with beam transformers (top left window in Fig. 2.) and trajectories with pick-ups.

5.6. Spreadsheet Applications

The equipment access has been integrated into a commercial spreadsheet (Wingz) so that spreadsheet applications can implement read or write access to any equipment parameter. This provides users with operation archives stored into spreadsheet files. These archives are either edited off-line or created from the

current values and they can be sent to the equipment in order to restore a particular situation.

5.7. Synoptic Applications

A part of the operator interface generation has been integrated into a commercial data visualization package (DataViews). Dynamic data links can be implemented in the synoptics, including the interpretation of the data with labels and colors. Some dynamic layout diagrams including interaction facilities are produced by this means.

6. Conclusion

The operator interface described here was a major development during the first three annual steps of the rejuvenation project. A big part of this interface has been used in operation since March 92. Additional functions have been added during the next 2 steps while the system was distributed to new parts of the complex (some consolidations are still under progress).

The user requirements were satisfied and the new operator interface is well accepted. The emphasis we put on the data management provides us with an environment where the software is largely independent of the equipment. Equipment additions and modifications are reflected in the data which are the responsibility of the equipment specialist.

Within the next two years of the rejuvenation project (94, 95), this operator interface will be extended to the whole PS complex while we hope to reach a low level in management and maintenance.

7. Acknowledgments

We would like to thank the many people who have contributed to this work, especially: J.M. Bouche and L. Merard for the equipment interface; M. Arruat, L. Cons, R. Hoh and A. Risso for the operator interface generation; M. Lelaizant for the database.

8. References

- [1] F. Perriollat, The New CERN PS Control System - Status and Overview, this Conference
- [2] L. Casalegno, J. Cuperus and C.H. Sicard, Process Equipment Data Organization in CERN PS Controls, Proceedings ICALEPCS, Vancouver, B.C., Canada October 1989.
- [3] F. Di Maio, Workstation as Consoles - User Manual, PS/CO/Note 93-26 (unpublished)