EE

EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN - SL DIVISION

CM-P00061192

CERN SL/93-42 (CO) /

su 9401

c2

The Data Exchange between LEP Machine and its Experiments

(Recent developments)

P. J. Brummer, CERN, Geneva, Switzerland

ABSTRACT

This paper describes the present data communications system between LEP machine and the LEP experiments.

The system should adapt easily to changes of experiments as well as to new acquisition methods and changes of the machine.

During 1991 and 1992, software quality and response times were improved. In order to cope with increasing data rates a new computer system has been installed. The need for continuing operation while the software is being adapted, imposed a phased approach to the changeover.

Some alternative methods of communication with experimenting groups are discussed and compared with the one presented here. In the future, fast data bases will play an increasingly important role in this activity.

ICALEPCS'93

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

THE DATA EXCHANGE BETWEEN LEP MACHINE AND ITS EXPERIMENTS. (Recent developments.)

P. J. Brummer.

CERN, Geneva.

September 30, 1993.

Abstract

This paper describes the present data communications system between LEP machine and the LEP experiments.

The system should adapt easily to changes of experiments as well as to new acquisition methods and changes of the machine.

During 1991 and 1992, software quality and response times were improved. In order to cope with increasing data rates a new computer system has been installed. The need for continuing operation while the software is being adapted, imposed a phased approach to the changeover.

Some alternative methods of communication with experimenting groups are discussed and compared with the one presented here. In the future, fast data bases will play an increasingly important role in this activity.

1 Introduction.

Long before the LEP machine was started it was realised that a system was needed which could provide information concerning machine parameters to the LEP experiments. This system should also be able to receive data from experiments such as background levels, radiation monitors etc. for display in the LEP control room.

The experiments of LEP are at a distance of up to several kilometers from the LEP control room, and a proper data communications method has been implemented.

The method used is one of fast transfer of fixed tables of topical information, from a computer in the control room to users' computers.

The communications medium is TCP/IP via the CERN ethernet and token-ring links. A remote procedure call protocol (RPC, [1], [2]) developed at CERN, which is now a standard for the accelerator, is used.

The data from the LEP machine is collected by routines reading the equipment modules and stored in shared memory tables. A set of interface routines, for use in remote computers of users, is made available as a library.

The data rates requested from this system have increased over time.

2 The Design.

2.1 Basic ideas.

The data exchange is based on data stored in the memory of a computer. Shared memory is used to provide read access for several programs at once. The data format in memory is the standard MOPS format [3], the service routines for MOPS data takes care of protection. This field of shared memory is divided in tables which contain data of different regions of the LEP machine. Each table can be accessed individually.

This set of tables can be seen as a database which aims to contain the 'present' status of the accelerator; the freshness of the data depends upon the equipment concerned but, is generally between 1/2 and 3 minutes. Timestamps are provided for groups of values.

Updating memory is done by a series of programs which read machine equipment status and set values. The following (non exhaustive) list gives an indication of the equipment included : bunch currents, beam energy, setup data such as nominal beam parameters and run number, radio-frequency parameters, currents of power supplies of main magnets, quadrupoles, correctors and other special magnets.

The tables are established in discussion with all users, and are regularly adapted to new needs. Changes to the table definitions are introduced at well defined times (e.g. shutdown of the machine) so that some

operational stability is obtained.

The RPC protocol defines clear tasks for the server (the computer with the data) and the client (users' programs reading or providing data). The software written for the data exchange can be divided into three parts: the data collecting software, the server software and the library software. These are all maintained by the data exchange service. The users write their own software for displaying and logging machine data, or sending experiment data, via the interface library.

2.2 Separation of systems.

An important feature of this system is the fact that the users are separated from the LEP controls system via the server-computer which acts like a buffer. By this arrangement they are shielded from changes of the controls system (which have been considerable in the last two years). In fact, the recent changeover of computer amounted to the change of a name in a file and the linking to a new library.

Another consideration is that although users can write tables with incorrect information, they cannot accidentally endanger the operation of the LEP controls system.

2.3 The format of a Table.

Each table is derived from one basic description file. A special program-set treats this description file to form C code files and include files. Also a FORTRAN include file is generated. These files are used by the users to generate their software, as well as by the data exchange system to generate memory maps and executable code.

At this time there are some 15 different tables in use.

All data in the tables form one structure representing that one table.

3 Operation and difficulties.

3.1 Behavior of the system.

The LEP machine started up in August 1989. During the first months of operation, the computer system and also the data exchange were still being modified and improved, which was the case of many systems of LEP. Systems were not very stable. Problems in the data exchange are very noticable, as the general machine display (Page101) depends on it.

The error rates of the data exchange never went down after this initial period. In fact, the computer on which the data exchange was implemented (eslpcr), got a bad reputation. It became common practice in the Control room to say: "when something does not work, restart eslpcr".

In order to improve matters, the data exchange was investigated in 1991. The following problems were found.

3.2 Reliability of coding.

As many different young programmers had worked on the system in sequence and as they all had some personal views and interests, it is no wonder that the various software parts did not match very well.

These authors suffered a certain lack of overview, and the consequences of this found were for instance: 1. The same data field being updated from different sources (causing jumping values). 2. Data being copied from one place to another, obscuring its real source. 3. Different styles of coding, going up to conflicting definitions. 4. The design of the exchange allows for the use of equipment names as variables, making programs rather folkloric. 5. Many bugs.

3.3 Problems with the computer system and communication package.

The server software suffered from a problem visible as 'timeouts' of connections made for data requests. Analysis revealed that there were 3 different causes: Occasional link closures or link overloads, restarts of the computer itself, and not quite explained error situations due to a locking problem between the RPC code and the computer system. The system in use is the XENIX SysV r2.3.1 which generally works well for the LEP control system, but this problem seems to be the exception. After many timeouts and error exits the system often decided to block entirely.

3.4 Increasing data-rates.

Once the system was in use, the physicists running the experiments had rather stringent requirements on update times and data rates. It became clear in the course of the study, that high data rates did enhance the timeout problems.

4 Remedies.

In order to improve the situation, a set of management measures were taken to study the causes and to improve reliability.

First the passing and copying of data was plotted in charts, which led to simplifications and to an overview of 'what is needed where'.

Thereupon all source code was studied, and where possible, simplified. A few simple coding rules were adopted and introduced, to make the code readable. Comments were added, etc.

All code was stored in a fixed directory arrangement, where partial and complete generation can be done efficiently.

The server software was studied and improved. This was a lengthy process, as certain effects became clear only when other problems were removed. Testing had to be done in a simulated environment, as the production machine could only be used for updates once tests were satisfactory.

Server spy outputs were analysed, and statistics made. In Appendix A some of this data is shown.

5 Introduction of a new computer system.

5.1 Reasons.

In order to cope with higher data rates and at the same time to get around the blocking problem, it was proposed to use a faster computer with a different system, to become the home for the data exchange.

An HP model 720 workstation with the system HPUX 08.07 was selected and ordered. Transfer speed measurements were done on various systems which indicated clearly the usefulness of this choice. The computer will be dedicated to the server software.

In Appendix B some of the measurements mentioned here are presented.

5.2 New data taking routines, and a phased implementation.

At the end of 1992, the LEP Controls group together with the LEP Operations group decided to introduce a new system for data gathering from LEP equipment. A central measurement server together with equipment modules (black boxes) will coordinate the data taking and distribution to user programs. This new measurement service was tried out at the end of 1992. It will be gradually made available for use in 1993, at the speed with which new equipment modules can be written and taken into service.

A consequence of this development is that the the whole set of data collecting programs of the data exchange needs to be rewritten.

It has therefore been decided to implement the new data exchange in two steps.

In a first phase, the users will access the data via the new server in the new computer and the old computer system will only do the data

collecting. Additional routines will transfer the data from the old to the new computer. The extra load due to these transfers on the new server is fully acceptable. In fig. 1. the layout of this system is presented.

To complicate matters, some facilities for obtaining data, like machine setting values such as nominal energy and operation mode, are not available anymore on the old systems, and can not yet be obtained on new (HPUX) systems. Therefore some data collecting software had to be planted temporarily on yet a third machine. (Not presented in fig 1.)

The second phase of implementation will cover the installation of all data collecting software on the new computer. This will probably take the rest of 1993 and part of 1994 to realise.

The new computer was installed in the second half of 1992, and tests of the server started immediately thereafter. The error rates are low and access speeds acceptable. It has been taken into production in March 1993.

6 Other Methods.

The simplest way to exchange data is by sending ASCII files. This is not a very optimised way of data transport as one will be sending characters, blanks and names and dimensions every time. This method is flexible, as changes only need to be made at the source, it assumes that the user only makes displays and does not need binary data.

The minimal amount of data is transferred in systems such as the one described here. Only binary numbers are sent, each table is a structure defined once, no unnecessary bytes are transported. The method is not very flexible. Changes have to be implemented by sender and receiver at the same time.

An intermediate solution could be to choose one structure as a standard for all tables, and to send the name information, (plus dimensions, boundaries, etc.) upon request. At starting time of the users' code, the name information would be requested and sent once, after which the binaries are defined and can be send as before. Once every hour the information request could be repeated. In this way, definition changes could be made during running.

7 A look at the future.

In the future our system is clearly developing in the direction of a database reader. In fig. 2. this situation is presented. The data exchange will collect and format the tables, which are then available for fast acquisition by the experiments. The data collecting will be more and more done from the measurement database and the logging database.

Also, a more flexible method for table definition changes could be envisaged.

The formatting of the tables, (composing them with SQL statements) is still a rather timeconsuming business, and it looks wiser to continue to do this in the exchange system rather than in each user system separately. The latter method would create vast duplication of work, even if it would be more flexible.

The faster databases and SQL searches will be in the future, the less there is a need for a data exchange, provided the database can cope with the load.

8 Conclusion.

Improvements have been made to the existing data exchange system at CERN, in the areas of software quality and in performance.

In the course of the investigation it became clear that the load on the system, which is the main factor for a design, had not been fully taken into account originally. However the choice of binary format for the data was correct from this point of view.

Flexibility in adapting to new circumstances is another important factor, and our system is not optimal in that respect.

9 References.

- [1] I.B. Kristiansen. Network Compiler Reference Manual. CERN, Geneva. LEP Controls Note 90. Nov. 24 1988.
- [2] P.S. Anderssen, V. Frammery, R. Wilkie. Tools for remote computing. Int. Conf. Accelerator and Large Exp. Phys. Controls systems. Vancouver, Canada. Oct. 30, 1989.
- [3] W. Herr. MOPS User Guide for 'C' programs. CERN, Geneva. CERN/SPS/ 88 - 43. (V. 3.10). May 2, 1991.

LEP experiment.

LEP machine-equipment.

Fig. 1. Current layout.

Drummer, ver10

LEP experiment.

LEP machine-equipment.

Fig. 2. Possible future stages.

Drummer, ver124

Appendix A. Error statistics.
Table A.1. Comparison of error rates.

computer	type	server version	period	Load c/min	mean error-rate err/hour	short term error-rate err/hour
eslpcr	ol.386	v4	July '92	124	45	240
eslpcr	ol.386	v5	Nov. '92	50	20	120
eslpcr	ol.386	v5	May. '93	29	22	50
breda	hp.720	v5	May. '93	110	8	15

Fig. A.1. Distribution of load of 9 users.

Appendix B. Computer performance.

Client computer	Server computer	server time msec/call	total transfer time. msec/call
sodev (486)	haag (69030)	43	180
haag (69030)	haag (69030)	34	50
haag (69030)	eslpcr (386)	14	60
sodev (486)	svlpcr (386)	14	60
rmspcr (386)	svlpcr (386)	14	60
haag (69030)	sodev (486)	5	60
sodev (486)	sodev (486)	8	110
haag (69030)	breda (720)	3.6	60
sodev (486)	breda (720)	3.6	60
hanoi (735)	breda (720)	3.6	16
breda (720)	breda (720)	3.6	8

Table B.1. Speed of server, and of total transfer.

(Results of 100 measurements)

Discussion.

In the column for server execution time it can be easily seen which computer is most appropriate for the server.

In those cases where the same computer was used for the client and the server (e.g. 'sodev' to 'sodev') the overall execution time is higher, due to the extra load. This effect is not even visible in the case of 'breda' where the CPU time is of the same order of magnitude as the transfer time for network hardware.

In certain cases the total transfer time is higher than one would expect. The explanation can be found in the rather special position of a computer in the network, -e.g. 'sodev' which is far from the control room - for which the routing is not optimised.

Response time of server
for
1 table transfer as seen
by remote user.

Key:

h haag (M69030) domain OS sys5.3
e eslpcr (386) Xenix
sv svlpcr (386) Xenix
s sodev (486) Xenix
b brenda (hp 720) HPUX

Fig. B. 1. Typical Service time of different computers.

Drummer, ber-08