Use of an INGRES database to implement the beam parameter management at GANIL.

Gillette Pascal, Lécorché Eric, Lermine Patrick, Maugeais Clotilde (Computers Group)
Leboucher Christian, Moscatello Marie-Hélène, Pain Pascal (Parameters Group)

GANIL BP5027 14021 CAEN France

e-mail : lecorche@ganexp.in2p3.fr

Since the beginning of the operation of the new Ganil control system in February 1993, the relational database management system (RDBMS) Ingres has been more and more widely used. The most significant application relying on the RDBMS is the new beam parameter management which has been entirely redesigned. It has been operational since the end of the machine shutdown in July this year.

After a short recall of the use of Ingres inside the control system, the paper first explains how the parameter management is organized. Then it shows the database implementation and how the physical aspects of the Ganil tuning have been integrated in such an environment.

Lastly, the paper tries to draw some outlines by showing more generally what advantages the use of the relational database brought in this context.

1. THE DATABASE INSIDE THE CONTROL SYSTEM.

1.1 The control system renewal.

From 1989 to December 1992 the Ganil control system was completely upgraded by replacing the previous and obsolete control system by a new architecture. The new system [1] relies on an Ethernet network onto which several kinds of processors are connected. VAX/VMS computers are used either for development purposes or as a control server at the real-time level. Each operator console consists of the logical association of an X-terminal and a workstation onto which is plugged a dial-box to implement the knob function. More than 2500 pieces of equipment have to be handled through front-end processors which are in VME or CAMAC crates and running the VAXELN operating system.

Most of the software is written in the ADA language and graphic developments rely on the Motif standard.

1.2 The relational database management system inside the control system.

Although it was a new technology in our environment, it was decided rather early in the project to build the data management for the new control system on a Relational Database Management System (RDBMS). Local considerations led us to choose the Ingres RDBMS for this and when the new system went into operation in February 1993, the RDBMS was already involved in many fields (ref. [2]) :

- The first important use was the equipment data management : software and hardware addresses, device scaling information, units etc. Files are extracted from the database and downloaded into the front-end crates or installed in shared memories in the workstations. For this application, access from the real-time level to the database is strictly limited to the update of trace flags for debugging and statistics.

- An other major part of the system built upon the database was the alarm logging system. Alarms are displayed on an X-terminal and VT consoles and at the same time they can be (if flagged) stored in the database for a short or long term period. This functionality is achieved by two processes communicating through mailboxes. A first process is in charge of the real-time level and display and it only interfaces the database after changes concerned with the alarm configuration; the second one directly stores the alarms or their acknowledgments in the database. This application constituted the first real-time application related to the RDBMS.

- Due to the capabilities brought by the RDBMS, it was also decided to manage basic operator menus in another Ingres database where tables are organized in a recursive way. Menus are loaded into the process memory when starting its execution and the coupling with the database is rather weak.

- An off-line application is in charge of the daily operation journaling. The database is filled by the operators without any interaction with the real-time level.

- The first version of the so-called "beam parameter database" BDPARAM was designed to store and manage the results from the off-line program PARAM which calculates the theoretical values of most of the machine devices according to the ion beam to be accelerated. The database was used as a gateway between the PARAM program and the on-line tuning programs which use files extracted from the database.

Graphic applications specific to the database are performed under the Ingres/Windows4GL environment. For the control software, integration of the ADA/SQL access is done through a preprocessor invoked before compiling.

1.3 First evolution.

The RDBMS was first applied to many aspects of the control system but apart from the control level. Due to the rising emergence of the relational technology in accelerator controls at that time and our lack of experience with such a system, we wanted to follow a very careful approach in this domain and as previously described direct links between the Ingres database and the on-line level were very restricted.

After the experience gained during the first year of operation with the new control system, the database was progressively integrated into the control level as we got a deeper knowledge of the RDBMS and as people came to appreciate the benefits brought by the RDBMS. The capabilities and features induced by such an approach are actually much more important than the interfacing response times found in most cases.

The most important development following this path is the new design for the beam parameter management which is now entirely built upon the BDPARAM database. The database has been widely extended and is accessed by on-line control applications as explained in this contribution.

2. THE BEAM PARAMETER DATABASE.

2.1 Ganil operation.

The Ganil can accelerate many different ion beams characterized by the beam specifications consisting of the particle to be accelerated, the ion charge at the source output and after the stripper, the beam energy, the RF frequency etc. As it is quite a complicated machine with three cyclotrons in cascade, the machine tuning involves sending the beam into several machine configurations, each of them with several possible beam optics for tuning or measurement. After the beam production and acceleration, it is adjusted and analyzed through a spectrometer before being sent to the experimental switchyard.

2.2 Machine description.

The design of the beam parameter database was first required to provide a complete description of the machine. So, all the entities associated to the beam parameters i.e. the pieces of equipment, beam characteristics etc. have been ordered according to their position along the beam path. Furthermore, these entities have been collected into objects sorted within an object-oriented approach by defining classes of entities: basic pieces of equipment (quadrupoles, dipoles, steerers, strippers, NMR probes etc.); more complex devices (RF systems, Magnetic spectrometers etc.); global machine components (cyclotrons etc.); beam properties (beam characteristics). Objects of each class have a fixed number of entities referred to by a predefined type. (For example, quadrupole objects have two entities of the "current" and "gradient" types.)

2.3 Database implementation.

The beam parameter database BDPARAM has to store all the values involved in the machine control for any part of the machine as managed by the object entity description. The aim is to provide on-line programs relying on the database and able to manage any machine configuration then to set all the pieces of equipment to the corresponding settings through the front-end crates; these on-line programs constitute the "PARAMETERS" application family. Beam parameters come either from the theoretical calculation (PARAM program) or previous settings. Settings for the beam lines are dynamically calculated by the PARAMETERS applications, as described below. The database is schematically represented in figure 1 and actually consists of more than 100 tables.

Each ion beam is specified by its BEAM_ID in the BEAM_DESCRIPTION table which is attached to a theoretical beam parameter set issued from the existing off-line program PARAM. This theoretical parameter set stored into the THEORETICAL_VALUES table contains the beam characteristics (the beam specifications and other beam parameters such as the magnetic rigidity etc.) and all the theoretical parameters independent of the beam course and optics (i.e. the cyclotrons).

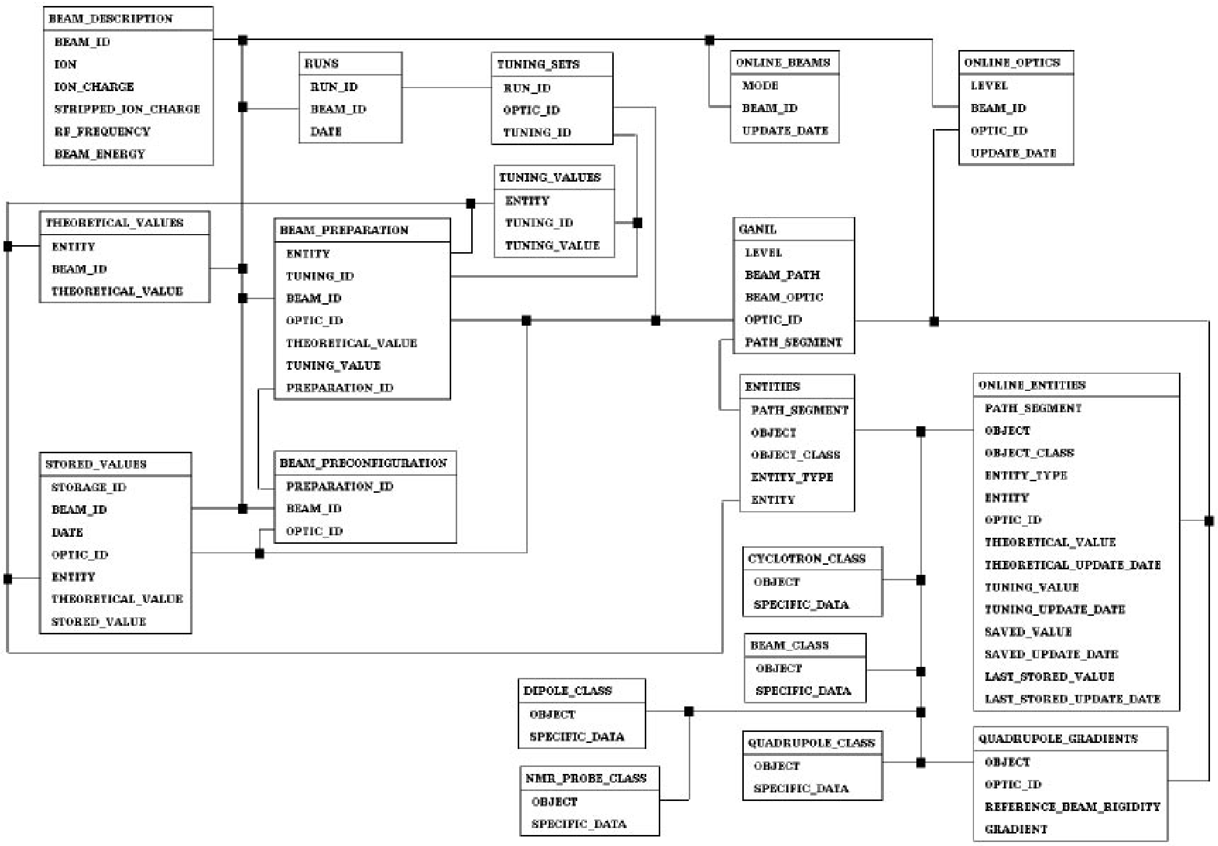

Fig 1 : Schematic overview of the beam parameter database.

The various tuning configurations achievable for both the GANIL machine and the experimental areas are listed into the GANIL table according to the BEAM_PATH with the corresponding BEAM_OPTIC optics configuration for each of them. Each beam path / beam optics couple belongs to an operational level of the Ganil facility; these levels are the "Machine", "Alpha spectrometer", "Medium Energy Output", "Experimental area distribution" and "Experimental Rooms". Any BEAM_PATH is divided into intrinsic segments named PATH_SEGMENT, each segment is defined as an indivisible element for the beam tuning and can belong to several beam paths.

The ENTITIES table then provides the complete description of the entities according to the object decomposition as described previously. Their adherence to the beam paths is seen through the PATH_SEGMENT attribute of the GANIL table. Object classes needing specific data for their complete definition involve the creation of particular tables (BEAM_CLASS, DIPOLE_CLASS, QUADRUPOLE_CLASS etc.). Also the QUADRUPOLE_GRADIENTS table had to be added defining the gradient value for each beam optic in which the quadrupole is involved according to the beam rigidity chosen as a reference.

Short term parameters set can be stored into the STORED_VALUES table ; they can be archived for a long term storage into the TUNING_VALUES table seen through the TUNING_SETS and RUNS tables.

Some tables are updated dynamically during the machine operation. First of all, the ONLINE_BEAMS table describes the beams currently produced both for the on-line and off-line modes, as one of the injector cyclotrons can deliver a "local" beam independently from the beam given to the physicists. The ONLINE_OPTICS table therefore specifies the beam path and beam optic currently applied to any of the machine levels. Lastly, the ONLINE_ENTITIES table is an on-line extension of the ENTITIES table used to keep dynamic object values.

3. MANAGEMENT OF THE BEAM PARAMETERS.

3.1 General overview of the data Flow diagram.

The beam parameter management is performed at two levels. Firstly, the off-line level consists of the parameter calculation by the PARAM program and the preparation of the machine operation. Then the on-line Motif programs PARAMETERS directly interface with the database and set pieces of equipment to the appropriate values obtained from the database management.

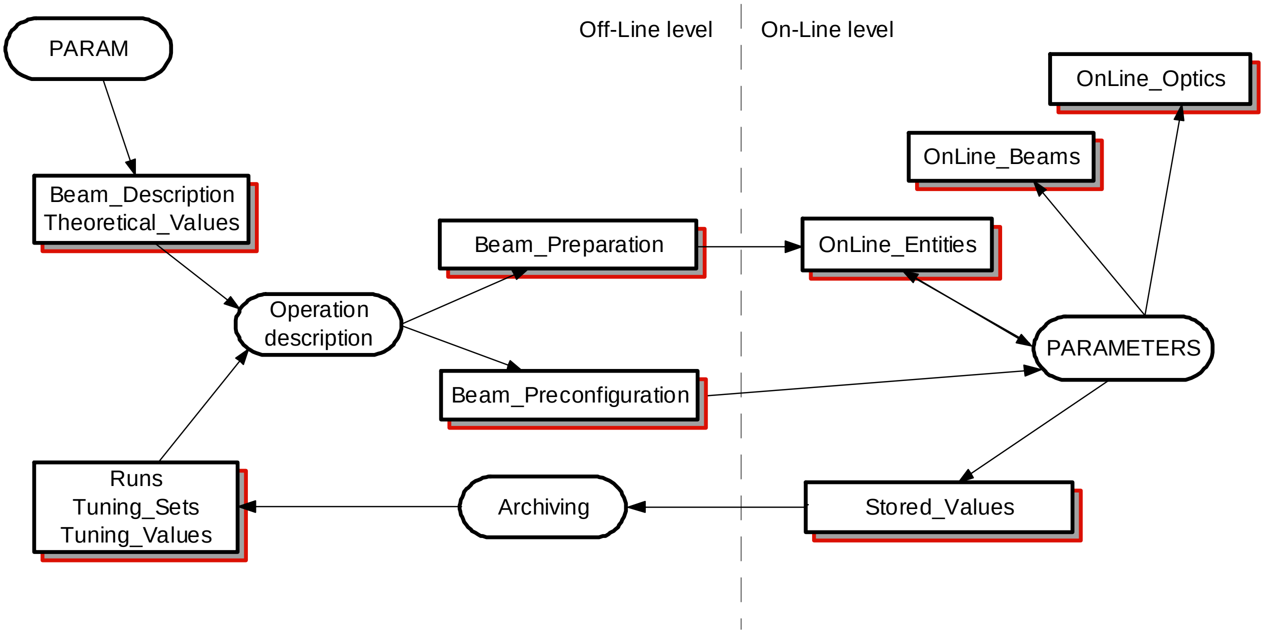

Fig 2 : Data flow diagram overview.

3.2 Off-line and on-line calculation.

The former FORTRAN off-line program PARAM calculates the theoretical values for any given beam specified by the ion to be accelerated, the energy required and the beam characteristics. SQL procedures load into the THEORETICAL_VALUES table only the theoretical values independent of the beam path and optic and not yet integrated inside the on-line program. All the beams known in the database are referenced in the BEAM_DESCRIPTION table are also updated from the PARAM program. The parameters concerning the beam lines, that is to say the optics parameters, are calculated on-line, all the data necessary for the calculation being in the database. The data flow diagram overview is presented in Figure 2.

3.3 Beam Preparation.

As the Ganil machine can accelerate a large variety of beams, it is necessary to prepare the beams which are going to be produced. So SQL operation description tools allow people to define beam parameter sets in the BEAM_PREPARATION table: first a "*basic set*" consisting only of the cyclotrons and beam parameters is generated from the theoretical tables independent of the beam path and optics to be applied later; then other "*dedicated sets*" attached to beam path / beam optic couples can be added, mixing the theoretical values issued from PARAM and archived values from the RUNS, TUNING_SETS and TUNING_VALUES tables.

The BEAM_PRECONFIGURATION table gives the on-line programs the basic information for the parameters set to be associated with a dedicated tuning mode.

3.4 On-line management.

The on-line management is done by the PARAMETERS applications allowing the operator to perform all the tuning phases concerning the machine configuration. First, when a beam has been chosen to be operated from the BEAM_PRECONFIGURATION table, the ONLINE_ENTITIES table is updated from the BEAM_PREPARATION table either for the theoretical or tuning values found in the basic set attached to the beam. The ONLINE_ENTITIES table is also updated after any optic change at any level of the machine : values can again be moved from the preparation tables (with the dedicated sets) but for the beam lines they also can be calculated dynamically. From this table, pieces of equipment can then be set to the theoretical, stored or archived values. The PARAMETERS applications also operate automatic beam rigidity adjustment either from current or stored values. Lastly, the scope of any action can be restricted graphically by the operator.

The current status of the whole machine is kept in the ONLINE_BEAMS and ONLINE_OPTICS tables.

The operator can also save any part of the machine in the ONLINE_ENTITIES table for a later recall or comparison purposes. Short-term storage is done into the STORED_VALUES table consisting of all the entities belonging to the beam path. They can be restored globally into the ONLINE_ENTITIES table or archived for further use using SQL archiving tools.

3.5 Software implementation.

The PARAMETERS applications are written in ADA and profit from the client-server architecture. Interaction with the operator is achieved through the Motif interface. The XRT/table graphic widget has been used for the data presentation included in the on-line applications.

Due to the internal multitasking capability of the language, ADA packages had to be first developed to interface to the Ingres database. They implement an internal database server for the application process to prevent concurrent multitasking access to the database, as the DBMS does not know the multitasking ADA specific features.

Most of the Ingres features have been used in the database design including database procedures (integrity rules, complex data management etc.), database rules (data integrity, coherence etc.) and database events raised on specific operation phases (beam or optics changes, modifications of the experimental area configuration).

In fact the database is duplicated on two different machines. The first version is used off-line level and comprises the complete database including the empty on-line tables to be used as backup if needed. The second one is an extraction from the first, consisting only of the data directly involved in the machine operation. Specific replication tools allow people to work on the off-line database and then to transfer their data in a secure way to the on-line level.

4. PRESENT STATUS AND FUTURE ENHANCEMENTS.

4.1 Status.

The whole beam parameter management system has been set into operation since the machine startup in July this year. From the operation point of view, it has already brought a higher reliability and more flexibility in the machine tunings; some specific experiments have benefited from the new capabilities offered. Also the group in charge of the beam parameters can now better cope with the increasing number of beams to manage, in a faster and more flexible manner than before. Moreover, the analysis of tuning parameters and statistics can be done more easily.

Some work is in progress to improve the whole system and new possibilities and features are being smoothly introduced, such as the propagation of the most important events dealing with the operation through asynchronous events in the database. Except when starting the on-line applications, response times are not a problem in this system. This aspect has been examined by looking carefully at the physical structure of the tables and the way ADA/SQL programming is done. We also use the statistics collected by the RDBMS to improve the optimizer efficiency.

Finally, access to the BDPARAM database through the ADA/SQL packages is beginning to be used in applications which had previously been unable to access any beam parameter values.

4.2 Extensions.

What has been done just constitutes the first and basic phase of the beam parameter management. In a second step, new tools or applications will be developed taking benefit of the database. At the off-line level, some tools for data analysis have to be defined as well as graphic tools for the beam preparation in a more convenient way. The on-line programs will progressively integrate the present off-line calculation: new object classes will therefore need to be created to follow this trend, the final objective being to be able to provide an autonomous on-line beam parameter calculation program.

4.3 Benefits of the database.

The beam parameter database is a typical illustration of the impact of a RDBMS in an accelerator control system. Looking at the benefits offered by such a system and its intrinsic properties (reliability, coherence, maintainability, methodology, global approach etc.), we can assume that the design of any control system should incorporate a database management system to provide all the required efficiency.

Considering that the challenge was not so obvious six years ago, the final conclusion could be that the use of Ingres inside the Ganil control system has proved to be a real success.

REFERENCES.

[1] L. David, E. Lécorché, C. Maugeais, M. Ulrich : "Controlling the GANIL from an Operator Workstation", Proc. Int. Conf. EPAC’94, London, Great-Britain 1994

[2] E. Lécorché, P. Lermine : "Database management in the new GANIL control system", Int. Conf. ICALEPCS’93, Berlin, Germany 1993