The Database System for the SPring-8 Storage Ring Control

A.Yamashita, T.Fukui, K.Kobayashi, T.Masuda, A.Taketani, R.Tanaka and T.Wada
SPring-8, Kamigori, Ako-gun, Hyogo 678-12,Japan

Abstract

The SPring-8 storage ring (SR) control system is built on a relational database management system. It contains all data needed by the SR control including static data and logging data. We also developed access methods in C functions and WWW.

1 Introduction

We have built a control system for the SPring-8 storage ring [1] fully based on a database system. All data needed for the SR operation and collected from the SR are stored on the database. The SR has 4386 equipment controlled by 28 VME computers distributed on the SR network [2] around the ring. Data acquisition programs collect over 5600 signals from equipment every 1 to 5 seconds. Programmers writing control and analysis programs need to access those data quickly and easily.

A consistent philosophy and methodology are required to manage such a large amount of data and fulfill programmer's requests.

We chose a relational database management system (RDBMS), Sybase SQL server system 11. It is not only a convenient way to store data but an unified, simple and fast way to access it. The database is designed along the relational database methodology and tables are normalized with a few exceptions.

We have built three kind of databases called parameter, on-line and archive databases. The parameter database manages the equipment, the computers, the signals, the on-line database tables and so on. The data collected from the equipment are logged on the on-line database which keep the data temporary (1 to 10 hours) for monitoring status of equipment of the SR. The archive database contains the data sampled from the on-line database and keeps them for a long period of time for the off-line analysis.

The C functions and the common gateway interface (cgi) programs provides access to the database for users who write operation programs or monitor signals.

2 Parameter database

The parameter database manages static part of database in 137 tables. The parameter database contains tables categorized as follows.

- Definition of the equipment and other information needed for the data acquisition from the equipment.
- The beam parameters and corresponding setting values for the equipment.
- The calibration data for the beam monitor equipment and the magnets.
- The alarm thresholds for the analog signals and reference bits for the digital signals.
- The data buffer for communication between processes.

The above tables are all closely related.

2.1 Equipments and data acquisition

We define the equipment as the core part of the parameter database. An object connected to a VME Crate is called an *equipment*. Every equipment is managed in one table independent from its nature. Other elements of the parameter database such as the signals, the CPU's, the on-line tables have relation to the equipment directly or indirectly.

The parameter database contains the information needed by the data acquisition processes such as the signal to be collected, the CPU to be asked, the on-line database table to be stored and so on.

The experts on equipment make spread sheet files which describe the equipment name, the signal name, the CPU name and the several parameters such as the hardware addresses of the different equipment or the calibration constants. The programs written in Python language [3] decompose them into normalized tables, maintaining their consistency.

2.2 Calibration constant

The calibration constants are stored according to their nature; some sets of constants have time stamps or version numbers as they need.

2.3 Alarm condition

For alarm condition see [4].

2.4 Beam parameter

The beam parameter tables provide a bridge from theoretical calculations to the real SR operations and a reproduction of previous run.

Experts of beam dynamics enter the beam parameters and the corresponding setting values for the equipment to the database. The calculated values are loaded to temporary table when they are needed for the SR operation. The control programs send the values in the temporary tables to the CPU's. When an operator changes setting values manually during a run, the values in the temporary tables are also changed. If the operator is satisfied with the set of values, he or she registers them from the temporary table to the parameter database as a new entry with logging information such as the date and the time. Operators can reproduce the beam parameters of any previous time by loading the data from the parameter database.

2.5 Communication buffer

The parameter database has another important role as a communication buffer between processes. Processes running on distributed CPU's can have the common data on the database. Some tables manage current status of the processes running on the distributed workstations or the equipment status of locking by the other process.

3 On-line database

The on-line database stores the current status of the SR read by the data acquisition programs and keep them for one to ten hours.

3.1 Data acquisition

The poller and collector [5] collects 5600 data from the equipment and enter them to 39 tables in the on-line database in one to 30 seconds cycle. The cycle are determined by the equipment's nature. For example, DCCT which monitors the SR's beam current is read every second while equipment of vacuum system is read every 3 seconds and the temperature of RF cavity is read every 30 seconds.

In addition to the periodical regular data acquisition, the alarm status is written to the database when an alarm occurs.

3.2 Data access

The programs monitoring the status of the SR extract the latest data from the on-line database instead of accessing the equipment directly. This scheme reduces the traffic of network and the load of CPUs attached to devices and enables software test without connecting to equipment.

3.3 Implementation

Speed for storing and retrieval is the primary design goal of the on-line database.

We pack as much data to one row to reduce the number of SQL statements. The RDBMS limits the number of data stored in a row to 250 columns. Thus, a process which collects data for one equipment group writes to several tables (up to 12, so far).

We have discovered that the time taken for one SQL is almost independent of the number of other transactions (up to 20 at the same time). It means that the sequential process is slower than the concurrent one, so that we have to write data to tables in parallel using asynchronous multithreaded functions built in ct libraries provided by Sybase. A 100 kb/s of writing speed was obtained with this method. It is good enough for the current data acquisition.

We use the tables like ring buffers and the rows are overwritten after a given period. We limit the number of rows in one table on the on-line database to 4000 for fast update and retrieval. The format for one data point are 4 byte integer or 4 byte floating value with one exception, integrated beam current which is 8 bytes floating point data.

Each row of the table contains a sequential number column and time column as keys to be accessed.

4 Archive database

The archive database permanently stores the data sampled from the on-line database.

The archive database has the identical structure to the on-line database except for the length of the row which has limit.

Processes for each equipment group sample the data from the on-line database to the archive database periodically, once an hour. For example, the DCCT data are reduced from 1 to 10 sec cycle and magnet current reduced from 5 to 30 sec. On the other hand, the data can be copied manually from the on-line database to the archive database with arbitrary period and sampling rate. In addition to the periodical entry, the alarm watch process and the closed orbit distortion (COD) measurement processes enter the data to the archive database.

The size of archive database is increasing at the rate of 1GB per month during regular SR operation. It is no problem because the RDBMS can manage up to TB scale.

The speed of data retrieval is acceptable using an index. One can see a graph of the DCCT data for one day which contains 8640 data points in less than 8 seconds from a Web browser. 20000 data sets of measurement for the COD, where one measurement consists of 576 data points, are retrieved in less than 2 minuts with C functions.

5 Data access functions

We provide two ways to access the databases for the programmers and the accelerator users. One is a set of C functions and another is cgi programs for the WWW.

The programmers use the C functions for the control and analysis software with UNIX, C and X-windows. One can look at a graph or download data of the on-line and/or archive database from the WWW browsers.

We have built over 300 C functions mostly for accessing the parameter database. The C functions hide SQL commands from the programmers. One C structure is assigned for each table of the parameter database that corresponds to it. The programmers obtain rows of a table with their selection criteria.

Users can access the on-line and/or the archive database with a small number of functions. Because of the identical structure of the on-line and the archive database, the programmers do not have to care in which database the data is stored. They can obtain the data by just specifying the human understandable signal name and period of the time they want or the signals that belong to the equipment group can be retrieved within a given time. The COD data are a special case. The programmers can obtain any number of the COD data set within a given period of time.

A set of cgi programs written in Python display the table of the signals on a Web browser. By clicking a hyperlink, users can see a graph drawn by gnuplot3.6 for any data in the on-line and the archive database. The graphs are drawn by user's request dynamically. Additionally, the cgi-program can display the data in text format with which, the user can download and process with their analysis program.

In order to distribute the status of the SR, a program periodically writes html files that display the beam current, the beam lifetime, operator's comments and a trend graph of the beam current. We use this system to provide the data throughout the site instead of a CATV system. 24 PC's distributed on site continuously display the SR status using the WWW browsers.

6 Management and operation experience

We have been running the database system on the Hewlett Packard J210 with two PA7200 120MHz CPU's, 512MB memory and fast-wide SCSI disks for over seven months. Important databases, such as the parameter and the archive databases are put in the 12GB RAID 0+1 disk. The on-line database is scattered on the four disks to reduce I/O time. Another J200 computer, which is identical to the main server except the CPU clock is 100MHz, is standing by for the backup and for off-line analysis.

The archive data size became over 6GB at the end of October 1997. The database is backed up once a day and it is restored to the backup system. The RDBMS in the main database server consumes about 60% of CPU power constantly.

Since the beginning of the commissioning, we had no trouble due to the RDBMS software except one disk trouble on the RAID and one operation failure. During the commissioning, the database flexibly has followed the addition, removal and replacement of the equipment and the equipment groups.

We began the commissioning with 4200 points of data and ended with 5600 points, taken at the end of commissioning.

7 Conclusion and future plan

The entire SR data is under management of the RDBMS. The relational database methodology provides simplicity for data access to the programmers, quick and easy access to the on-line and the archive databases to the users and the data consistency to the data manager. The database system has been accelerating the software development and the SR commissioning.

The data of beam lines and synchrotron are added on this database system. We have decided to replace the main server to K250 with four CPU's and 2GB memory for the future database expansion.

The ability for extension of the database has been proven and the database is growing.

Acknowledgments

We greatly appreciate members of Mitsui Zosen System Research Co. Ltd. for their assistance to build this system.

References

[1] R.Tanaka et al., in these proceedings of ICALEPCS '97, Beijing, China.

[2] T.Fukui et al., in these proceedings of ICALEPCS '97, Beijing, China.

[3] http://www.python.org/

[4] A.Yamashita et al., "Alarm system for Spring-8 storage ring.", in these proceedings of ICALEPCS '97, Beijing, China.

[5] A.Taketani et al., "Data Acquisition System with Database at the SPring-8 Storage Ring",in these proceedings of ICALEPCS '97, Beijing, China.