DATABASE OPERATION USING ODBC/JDBC IN THE KEK 8GeV LINAC

Masahiko Tanaka*, Isamu Abe, Hitoshi Kobayashi, and Shin-ichi Kurokawa
High Energy Accelerator Research Organization (KEK)
1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan
*Mitsubishi Electric System & Service Co., Ltd.

Abstract

The KEK 8GeV Linac (linear accelerator) has been using an operation database consisting of PCs (personal computers) and MS-SQL [1] on a Windows NT server. An automatic database logging system for Linac operation and attendant problems has been completed.

The Linac has both static and dynamic databases, which store accelerator data, a troubleshooting log and operator's notes [2,3]. The console is composed of an MS-Access integrated, and GUI (graphical user interface) and applications written in VB (visual basic) linked to the database using ODBC (open database connectivity).

Recently, the need for web-based applications linked to a database has grown [4], and JDBC (Java database connectivity) has enabled this. This treatise compares ODBC and JDBC, both of which are used in the Linac PC system, and reports the issue of their application.

INTRODUCTION

The creation and maintenance of an database requires a great deal of labor, far more than we initially expected, and not many have succeeded in developing a satisfactory database. However, the use of advanced PCs simplifies the building of a flexible, practical accelerator database system. Moreover, particularly in the case of the display function, by making applications web based, their operation becomes more generalized

The KEK LINAC accelerator consoles are composed of an MS-access integrated GUI and applications written in VB linked to a database using. Recently, the need to link with a database via a browser has grown, and JDBC lets the Web-based connection to do the job. This treatise, in addition to conventional ODBC, reports on the consolidation of JDBC, gives a comparison of ODBC and JDBC, and describes advances in their application.

1 DATABASE FOR ACCELERATOR OPERATION

Lately it has become possible to construct practical PC databases that can be applied easily to accelerators at very reasonable cost and attain high performance, as it is simple to secure an environment which can realize multi-CPU and multithread operations via a Windows NT server. For PC clients, the readiness of tools for database operation has also contributed to the construction of a flexible system. The KEK 8GeV/3.5GeV electronic positron Linac (linear accelerator) has been operating with an MS-SQL 6.5 database since 1995.

From the viewpoint of GUI, several methods have been studied to elucidate software development and productivity. The operational data of accelerators, including operational log, records of trouble and maintenance, accelerator and device data tables, and accelerating tube temperatures are given by MS-Access applications in VB using ODBC for compatibility with the Windows system. To date, we have focused on developing an environment in which the console can be operated via Windows, since it was felt that safety problems existed in accelerator operations in context with OS databases, except for Windows. However, after providing a viable environment for Java and JDBC, we started to develop software for multi-platform operations. Minimal necessary functions have been completed for displays of conventional factors using the Web browser.

2 CONVENTIONAL DATABASE FUNCTIONS

Currently, MS-SQL is used as a master database, one built and maintained by the MS-SQL Enterprise Manager.

The database is simple to use and has a plurality of functions. GUI windows were operated and maintained by tools written in MS-Access (Fig. 1) and VB (Fig. 2),

Fig. 1: Operation log (by MS-access with ODBC)

linked to a database using ODBC, since it is easy to construct GUI as the front end. This enables processing uncomplicated problems quickly on local computers having MS-Access. The large database is saved as a master DB to reduce the latter's load.

All accelerator operators memorized the displays and not only automatically but also manually input such recorded data as the daily operation log, trouble and maintenance. They also used accelerator operation search and display functions. Too, the screens serve effectively for retrieving operational information like data storage and statistics on accelerator use from device layers.

The above screen shows accelerator operators, via connection to servers and CGI (common gateway interface), using MS-Access and VB or Web-browser.

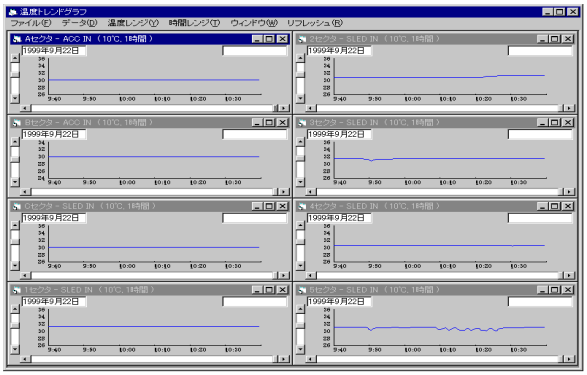

Fig. 2: Temperature date by ODBC, VB

A commercially available component of OCX (OLE custom controls) is adopted and embedded in a VB container. This makes it possible to reduce the load for development of original software and allows full use of convenient functions.

The conventional screen can be used only with MS-Windows, that is, solely on the control console. Still, the need for screens corresponding to multi-platform has grown owing to improvements in Java environment, etc. Accordingly, we have been promoting the introduction of JDBC and preparing a Java screen.

3 CHARACTERISTICS OF JDBC

Introducing Java as a language enables platform operation and using JDBC permits connections to several kinds of general databases. With visual basics as developing tools, it made links with databases easy.

Once JDBC is developed as a connection class, it can be applied as a tool for other types of development.

4 SYSTEM CONFIGURATION

The system comprises three layers:

4-1 First layer

For GUI clients, only the standard Web-browser is needed as an operational environment. This makes multi-platform display possible via a network without developing software for each platform. Conventionally, the screen shows programs written in VB and MS-Access, which limits its display to MS-Windows. But using Java results in a screen that can be accessed from anywhere.

4.2 Second layer

The dbANYWHERE server is used as middle ware. In this layer, a JDBC/ODBC connection process between clients and database servers was accomplished.

4.3 Third layer

This layer is a database server. Although the SQL 6.5 server is being used, it has not been upgraded to the latest version since doing so requires accelerator shutdown.

4.4 Functions

- Management of client server
- Client communication, command control, buffering
- As a database (Dynamic DB, Static DB, Archiver)
- Component ware for generalizing accelerator control
- COACK[5] (Component Oriented Accelerator Control Kernel)-II connection
- COM (Component Object Model) server connection
- Extraction and reuse of macro-operation command
- Sequential operations

These functions are possible to run from the first layer.

5 COMPARISON OF ODBC/JDBC

(1) Access speeds for recording in both ODBC and JDBC systems were compared. First-access speeds are given in the table below. In both, booting speed, database connection and access speed for recording were gauged. Access speed depended on the number of records. In this test, each access time for approximately 2,000 records in the SQL database was measured.

	1	2	3	4	5	Average
ODBC	480ms	170ms	161ms	160ms	170ms	228.2ms
JDBC	691ms	1172ms	1142ms	1172ms	942ms	1023.8ms

(2) Next, a comparison was made of records on search speeds in ODBC and JDBC. First-accessed database search time in front and behind and last-accessed records were measured, with the results shown below.

	1	2	3	4	5	Average
ODBC	0ms	0ms	0ms	0ms	0ms	0ms
JDBC	<10ms	0ms	<10ms	<10ms	<10ms	<8ms

These results make it clear that access speed using a database connection with JDBC was a bit slower in input operation compared with that using ODBC. But the difference in access speed between JDBC and ODBC is negligible, which confirmed JDBC applicability for accelerator operation.

6 APPLICATION SCREEN

Example of a screen corresponded to multi-platform is indicated as follows.

This is a search and display screen for the accelerator operation log, trouble and maintenance records, the use of which enables operation through platforms with OS, except for Windows. The introduction of COACK, as described in the next section, expedites operation by cutting down loading time. Since various database functions were prepared and the number of required software was reduced, GUI program size became smaller too.

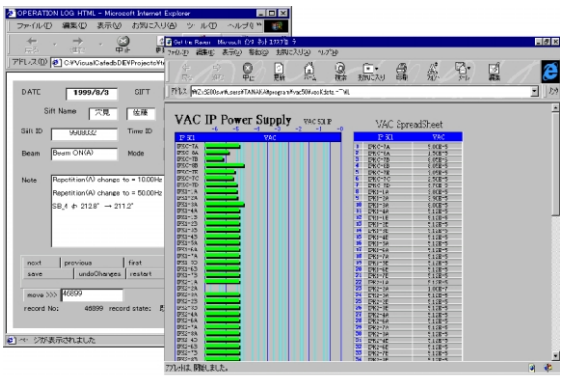

Fig.3: Operation log & VAC data

7 COACK-II CONNECTION

Kernel software for an accelerator has been developed at KEK. In COACK-II the functions required for accelerator operation were prepared as component ware. Next, the connection method was prepared. By applying this software, the client load in original software development will be less. But if database operation functions are generalized, chances for client direct database operation using ODBC/JDBC are also reduced. Consequently, the software for database operation is divided to components and stored in the DB and server.

8 FUTURE PLANS

We intend to develop a prototype control kernel for common use in all accelerators. However, we wish to aim at continuous development by verifying the connection of various components so as to establish an integration with sophisticated software called SCADA (supervisory control and data acquisition). Inherent parts of an accelerator that are not generalized and standardized (which in many cases are device drivers for each type of equipment) are in planning for fabrication in a second phase project in the development of a generalized control kernel.

SUMMARY

This study elucidated the fundamental characteristics of ODBC and JDBC, and that a wide operational environment can be realized. Compared with conventional GCI mode, it is easy to develop. There are possibilities to add heavy loads to servers and DB by increasing the number of JDBC platforms. Therefore, to facilitate easy operation in various platforms, the development of component ware for COACK is also being studied by reducing master database load through attaining generalized functions on the level of COACK component ware and effecting pretreatment. This method is similar to a way to reduce the load of master DB engines by scattering them in commensurate manner using MS-Access.

REFERENCES

[1] Database for accelerator operation support,
Masahiko Tanaka, Isamu Abe and etc.,
Linac conference ’97, Sendai, Japan

[2] Feedback of operators' experiences to console programs in the KEK e-/e+ Linac
Masahiko Tanaka, Isamu Abe and etc.,
ICALEPCS97 China

[3] Database system in the KEK LINAC PC based control
M. Tanaka, I. Abe and H. Kobayashi
PCaPAC ’99, Tsukuba, Japan

[4] GUI and I/O Interface for COACK-II
K. Nigorikawa, M. Mutoh, I. Abe, J. Kisiro,
T. Kosuge and S. Kurokawa
ICALEPCS99 Trieste, This conference

[5] COACK-II project on accelerator control kernel development
Isamu Abe, and etc.,
ICALEPCS99 Trieste, This conference