Journal of the Korean Nuclear Society
Volume 34, Number 6, pp.610～618, December, 2002

<Technical Note>

Conceptual Data Modeling on the KRR-1&2 Decommissioning Database

Hee-Seoung Park, Seung-Kook Park, Kune-Woo Lee, and Jin-Ho Park
Korea Atomic Energy Research Institute
150 Dukjin-dong, Yuseung-ku, Daejon 305-353, Korea
parkhs@kaeri.re.kr

(Receved July 25)

Abstract

A study of the conceptual data modeling to realize the decommissioning database on the KRR-1&2 was carried out. In this study, the current state of the abroad decommissioning database was investigated to make a reference of the database. A scope of the construction of decommissioning database has been set up based on user requirements. Then, a theory of the database construction was established and a scheme on the decommissioning information was classified. The facility information, work information, radioactive waste information, and radiological information dealing with the decommissioning database were extracted through interviews with an expert group and also decided upon the system configuration of the decommissioning database. A code which is composed of 17 bit was produced considering the construction, scheme and information. The results of the conceptual data modeling and the classification scheme will be used as basic data to create a prototype design of the decommissioning database.

Key Words : decommissioning database, ERD, client-server, relational DBMS

1. Introduction

The TRIGA research reactor D&D project team in KAERI (Korea Atomic Energy Research Institute) submitted a report for a decommissioning plan to MOST (Ministry Of Science Technology) and started the decontamination and decommissioning project of the KRR-1&2(Korea Research Reactor1&2) research reactor in January of 1997[1]. The decommissioning activities of the radioisotope production facilities and lead hot cells and set to the decommissioning database project have been launched upon in August of 2001. Figure 1 is a view of the KRR-1&2. As a preliminary study to materialize the decommissioning database on the KRR-1&2, a study on a conceptual data modeling in relation to various kinds of data was carried out. A DBMS (Database Management System) is divided into Relational DBMS, Object-Oriented DBMS, and

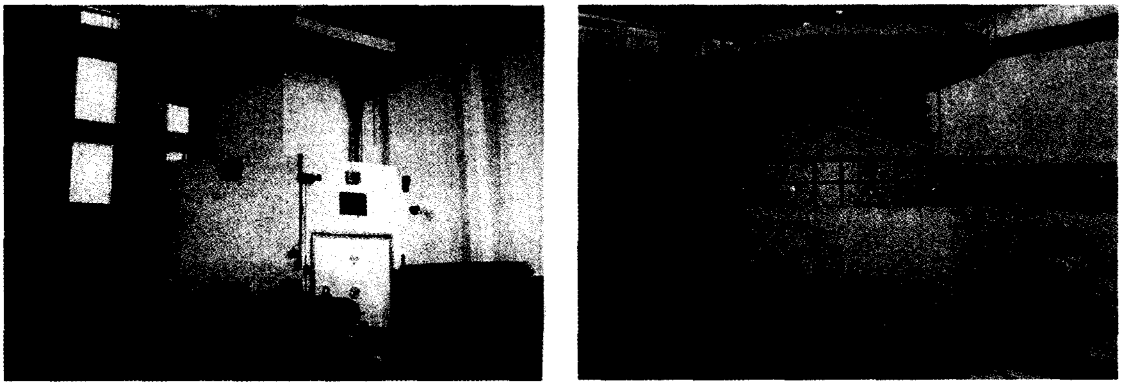

Fig. 1. A Panoramic Photograph of KRR-1 (left) and KRR-2 (right)

Object-Relational DBMS. Relational DBMS that has an outstanding capability to manage a structured huge data was selected. For hardware, a Client/Server system that has excellent security in the system, network, and data was chosen. The code of the decommissioning activities that has information such as work, facilities, radioactive waste, and radiology was designed. In order to maintain data integrity and data redundancy, ERD (Entity Relationship Diagram) was created. The current state of the decommissioning database in EU (European Union) and Japan has been investigated. The important factor to be considered first of all during decommissioning activities is to minimize a worker dose, amount of radioactive waste, and decommissioning cost.

2. Current Status of the Abroad Decommissioning Database

2.1. Japan

The Japan Power Demonstration Reactor (JPDR) dismantling demonstration project, which was successfully completed by the year 1996, was the first decommissioning experience of a nuclear power plant in Japan. The project intended to demonstrate the dismantling technologies and work activities and to acquire experience and data on the technologies applied and implemented in ensuring that the dismantling project was safely completed. The lessons learned from the project indicated that information integration is one of the key factors in the conduct of the dismantling project in an efficient way. After completing the dismantling project, efforts have been made to analyse the data for prescribing dismantling activities, together with the development of computer tools for the planning and management of the dismantling project. By analysing the JPDR dismantling project data, dismantling activities were characterized to derive unit productivity factors, work difficulty factors and typical work structures, which will be applied to the estimation of manpower needs and worker doses in other decommissioning projects. The computer tools that developed in the research and development phase for the dismantling of JPDR were improved and systematized to effective estimation of radioactive inventory, resource and worker doses of radiation, etc. in planning a decommissioning project. COSMARD(Code System for Planning and Management of Reactor Decommissiming) consists of several computer programs and related databases to manage the information efficiently [2].

2.2. EU (European Union)

The European Union (European Community until 1993) has been conducting four successive R&D programs on the decommissioning of nuclear facilities performed under sharing contacts since 1979. The main objective of this program was, and is to establish a scientific technological basis for the safe, socially acceptable and economically affordable decommissioning of obsolete or redundant nuclear installations. The objective of ECDB NET, which is one of the scientific technologies, is to set up a database network to provide easy access to the databases EC DB Tool and EC DB Cost within the European Union. The work, which started in 1997 and is due for completion at the end of 1998, is funded by the European Commission in the project "Set-up of an infrastructure for a EU-wide use of the databases EC DB COST and EC DB TOOL. In addition, information pages for the World-Wide-Web are being developed to give the public an understanding of the subject "Decommissioning of Nuclear Installations". At the heart of the concept chosen is a 'client-server' configuration. It consists of two or more computers working together. One computer, the server, performs the job of coordinating access to the ORACLE database, the other computer, the client, servers the application user. The communication between the server and the client computer(s) is established via the Internet. In the ORACLE-software solution chosen the user interface does no longer run as a form module binary on the client that has to be generated for each operating system, instead, the user interface will be send a database logon to the client as a programmed (Java Applet) which is executed by a Web Browser. Thus, the only software that has to be installed and started on the client is a Java capable Web Browser [3].

3. Scope of Decommissioning DB

3.1. User Requirements

Plenty of methods are discussed in the literature regarding requirement determination [4]. Most of them are business oriented and related to overall of process in the organization starting from the mission and ending with the final outcome. The general method is based on 1) reviews of current reports, 2) conducting of research what is already done, 3) visiting similar system installations. Foundations of the individual method are interviews, observations, questionnaires, and prototype system.

3.2. Design of the Decommissioning DB System

3.2.1. Hardware Configuration

The development of the computer hardware is going on from the environment of the central computing at mainframe to the distributed the client/server system and at the present circumstances is connected to the environment of Web [5].

3.2.1.1. Client/Server System

Client/server computing is the logical extension of modular programming. Modular programming has its fundamental assumption that the separation of a large piece of software into its constituent parts ("modules") creates the possibility for easier development and better maintainability. Client/server computing takes this a step further by recognizing that those modules need not all be executed within the same memory space. With this architecture, the calling module becomes the "client" (that which requests a service), and the called module becomes the "server" (that which provides the service). The logical extension of this is to have clients and servers running on the appropriate hardware and software platforms for their functions. The client is a process (program) that sends a message to a server process (program), requesting that the server perform a task (service). The client-based process is the front-end of the application that the user sees and interacts with. The client process contains solution-specific logic and provides the interface between the user and the rest of the application system. The client process also manages the local resources that the user interacts with, such as the monitor, keyboard, workstation CPU and peripherals. One of the key elements of a client workstation is the Graphical User Interface (GUI). A server process (program) fulfills the client request by performing the task requested. Server programs generally receive requests from client programs, execute database retrieval and updates and manage data integrity and dispatch responses to client requests.

3.2.1.2. Web (Intranet) System

A Web server is a program that, using the client/server model and the World Wide Web' s Hypertext Transfer Protocol (HTTP), serves the files that form Web pages to Web users (whose computers contain HTTP clients that forward their requests). Every computer on the Internet that contains a Web site must have a Web server program. Two leading Web servers are Apache, the most widely installed Web server, and Microsoft' s Internet Information Server (IIS). Web servers often come as part of a larger package of Internet- and intranet-related programs for serving e-mail, downloading requests for File Transfer Protocol (FTP) files, and building and publishing Web pages.

3.2.2. Software Configuration

3.2.2.1. Relational DBMS

A relational database management (RDBMS) is built on the three-schema structure, such as external, conceptual, and internal. The external schema defines how users access and view the output from the RDBMS, independent of how the data is physically stored or accessed. Such access and manipulation is performed by users who employ procedural languages, such as Structured Query Language (SQL). In some instances, a query language may be embedded in a procedural language to facilitate Online Transaction Processing (OLTP) and similar applications. The conceptual schema defines the relational database model. It consists of a set of normalized tables. The internal schema consists of the physical organization of the data (e.g., sequential, indexed sequential, and direct) in terms of physical data structures and access methods of the computer' s operating system. The internal schema is usually covered in computer hardware courses. The relational model is based on the set of theory of mathematics [6]. Tables define the structure. Tables are called relations in mathematical terms. Systems professionals often use terms "table" and "relations" interchangeably [7].

3.2.2.2. Object-Oriented DBMS

Object-oriented database management (OODBMS) concepts have evolved in three different disciplines; firstly in programming languages, then in artificial intelligence, and then in databases. An object-oriented approach to programming is based on the concepts of encapsulation and extensibility. Object-oriented programming encapsulates in an object some data and, programs to operate on the data; the data is the state of the object, and the code is the behavior of the object. Extensibility refers to the ability to extend an existing system without introducing any changes to it. Extensibility is an especially powerful concept for building and evolving very large and complex software system. An object-oriented approach to programming offers extensibility in two ways: behavioral extension and inheritance [8].

3.2.2.3. Object-Relational DBMS

Central to object-oriented analysis and design is the idea that an object' s interface (the means by which it is handled) should be separate from any details on its implementation (the data structures and logic within it). The role object-oriented analysis plays in developing ORDBMS databases is as a conceptual framework for working with the extensible type system. What distinguishes an object-relational DBMS from more conventional software frameworks-such as pure object-oriented DBMS, application-servers, or TP-monitor middleware-is that the embedded object classes are deployed within an abstracted or logical data model. Some important principles of the ORDBMS data model are those which types do not have to be used to define a table' s structure or a column. The behaviour of this object can be used to implement complex query operations over otherwise conventional data. Also, most object-oriented methodologies acknowledge that sometimes what you want is not really an object, but simply a function or procedure [9].

3.2.3. Classification Scheme on Decommissioning Data

Codes, also called account numbers, identifiers, and keys, usually represent an item that is entered in a form to identify a particular item or to signify certain kinds of processing. Codes represent a very important element used to classify and identify people, resources, documents, forms, events, and transactions into specific group and for distinguishing these items within groups. The following guidelines should be kept in mind when designing a code.

1. The code design must be flexible to accommodate changing requirements. It is costly and confusing to change the coding structure every few months. The coding structure should not be so extensive, however, that section of it will not be used for a number of years.
2. Standardization procedures should be established to reduce confusion and misinterpretation for persons working with the code. Some of the procedures that can be easily standardized in most systems.
3. The layout of the code itself should have sections, which are of equal length. For example, a chart of accounts code should read 001-199, not 1-199
4. Codes longer than four alphabetic or five numeric characters should be divided into smaller segments for the purpose of reliable human recording and recall.

Codes are normally classified by the symbol arrangement used to identify the groups to which coded items belong. These structures are sequence, block, and group [7].

4. Results and Discussion

4.1. User Requirements

In order to perform the analysis of the user requirement for the establishment of the scope of decommissioning database, Firstly, after collecting

20 materials such as a decommissioning project report, a decommissioning design specification, facility status and radiation & activity survey report [10] etc., we had analyze the above materials. Secondly, people who are involved in the development of decommissioning database visit the KRR-1&2 site and make a close inspection of the facilities status and the process of data production. Thirdly, a 4 major information such as facility information that includes the reactor halls, auxiliary facilities, and yard facilities and work information in relation to worker and radioactive waste information and radiological information had extracted. Table 1 explains major information that has to be extracted. There are more than twenty modeling tools of the trade exist such as Data Flow Diagram (DFD), Data Dictionary, Entity Relationship Diagram (ERD), State Transition Diagram (STD), Structured Program Flowchart, Warnier-Orr Diagram (WOD), and Jackson Diagram. Most of these modeling tools are based on the structured approach. In this research, the 4 major information using ERD tool had designated. Figure 2 illustrates an ERD on the decommissioning information.

4.2. Design of the Decommissioning DB System

4.2.1. Hardware Configuration

The Client/Server system that has more an advantage with a standard technology and safety and response time than Web (Intranet) that has outstanding features in an area of economical efficiency and support to the user was chosen. Figure 3 illustrates the configuration of decommissioning database system that is based on a Client/Server system.

Table 1. Major Information Including User Requirements

Major Information	User Requirements
1. Facility Information	- Radiological information - Nuclide information - Work information - Radioactive Waste information
2. Work Information	- Manpower information - Organization information - Technical level - Worker information
3. Radioactive Waste Information	- Facility information - Waste-Radiological information - Container information
4. Radiological Information	- Radiation dose rate - Alpha removable contamination - Beta removable contamination - Fixed contamination - Surface dose rate - Air, Water contamination

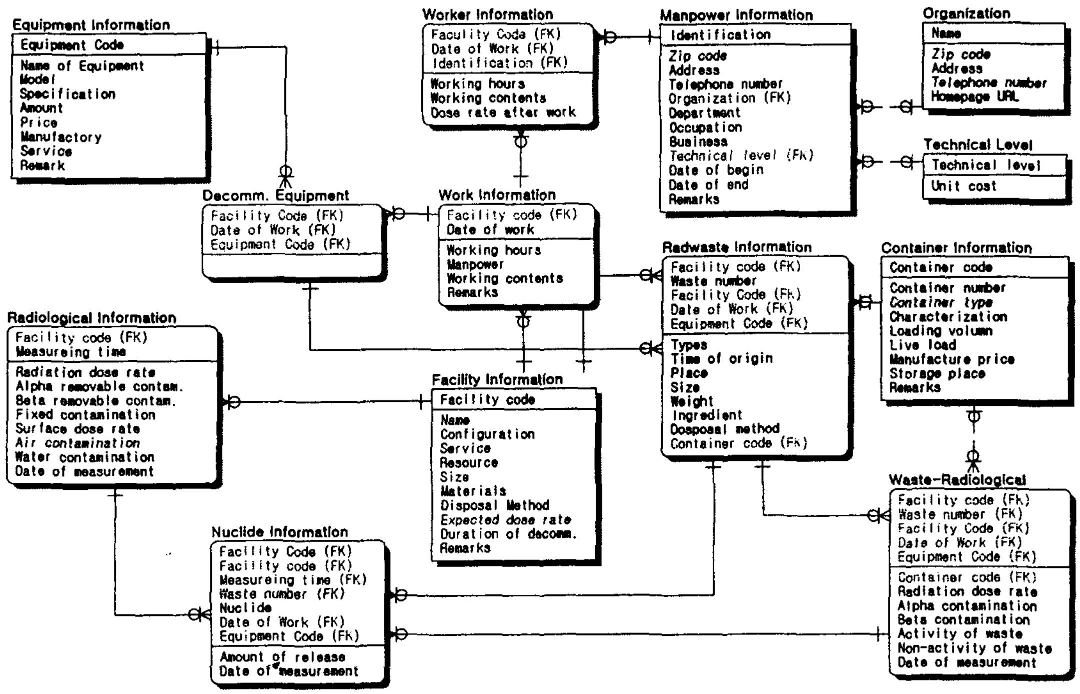

Fig. 2. ERD (Entity Relationship Diagram) on the Decommissioning Information

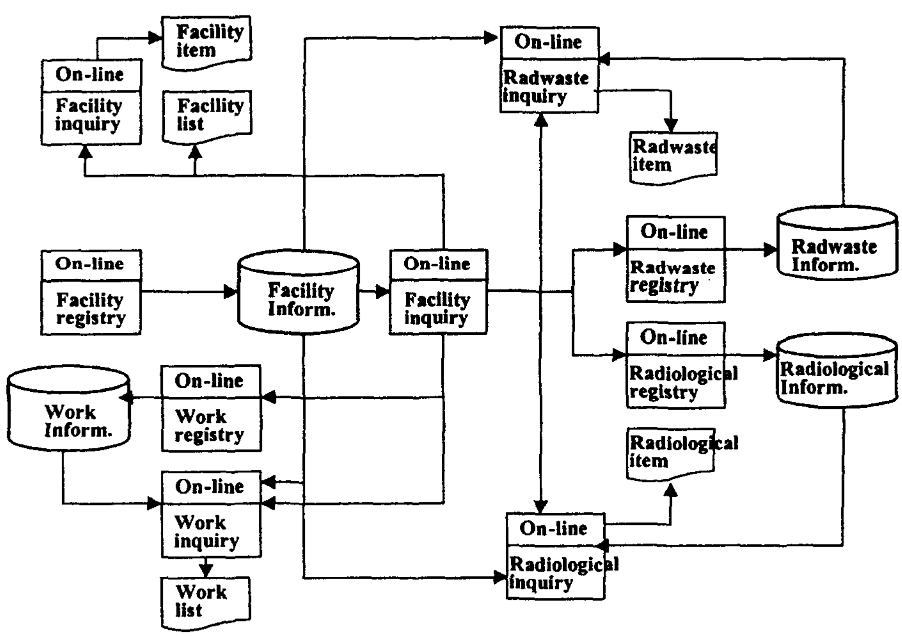

Fig. 3. System Flow Diagram of Decommissioning Database

4.2.2. Software Configuration

There are 3 different products such as RDBMS, OODBMS, and ORDBMS. In the decommissioning database on the KRR-1&2 case, RDBMS that has a great ability to manage a huge data and a structured programming was chosen.

4.2.3. Classification Scheme on Decommissioning Data

The codes that have designed in relation to decommissioning activities were classified 7 items. These structures are:

①XXX-②XXX-③XXX-④XX-⑤XX-⑥X-⑦XXX

① Facility group
- K2A: KRR2 Auxiliary facilities
- K2R: KRR2 Reactor Hall
- K1A: KRR1 Auxiliary facilities
- K1R: KRR1 Reactor Hall
- YFS: Yard facilities

② Main object in facility (for example, KRR2 Auxiliary facilities)
- ILE: Isotope experimental laboratory
- IPR: Isotope production room
- CHC: Concrete hot cell

③ Sub object in main facility (for example, Isotope experimental laboratory)
- L32: Experimental room 132
- C01: Concrete hot cell 1

④ Sub-sub object in main facility (for example, Fume Hood3)
- CC: Concrete hot cell crane
- F1: Fume hood 1
- M1: Manipulator 1

⑤ Sequence number

⑥ Facility and Waste (original facility or waste type)

⑦ Number of waste (sequence number)

Here, Table 2 is an example for the code of the KRR2.

Table 2. Sample Code on the Auxiliary Facilities at KRR2

Code	Facility Group	Main object	Sub Object	Sub-Sub Object
K2AC0R00000000I000	Auxiliary Facilities at KRR2			
K2AIELL32T100I000		Radionuclide Laboratory		
K2AIELL34T100I000			Experiment 134	
K2AIELL34F100I000				Fume Hood 1
K2ACHC0000000I000		Concrete Hot Cell		
K2ACHCC010000I000			Concrete Hot Cell 1	
K2ACHCC01CF00I000				Concrete Hot Cell floor
K2ACHCC01CC00I000				Concrete Hot Cell crane
K2ACHCC01M100I000				Manipulator1

5. Conclusions

Conceptual data modeling of the whole decommissioning activities at the beginning of the establishment of a decommissioning database was carried out. The major information related to decommissioning activities such as a facility, a work, a radioactive waste, and a radiological from the user requirements was extracted. The hardware system for the decommissioning database was selected a client/server system that has outstanding securities features. The software elements will be used RDBMS that has a widely used and an ability to manage a huge data. The ERD will be based a design of logical data modeling and a physical one and will contribute to the design of the prototype and user interface of the decommissioning database. The decommissioning DBMS will be useful tool to analyze radiation safety management, radioactive waste management, and worker' s dose. Also, This system is planning to share the information with an associated company. Moreover, it can be expected to support the technology on the research and development for the decommissioning of the nuclear facilities abroad.

Acknowledgment

This work has been carried out under the nuclear research and development program of Korea Ministry of Science and Technology.

Reference

1. K.J Jung et al., "Dismantle Plan on Research Reactor 1&2", KAERI/TR-1654/(2000).
2. S. Yanagihara, " Development of Computer Systems for Planning and management of Reactor Decommissioning". J. Nucl. Sci. Technol., vol.38, No.3, pp.193~202, March (2001).
3. ECDB NET, "European Commission Database Network". http://www.ec-decom.be
4. Norman, R.J., "Object-oriented Systems Analysis and Design" , Prentice Hall International, Inc., New York, pp 431, 1996
5. J.S. Sunwu et al., "The Implementation and Security Guideline to the Intranet". NCA IV-RIR-98054/(1998).
6. Cecelia Bellomo, "Relational Database Systems: Data Integrity", Interact, pp.107, October (1990).
7. J.G. Burch, "Systems Analysis, Design, and Implementation", Published by Boyd & Fraser publishing company, (1992).
8. W. Kim, "Introduction to Object-Oriented databases", Published by The MIT Press, (1990).
9. Paul Brown, "Developing Object-Relational Database applications", Published by Informix Press, June (2002).
10. S.K. Park et al., "Facility status and radiation & activity survey report for decommissioning of TRIGA Research Reactors", KAERI/TR-1153/(1998).