How Valuable is Planned Data Redundancy in Maintaining the Integrity of an Information System through its Database

Eghosa Ugboma
eugboma@fmuniv.edu/eugboma@aol.com
Department of Computer Science and Mathematics
Florida Memorial University
Miami Gardens, Florida 33054 USA

Abstract

Although planned (controlled) data redundancy increases the distribution of redundant data to a very meager degree, this type of redundancy most often involves few columns of database data files that supports an information system and helps to enforce the integrity of the information system. The effectiveness of an information system depends largely on the database that supports the information system. This paper discusses the importance of planned data redundancy and how it is implemented in assisting an information system to be deemed effective in generating the required data and information to its intended users. Due to their unique functions and for easy understanding, this paper uses the primary keys and foreign keys columns to demonstrate how planned data redundancy is implemented to help maintain the validity of the data pool that an information system needs to produce the desired information. First normal form (1NF), third normal form (3NF), fourth normal form (4NF), and the Redundant Data technique of denormalization process are the yardsticks used to illustrate how controlled data redundancy is generated, in which keys' columns it occurs, and its benefits to information systems. Planned data redundancy is classified into two groups, namely, internal controlled data redundancy and external controlled data redundancy. To understand their formation and how they are used to maintain the integrity of a database and uphold the reliability of the information system that uses the database, normalization and normal forms as well as denormalization are discussed at the elementary level.

Keywords: Planned Data Redundancy, Normalization, Normal Forms, Denormalization, Database, Data Pool, Data Files

1. INTRODUCTION

The intent of this paper is not to discuss normalization and denormalization processes, but to demonstrate how an information system can be productive by maintaining the integrity of the database that supports it through planned data redundancy approach. The paper's focus is on the planned replication of data of the primary keys and/or the foreign keys columns of data files that supply an information system with the needed data.

Planned data redundancy is mandatory for an information system's data pool to be efficiently restructured if the tasks performed by the information system are to be successfully accomplished. The effectiveness of an information system depends largely on the database that supports the information system. To help understand the formation of planned data redundancy, this article discusses normalization and denormalization at their elementary levels.

Keys columns (primary and foreign) are used to explain this type of needed data redundancy because of their unique functions and the stability of the data they store. The primary keys and the foreign keys values of data files rarely change or are altered very infrequently and as a result, they are used to form the pillars for establishing database integrity and making the information system the database supports worthwhile.

2. PLANNED DATA REDUNDANCY

Planned (controlled) data redundancy is the required replication of data to achieve a positive end to the successful implementation of databases that can adequately support any data- or information-generating system. This type of redundancy does not usually require the replication of rows (records) and it is needed when reorganizing data files that any system needing data depends on.

This sort of redundancy typically enhances database performance, and as such, it also enhances the information system that depends on the database in terms of generating reports using data from more than one data pool. Planned data redundancy assists an information system to ensure data dependency. Data dependency, in this presentation, is a database aspect where related data are stored in the same data pool. With planned redundancy, a database that supports an information system is easily maintained, and in turn, the information is also easily maintained.

The controlled redundancy plays beneficiary roles in the integrity of data in databases. Although the redundancy increases the number of times the same data appear in data pools, the data are planned and needed in order to maintain the validity of data toward a positive end. Controlled data redundancy does not introduce the unnecessary duplication of data rather it creates the required data to solve certain database problems that puts the database in a consistent state. As a result, the information system that depends on the database for its data resources is also in an error-free state. One benefit of the planned redundancy is the speed with which data files are queried.

3. NORMALIZATION PROCESS

Normalization is the course of action used by database developers to identify and remove problems from databases in order for data in the data files to have integrity. Normalization uses yardsticks called normal forms (NFs) to construct new data files that are free of anomalies and that contain the same data as the old problem-stricken files.

Normal forms (NFs) are the standard measurement tools used for reorganizing the structure of databases. In addition to reducing unnecessary duplication of data, normalization assists in improving an entire database structure.

4. REVIEW OF LITERATURE

Keller (2002) defines planned data redundancy as “a technique to use redundant fields in a physical database in order to speed up reading database access”. He further points out that such redundancy should be reserved and used for data that are usually stable in a database. In other words, Keller is saying that the planned data redundancy approach should involve data that are rarely altered such as the values contained in the primary keys and foreign keys columns of data files. The values of these keys’ columns are vital to the successful processing of databases and the information systems the databases support. Frequently altering or changing the data stored in the keys columns might result in situations where the databases become unreliable and thus making the information systems that depend on them non dependable.

Plew and Stephens (2003) refer to controlled data redundancy as a requirement for increasing a database performance through denormalization process. They stress that although the planned replication increases data in denormalizing a database, the aim of such a redundancy is to improve performance. For the purpose of this paper, controlled data redundancy is defined as the necessary duplication of data to achieve certain acceptable levels of organization and performance for a database and the system that depends on the database.

5. OPERATION METHODS

This paper reveals the valuable role of planned data redundancy and demonstrates how such type of redundancy is implemented in assisting to maintain an information system’s effectiveness. For easy comprehension, this article uses the first normal form (1NF), third normal form (3NF), and fourth normal form (4NF) of normalization process as well as the Redundant Data method of denormalization process to demonstrate the implementation.

The most important step in understanding planned data redundancy in a database is to understand the formation of the different levels of normalization's normal forms. The primary keys and the foreign keys of the data files used as examples in this article are employed to explain the formation of the controlled redundancy.

In this paper, planned data redundancy is categorized into two classes namely, internal controlled data redundancy and external controlled data redundancy. First normal form (1NF) and fourth normal form (4NF) are used to illustrate internal controlled data redundancy, while the third normal form (3NF) and the Redundant Data technique of denormalization process are used to demonstrate external controlled data redundancy. The explanations are straightforward as the sample data files used in the various illustrations in this paper show how planned and needed redundant data are achieved, and where the redundant data can occur in the keys columns of data files. Database data files where the keys (primary, foreign) columns accept the replication of data to maintain the needed integrity are indications of the existence of planned data redundancy.

Naming Conventions

To differentiate between file names and column names, the article employs the usage of uppercase letters to assign names to data files and mixed case letters to assign names to columns. In addition, both files' and columns' names are boldfaced. The abbreviations "PK" and "FK" are used to denote files' primary keys and foreign keys columns respectively. The primary keys of files are also underlined.

Internal Controlled Data Redundancy

Internal controlled data redundancy assists in establishing data integrity in database data files with composite primary keys. To illustrate this category of planned data redundancy, this section uses first normal form (1NF) and fourth normal form (4NF) as examples to show how the redundancy is implemented.

First Normal Form: The first objective of first normal form in normalization process is to eliminate repeating groups that exist within the rows (records) of data files and to make each row in the files unique. The second objective is to identify the primary keys columns for the data files. Typically, the primary keys of data files in first normal form are made up of more than one column, thus making the keys composite primary keys. For each row in the data files with composite primary keys to be unique, the columns that made up the primary keys must allow the duplication of data. However, this type of data replication is planned or controlled to achieve the normal form and make the data contained in the database valuable.

Although data files with composite primary keys might still contain database anomalies, the composite keys are, usually, required to convert files to first normal form. The type of controlled data redundancy that exists in first normal form is referred to as internal because the needed duplication occurs within the data files.

For example, consider the un-normalized data file shown below that contains data about faculty members and the courses they are assigned to teach. The structure contains repeating groups that are made up of **Section**, **FacultyNumber**, and **FacultyName** columns and uses the **CourseCode** column as its primary key. The structure is used as the starting base for demonstrating internal controlled data redundancy.

File Name: FACULTY_CLASS
PK

Course Code	Course Name	Section	Faculty Number	Faculty Name
CSC 101	Intro. To Computers	A	FAC003	Paul Mathew
		B	FAC011	John Philip
CSC 232	Principles of Programming	A	FAC003	Paul Mathew
		B	FAC006	Mary Flyes
		C	FAC011	John Philip
CSC 483	Database Concepts	A	FAC001	Angela Stands

Due to the progressive nature of normal forms, the above un-normalized structure is converted into a file in first normal form. The resulting data file (the new 1NF data file) with its new composite primary key and the needed redundant data is displayed below.

File Name: FACULTY_CLASS

PK		PK	PK	
Course Code	**Course Name**	**Section**	**Faculty Number**	**Faculty Name**
CSC 101	Intro. To Computers	A	FAC003	Paul Mathew
CSC 101	Intro. To Computers	B	FAC011	John Philip
CSC 232	Principles of Programming	A	FAC003	Paul Mathew
CSC 232	Principles of Programming	B	FAC006	Mary Flyes
CSC 232	Principles of Programming	C	FAC011	John Philip
CSC 483	Database Concepts	A	FAC001	Angela Stands

An example of internal planned data redundancy in a key column

The above data file has been converted to first normal form. It no longer has repeating groups and the composite primary key consists of the **CourseCode**, **Section**, and **FacultyNumber** columns. In this case, controlled data redundancy is implemented by allowing each of the three columns that constitute the composite primary key to accept and store the duplication of its data. As a result of the duplication, each of the three columns fails to qualify individually as a candidate key and they must, therefore, work as a unit to distinguish one row from another. This type of column combination helps make a database reliable. The **Section** column of the file is chosen as part of the composite primary key for the fact that a faculty member can be assigned to teach several sections of the same course.

In the resulting file above, the arrows point to an example of the planned replication of data in a key column that is needed to (a) convert the file to first normal form, and (b) uniquely identify each row. The duplication, in this case, is necessary to arrive at the normal form and, as such, the replication is referred to as controlled data redundancy. Using the data file above, controlled data redundancy occurs in the **CourseCode**, **Section**, and **FacultyNumber** columns and because the duplication exists within the file, this type of planned redundancy is specifically known as internal controlled data redundancy. This type of redundancy helps make the information generated by an information system dependable.

Fourth Normal Form: The objective of fourth normal form is to remove multi-valued dependencies from databases. In other words, fourth normal form does not allow two or more independent multi-valued data with no direct association between them to exist in data files.

For example, the data file below contains data about faculty members, the courses they teach, and the students they advise. The file is used as the base for illustrating internal controlled data redundancy. It is assumed that the file is in third normal form with a composite primary key that comprises the file's three columns.

File Name: FACULTY_COURSE_ADVISEE

PK	PK	PK
Faculty Number	**Course Code**	**Advisee Number**
FAC003	CSC 232	STU090912
FAC003	CSC 232	STU107823
FAC003	CSC 483	STU090912
FAC003	CSC 483	STU107823
FAC011	CSC 101	STU118952
FAC011	CSC 101	STU123456
FAC011	CSC 473	STU118952
FAC011	CSC 473	STU123456

To remove the multi-valued data and to demonstrate how planned data redundancy is implemented, the above data file is converted into two files of fourth normal form. The resulting files (the new 4NF files) with their new composite primary keys and the needed redundant data are shown below.

File Name: FACULTY_COURSE

PK	PK
FacultyNumber	**CourseCode**
FAC003	CSC 232
FAC003	CSC 483
FAC011	CSC 101
FAC011	CSC 473

An example of internal planned data redundancy in a key column

File Name: FACULTY_ADVISEE

PK PK

FacultyNumber	**AdviseeNumber**
FAC003	STU090912
FAC003	STU107823
FAC011	STU118952
FAC011	STU123456

An example of internal planned data redundancy in a key column

Typically, all columns of data files in fourth normal form are used to establish the files' primary keys. Because all columns are involved in the formation of the primary keys of files in fourth normal form, the keys are composite primary keys. Again, columns of composite primary keys allow the necessary duplication of data to achieve the desired normal form.

In the above example, controlled data redundancy is implemented by allowing the needed duplication of data in the **FacultyNumber** columns of the two resulting files. These accepted redundant data are necessary to achieve fourth normal form. The arrows point to the needed duplication. This type of redundancy is specifically referred to as internal controlled data redundancy for the fact that the duplication of data occurs in the **FacultyNumber** column of each resulting data file.

Note: Fourth normal form (4NF) can also be used to illustrate the implementation of external controlled data redundancy. By reviewing the content of the two resulting data files above, you will notice that both files contain identical data values in their **FacultyNumber** columns. That is an example of external planned data redundancy.

External Controlled Data Redundancy

Third normal form (3NF) and the Redundant Data technique of denormalization process are used to illustrate external planned data redundancy. This section discusses the implementation of this category of the planned redundancy.

Third Normal Form: The objective of third normal form is to remove transitive dependencies from databases. A transitive dependency exists in a database data file when a non-key column that depends on a data file's primary key depends also on another non-key column in the same file.

In third normal form, external controlled data redundancy helps to define and enforce data integrity. This type of planned redundancy is implemented by establishing associations between files with primary keys (parent files) and files with foreign keys (child files) where the child files' foreign keys reference the primary keys of the parent files.

In this normal form, controlled data redundancy exists when the primary keys values of the parent files are replicated as the foreign keys values of the child files. The foreign keys of the child files serve as lookup functions (features) that are used to relate to the records in the parent files.

To demonstrate how planned data redundancy is achieved in third normal form, consider the data file below that contains data about students, their major disciplines, and their academic departments. It is assumed that the file is in second normal form with the **StudentNumber** column as the file's primary key.

File Name: STUDENT_DEPARTMENT

PK

Student Number	**Student Name**	**Major**	**Department Number**	**Department Name**
STU090910	John Stevens	Airways Management	AV19	Aviation
STU107823	Florin Kelly	Computer Science	MC24	Mathematics and Computer Science
STU118955	Angelica Lite	Elementary Education	ED05	Education
STU123461	Johnson Slate	Economics	BA03	Business Administration

Although the **StudentNumber** column is the file's primary key, but knowing the value of the **DepartmentNumber** column also determines a value in the **DepartmentName** column. This is a problem in database which can also affect the performance of the information system the database supports. This is due to the fact that the **DepartmentNumber** column is not part of the file's primary key.

The conversion of the data file to third normal form demonstrates how controlled data redundancy is implemented. The following two resulting files establish planned data redundancy after the conversion.

File Name: STUDENT

PK **FK**

Student Number	**Student Name**	**Major**	**Department Number**
STU090910	John Stevens	Airways Management	AV19
STU107823	Florin Kelly	Computer Science	MC24
STU118955	Angelica Lite	Elementary Education	ED05
STU123461	Johnson Slate	Economics	BA03

An example of external planned data redundancy in key columns

File Name: DEPARTMENT

PK

Department Number	**Department Name**
AV19	Aviation
MC24	Mathematics and Computer Science
ED05	Education
BA03	Business Administration

The resulting two data files above are now in third normal form. Controlled data redundancy is implemented by using the common column of both files. In the resulting files, the common column is the **DepartmentNumber** column. The column is contained in each file. In the data file named **DEPARTMENT**, the column is the primary key while in the file named **STUDENT** the column is the foreign key.

Here, planned data redundancy is illustrated by using both the primary key and the foreign key. The foreign key column of the child file (**STUDENT**) can contain the necessary duplication of data that are contained in the primary key column of the parent file (**DEPARTMENT**). In other words, values occurring in the foreign key column of the **STUDENT** file must be the replication of the data that exist in the primary key column of the **DEPARTMENT** file. The foreign key of the child file (**STUDENT**) serves as a lookup function and it is used to directly connect the rows in the child file to their corresponding rows in the parent file (**DEPARTMENT**).

In the above data files, the arrows point to an example of planned data redundancy and where it occurs in third normal form. In this situation, without defining a common column that allows the necessary duplication of data between the two files, the third normal form will not be realized. Due to the fact that the planned data duplication occurs in separate database data files, this type of redundancy is specifically referred to as external controlled data redundancy. This type of redundancy helps to keep the information system that the database supports in a error-free condition and thus making the information system dependable.

Denormalization Process: The objective of denormalization process is to move down the normal forms' ladder one or two steps to increase database access performance. Denormalization is a process that violates normalization and its main job is to reduce or remove the links (joins) among database data files. Denormalization can be considered as normalization process in reverse (downward) to a limited degree.

Denormalization process uses several techniques to speed up database access, but for this paper, the method referred to as "Redundant Data" is used to demonstrate the implementation of planned data redundancy. The Redundant Data technique of denormalization process involves the replication of data from few columns of data files to other files where the replicated (few) columns are accessed frequently by the other files within the database.

The following three normalized data files (**COURSE**, **FACULTY**, **STUDENT**) in third normal form are used to illustrate controlled data redundancy using the Redundant Data method. For demonstration purposes, it is assumed that users of the **FACULTY** file and the **STUDENT** file very often access the **COURSE** file to find out the courses that are being offered.

Again, for the purpose of this article, it is assumed that the data contained in the **COURSE** file rarely change. In this circumstance, denormalizing both the **FACULTY** and **STUDENT** files to increase access speed is necessary.

File Name: COURSE

PK

CourseCode	**CourseName**
CSC 101	Intro. To Computers
CSC 232	Principles of Programming
CSC 483	Database Concepts

File Name: FACULTY

PK **FK**

<u>**Faculty Num**</u>	**Faculty Name**	**Course Code**
FAC001	Angela Stands	CSC 101
FAC003	Paul Mathew	CSC 232
FAC011	Mary Flyes	CSC 483

File Name: STUDENT

PK **FK**

<u>**Student Num**</u>	**Student Name**	**Course Code**
STU090910	John Stevens	CSC 101
STU107823	Florin Kelly	CSC 232
STU118955	Angelica Lite	CSC 483

Using the Redundant Data technique to denormalize both the **FACULTY** and **STUDENT** files in the above example removes the joins, put the files in second normal form, and enhances data retrieval performance of the database. Although this method creates database's partial dependency problem, the intended outcome is achieved – improvement of data retrieval performance through the needed redundant data. Controlled data redundancy is implemented by duplicating the necessary columns of the **COURSE** file in both the **FACULTY** and **STUDENT** files.

The new **FACULTY** and **STUDENT** files after denormalization process are shown below.

File Name: FACULTY

PK

<u>**Faculty Num**</u>	**Faculty Name**	**Course Code**	**Course Name**
FAC001	Angela Stands	CSC 101	Intro. To Computers
FAC003	Paul Mathew	CSC 232	Principles of Programming
FAC011	Mary Flyes	CSC 483	Database Concepts

An example of external planned data redundancy in former foreign key columns

File Name: STUDENT

PK

<u>**Student Num**</u>	**Student Name**	**Course Code**	**Course Name**
STU090910	John Stevens	CSC 101	Intro. To Computers
STU107823	Florin Kelly	CSC 232	Principles of Programming
STU118955	Angelica Lite	CSC 483	Database Concepts

In the two resulting data files above, the arrows point to an example of planned data redundancy and where it occurs using the Redundant Data technique of denormalization process. This type of planned redundancy is specifically called external controlled data redundancy due to the fact that the duplication of the needed columns occurs in separate database data files (**FACULTY** and **STUDENT**).

6. CONCLUSIONS

The intent of this paper is to illustrate the valuable role of planned (controlled) data redundancy and how and where it is implemented in a database to assist in maintaining the integrity of an information system. The paper's outcomes comply with that intent. Implementing planned data redundancy is not an option but a mandatory approach when it comes to making data contained in a database reliable or making the information system that uses the database dependable. Normal forms such as first normal form and denormalizing process using particular techniques such as the Redundant Data method are the yardsticks used in this article to demonstrate the implementation and benefits of planned data redundancy in terms of producing dependable information.

Again, it should be noted that the aim of the paper is to demonstrate the good points of controlled data redundancy in an information system's arena. These good points are manifested in the sample tables used as examples in different sections of the paper to illustrate the planned redundancy.

Although, controlled data redundancy increases the replication of data, the duplication is necessary and assists in establishing and securing efficient ways of reorganizing database structures and maintaining the integrity of the information system the database supports. On that note, it can be said that this type of redundancy helps to (a) maintain the consistency of data contained in an information system's database and (b) make a database data valid.

Finally, it is hoped that this paper adds to the understanding of how controlled data redundancy is realized, where it occurs, and its benefits to database and information systems environments.

7. REFERENCES

Center for Technology in Government – University at Albany/SUNY (1997) A

Practical Guide to State-Local Information systems. http://www.ctg.albany.edu/resources/pdfrpwp/iis1.pdf.

Connolly, M. Thomas and Carolyn E. Begg (2002) Database Systems: A Practical Approach to Design, Implementation, and Management (3rd ed.). Addison-Wesley, New York.

Elmasri, Ramez and Shamkant B. Navathe (2002) Fundamental of Database Systems (3rd ed.). Addison-Wesley, New York.

Frick, R. David and Co (2004) Data Integrity. http://www.frick-cpa.com/ss7/Theory_DataIntegrity.asp.

Keller, W. (2002) Pattern: Controlled Redundancy.http://www.objectarchitects.de/ObjectArchitects/orpatterns/Performance/ControlledRedundancy

Kroenke, M. David (2003) Database Concepts (2nd ed.). Prentice Hall, New Jersey.

Modell, Martin (2002) The Various Types of Information Systems Analysis Projects. http://www.dai-sho.com/pgsa2/pgsa02.html.

Oz, Effy (1998) Management Information systems. Course Technology, Cambridge.

Plew, Ronald and Ryan Stephens (2003) The Database Normalization Process. http://www.informit.com/articles/article.asp?p=30646&redir=1

Pratt J. Pratt and Joseph J. Adamski (2002) Concepts of Database Management (4th ed.). Course Technology, Cambridge.

Rob, Peter and Carlos Coronel (2002) Database Systems – Design, Implementation, and Management (4th ed.). Course Technology, Cambridge.

Shasha, Dennis and Philippe Bonnet (2003) Database Tuning: Principles, Experiments, and Troubleshooting Techniques. Morgan Kaufmann Publisher, San Francisco.

Stair, M. Ralph and George W. Reynolds (2001) Fundamentals of Information Systems. Course Technology, Cambridge.

Swanson, Marianne and Barbara Guttman (1996) Generally Accepted Principles and Practices for Securing Information Systems. http://csrc.nist.gov/publications/nistpubs/800-14/800-14.pdf.

Wiederhold, Gio (2004) From Data Engineering to Information Engineering. http://www-db.stanford.edu/pub/gio/1994/inf-eng-abstract.html.

Wu, Jonathan (2004) Ensuring Data Integrity through the Use of Prevent and Detect Controls, Part I. http://www.evaltech.com/wpapers/ensuringdataintgrity.htm.