Designing a Research Database Management System

Kathryn S. Dawson and Sidney H. Schnoll

INTRODUCTION

For a Perinatal-20 study, many data were to be collected over an extended period. A system for managing the data was needed so that the integrity of the data, and the quality of the research based on the data, could be ensured from the study's start. More than 60 types of data collection forms were designed. To monitor a subject's progress through the treatment program, some forms were collected at multiple time points. Data from more than 45,000 forms were entered throughout the course of this 5-year study.

Before developing the database, it was necessary to have in place the research protocol that formally described the testing instruments and data collection forms used. The time points at which the forms were to be collected also had to be well defined. The use of this information allowed the database system implementation to include methods of validating the data. Data validation was accomplished on several levels: The system checked for data entry errors on an item-by-item basis and also jointly guaranteed the validity of several items. To ensure that only authorized personnel had access to the data, security protections also were implemented.

The design of the database was based on a relational database structure that allowed data to be viewed logically in tables of information. In general, a table was designed to correspond to each form. Columns in the table represented data items on the form. Ensuring that appropriate forms shared common fields allowed the data from multiple forms to be appropriately merged later. One advantage of this type of database structure was that additions to the database could be made easily. This proved to be an important advantage; at several times throughout the course of this study, additional forms were added to the protocol.

The following discussion describes various guidelines that were considered in the design and implementation of this project's database management system. Although the implementation of any database management system depends in part on the computer software chosen, the overall design considerations are the same.

DATA ENTRY FORMS

Well-designed data entry forms are essential to collecting data in any large project. In most large projects, numerous staff members enter data. To ensure consistency in the data collection, a manual should be written that defines all data entry fields and should be read and used by all personnel who collect data. Otherwise, there is no assurance that consistent meanings will be given to the data.

In general, the flow of the research data can be described according to the diagram shown in figure 1. The data originating from the subject are recorded on the data entry form and then entered into the computerized database. A data entry form can range from a piece of paper to a computerized data entry screen. These forms can correspond to predesigned testing instruments, such as the Addiction Severity Index (McLellan et al. 1985), or to project-specific information, such as session attendance or urine toxicology results. Each data item to be completed on the form is a field. The form should be designed to ensure that the valid values for each field are well defined.

In general, forms fall into two categories. One-to-one forms are those that are collected only once for each study subject. In contrast, one-to-many forms can be collected at one or more time points.

In the illustrative one-to-one Demos form shown in figure 2, the valid entries for the race field are clearly specified. When possible, short character or numeric codes should be used to enter data into each field. Variable length fields always should be avoided because they are difficult to validate and analyze later. *DSM-IV* (American Psychiatric Association 1994) and *ICD-10* (World Health Organization 1992) diagnostic codes, rather than text fields, can be used to describe diagnoses and medical conditions. The data entry fields should be arranged in a manner that makes it easy for the person who enters the data. For example, all fields are aligned in the form shown in figure 2.

Each form should contain a single field that uniquely identifies each study subject. Although names, Social Security numbers, and other personal identifying information may be included on the paper data entry form, to guarantee patient confidentiality, the database should not include these identifiers. Ideally, a unique study-specific case number should be assigned to each new subject. The project director should maintain the list, matching the case number with names in a secure location separate from the database. For data entry purposes, the unique case number should be included on all forms.

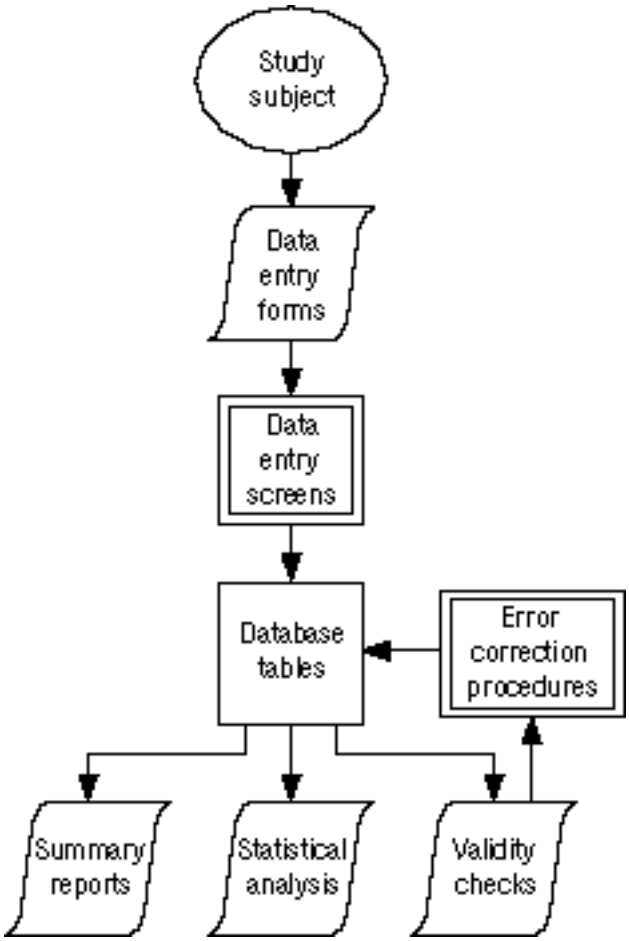

FIGURE 1. *Data flow from subject to database to application*

Figure 3 illustrates a one-to-many form called Beck (based on the Beck Depression Inventory) (Beck et al. 1988) that is collected at three time points. Within both one-to-one and one-to-many forms, a set of fields should be defined that uniquely identifies a single particular data entry form. This set of fields is used to retrieve the data for a single subject at a fixed time or to merge those data properly with data originating from other forms. The case number can serve as the unique identifier for one-to-one forms. For one-to-many forms, at least two fields must be used. For the one-to-many form in figure 3, the combination of case number and the date can be used. When more than one person collects the same data on a single subject, a tester identification code also may be needed.

Form

Intake Demographic Information
Collected at Admission

Subject Case Number: ___ ___ ___ ___ ___

Date of Admission (mm/dd/yy): ___/___/___

1. Date of Birth (mm/dd/yy): ___/___/___

2. Race: A. White B. Black ___
C. Hispanic D. Other

3. Gender: M F ___

4. 5-Digit ZIP Code of Residence: ___ ___ ___ ___ ___

Table Definition

Table Name: Demos

case	admit	dob	race	gender	ZIP

Table Documentation

Table Name: Demos

Field Description	Variable Name	Key Fields	Data Type	Validity	Null
Subject Case Number	case	key	character-5		N
Date of Admission	admit		date		N
Date of Birth	dob		date		N
Race	race		character-1	A,B,C,D	N
Gender	gender		character-1	M,F	N
5-Digit ZIP Code	ZIP		character-5		Y

FIGURE 2. *Form, table definition, and table documentation for a one-to-one form*

For one-to-many forms that are collected over time, a visit indicator, in addition to the date of the evaluation, may be helpful. For example, the Beck form in figure 3 is filled out during three specified assessments. Therefore, the visit field lists I, M2, and D as valid entries, which indicate specific data entry time points (Intake, Month 2, and Discharge) and may be more useful than the specific date on which the data are entered. For

Form

Monthly Beck Depression Inventory (BDI)*
Collected at Intake, Month 2, and Discharge

Subject Case Number: ___ ___ ___ ___ ___

Date of Evaluation (mm/dd/yy): ___/___/___

Visit (I, M2, D): ___

BDI Score (0-63): ___

Table Definition

Table Name: Beck

case	evaldate	visit	score

Table Documentation

Table Name: Beck

Field Description	Variable Name	Key Fields	Data Type	Validity	Null
Subject Case Number	case	key	character-5		N
Date of Evaluation	evaldate		date		N
Visit	visit	key	character-2	I, M2, D	N
BDI Score	score		integer	0-63	Y

FIGURE 3. *Form, table definition, and table documentation for a one-to-many form*

KEY: I=Intake; M2=Month 2; D=Discharge

*SOURCE: Beck et al. 1988

example, the forms corresponding to a given assessment could have varying dates and hence may be difficult to merge by assessment date. The use of a visit indicator also eliminates the necessity of determining the type of assessment, by comparing two date fields.

Null vs. Unknown

Answers may not be provided for all questions. In general, nonresponses can be categorized as either missing or unknown. A missing or null value is an indicator of a nonresponse, which is distinguished from an

"unknown" response where the subject has responded to the question. Careful review of each question will determine when either of these values can be considered the valid option.

Null responses are appropriate if a subject does not complete an entire form or, for some reason, is not presented with a given question. These responses can be indicated on the data entry form as blanks. The implementation of null values in the database depends on the database computer software. Some associate null values with blanks. Other database software requires the use of a dummy value to represent a null value. For example, for a multiple-choice question with valid options A, B, and C, the additional value X can be used to indicate a null value. However, when analyzing these data, it is important to ensure that these null values are handled specifically as missing data. Certain fields, such as the fields that compose the unique identifiers in a form, should never assume null values. Null values also can be restricted from data fields that should be readily available, such as gender on the demographic form or fields considered of primary importance to the research questions being addressed.

In some instances, multiple answers may be given for a single question. In figure 4a the question prompts for a list of physical complaints. Only three response spaces are given in this example, but the subject may have all the complaints. In the analysis of these data, the absence of a particular code is intended to imply that the subject did not have that complaint. However, this fact cannot be distinguished from a missing or unknown value. A preferred format for this question is given in figure 4b, in which an answer is given for each complaint and a missing value can be assumed to have been omitted deliberately.

For some multiple-choice questions, an "other" option can be included in the list of valid answers. In conjunction with an "other" response, a small text field, where the option can be listed, should be included on the form. Although this text field need not be entered into the database, it is available for later review. If the "other" category is checked many times, the data forms can be updated to reflect additional options.

For multiple-choice questions, an "unknown" option can sometimes be included in the list of valid options (figure 4b). Whenever there is even a remote possibility that the subject will not know the response to a question, this option should be listed. The unknown option may not be appropriate for all questions, such as gender on the demographic form.

a. Limited Number of Responses

List Medical Complaints: ___ ___ ___

A. Headache B. Dizziness C. Pain

D. Fatigue E. Numbness F. Other _____

b. Preferred Format

Medical Complaints:	Circle One		
A. Headache	Y	N	U
B. Dizziness	Y	N	U
C. Pain	Y	N	U
D. Fatigue	Y	N	U
E. Numbness	Y	N	U
F. Other _____	Y	N	U

FIGURE 4. *Two formats for a question with multiple responses: limited number of responses and preferred format*

RELATIONAL DATABASE

For any large research project, a relational database structure is recommended (Date 1982, 1983). Many software packages are available to implement this structure on a mainframe, stand-alone, or networked personal computer. The minimal components of relational database software should include (1) procedures for creating tables, (2) procedures for developing the data entry process, (3) validation checks at data entry, (4) built-in security procedures, and (5) query language to facilitate retrieval of the data. As discussed in the "Null vs. Unknown" section above, the implementation of null values is also helpful.

In this type of database, the data can be viewed logically in tables. Each table contains information that, typically, is related in some way. In figure 2 the columns of the Demos table are shown. This table contains the intake demographic information for each subject. The Beck table, illustrated in figure 3, contains the information concerning a subject's level of depression. Each table comprises columns and rows. A column

corresponds to a single data field, or attribute. The values in the column are drawn from a predefined set of values. For example, the gender column in the Demos table will contain only the values M and F. Each row in a table is a collection of column attributes that describe a single study entity. A row in the Demos table describes a study subject, whereas a row in the Beck table describes the subject's depression on a given visit. To avoid use of data more than once, multiple instances of an identical row should not be included in a table.

There are several advantages associated with this type of database design. Viewing the data as a series of tables clearly presents the overall organization of the project's database, which helps facilitate the manipulation and retrieval of the data for reporting and analytic purposes. This is especially useful when a particular application requires that data be merged from several tables. Another important advantage is the ability to easily accommodate a nonstatic data system. It is not uncommon for research instruments and data fields to be added to a database after it has been implemented. The relational database structure is well suited to a dynamic environment where these changes can be made with little or no modification of preexisting reports, analyses, or procedures. Last, a rigorous set of guidelines, normal form theory (Dawson and Parker 1988; Kent 1983), has been developed that is helpful in designing this type of database. The set of properties contained in the guidelines ensures that the integrity of the data is maintained and results in an overall database design that is easier to manipulate for reporting and analysis purposes.

DATABASE TABLES

A useful guideline for creating the database tables is to associate a single table with a single form. Even in cases where a form might consist of only one or two pieces of information, a separate table is recommended. In this way the absence of a row in a given table can indicate that a particular subject's form is missing or has not been entered into the database. Each field in the form can correspond to an attribute in the table. In figures 2 and 3, tables are shown that are associated with each of the illustrated forms.

Within a form, a set of fields is noted that can be used to identify a single, specified data entry form. In general, the set of table attributes corresponding to these fields is called the key of the table and therefore uniquely identifies each row of the table. As examples, the attribute case number uniquely identifies a row in the Demos table (figure 2), and the combination of attributes, case number and visit, identifies a row in the Beck table (figure 3). Because the key identifies each row in a table, it

follows that no two rows in a given table will have identical entries for these key fields. The key attributes are used to retrieve specified single rows from a table. The key values also are used to merge data from two or more tables. Hence, for programming convenience, common key fields from multiple tables should be assigned the same variable name. For example, suppose a researcher were interested in knowing whether the mean Beck Depression Inventory score at intake was related to the age of the subject. Age at intake can be determined from fields in the Demos table (figure 2). The rows of the Beck table (figure 3) with the visit "I" could then be merged with the rows of the Demos table over the common field case number. Therefore, case number is the key value that allows all the information on a given subject to be merged.

NORMAL FORMS

As mentioned earlier, the relational database structure has a theoretical basis, normal form theory, that can be used to validate the design of tables in the database. The goal is to ensure that the integrity and quality of the data will be maintained throughout the process of updating and accessing the database. This theory, which to some extent is founded in common sense, also may be helpful in designing the data entry forms and creating the corresponding tables. Although it is beyond the scope of this chapter to describe this theory in detail, the following guidelines, which define three normal forms, are useful.

First Normal Form

To ensure the validity of updates and data retrieval, each column in a table should contain single data items whose values are selected from a predefined set of possible values. To illustrate a table that would violate this constraint, consider the form shown in figure 5a. In the table construction shown in figure 5b, a single three-character text column, "answers," is defined to contain the answers to the three multiple-choice questions. If a subject answered the three questions as B, D, and C, the value entered in this column would be the three-character value "BDC." Several practical difficulties are associated with this construction. Validating this field involves comparing its value with the set of all three-letter combinations of the letters A through F. To analyze or update the values for one of the three questions implies the user must take care to ensure that only the appropriate character is modified. A more reasonable approach, which satisfies first normal form, is shown in figure 5c. Here three columns are used that correspond to each of the three questions, "condom use," "intox during sex," and "needle sharing."

a. Form

HIV Transmission Risk Behaviors

Subject Case Number: ___ ___ ___ ___ ___

Date of Evaluation (mm/dd/yy): ___ /___ /___

1. Frequency of condom use in past 6 months: ___
   A. Always B. Most of time C. Half the time
   D. Seldom E. Never F. Not applicable/abstinent
2. Frequency of intoxication during sexual encounters: ___
   A. Always B. Most of time C. Half the time
   D. Seldom E. Never F. Not applicable/abstinent
3. Frequency of needle sharing: ___
   A. Always B. Most of time C. Half the time
   D. Seldom E. Never F. Not applicable/non-IV user

b. Table in Violation of First Normal Form

Table Name: HIV

case	evaldate	answers

Field Description	Variable Name	Key Fields	Data Type	Validity	Null
Subject Case Number	case	key	character-5		N
Date of Evaluation	evaldate	key	date		N
Answers	answers		character-3	(A-F)(A-F)(A-F)	Y

c. Preferred Table Construction

Table Name: HIV

case	evaldate	condom	intox	needle

Field Description	Variable Name	Key Fields	Data Type	Validity	Null
Subject Case Number	case	key	character-5		N
Date of Evaluation	evaldate	key	date		N
Freq Condom Use	condom		character-1	A-F	Y
Freq Intox During Sex	intox		character-1	A-F	Y
Freq Needle Sharing	needle		character-1	A-F	Y

FIGURE 5. *Illustration that shows violation of first normal form: form, table in violation of first normal form, and preferred table construction*

KEY: HIV=human immunodeficiency virus

Second normal form and third normal form depend on the identification of dependences that exist between or among data fields within a form. In this context, a dependence is defined as follows: Field X is dependent on field Y; if given the value of field Y, a single value for field X is retrieved. For example, in the Demos form (figure 2), date of birth is dependent on case number; that is, given a subject's case number, a single date of birth can be retrieved. Note that the converse is not necessarily true; that is, case number is not dependent on date of birth because, theoretically, several cases could be retrieved with the same date of birth. Dependences can be defined in terms of set of fields. In the Beck form (figure 3), a particular subject's monthly score is dependent on the combination of the case number and visit. This is because, given just the case number, several rows of Beck data theoretically could be retrieved because this form is collected repeatedly over time. However, given the combination of case number and visit, at most one row of data and hence one score will be retrieved.

Second Normal Form

Each field in the table should depend on the entire key and not a subset of fields in the key. Consider the form shown in figure 6, which was designed to collect weekly information concerning subject participation in certain self-help groups; a table associated with the form is also shown. The key for this table is the combination of case number and visit. The attributes associated with questions 1, 2, and 3 are dependent on both these key fields. In contrast, the last question or attribute is dependent solely on the case number because the answer to this question is constant over time. This table design therefore violates second normal form. If a coding error for the last question is made at one of the visits, these data are inconsistent for that subject and may later be reported incorrectly. In general, any field that is constant over time should not be collected in a one-to-many form. Collecting it more than once presents the chance of introducing an error in that field each time the data are collected and modified. The last question should be removed from the form shown in figure 6 and placed more appropriately in a form such as the one-to-one demographic form (figure 2).

Third Normal Form

All data fields that constitute a row in a table should be dependent on the key and on no other field. Consider now the form and table shown in figure 7. This form, collected at patient admission, contains five true/ false questions as well as the total number of questions answered as true. Because this is a one-to-one form, the case number is the unique identifier of the form and key of the corresponding table. However, note that the "total" field is dependent on the combination of the case number and the set of fields associated with all five true/false questions. Therefore,

Form

Weekly Participation in Self-Help Groups

Subject Case Number: ___ ___ ___ ___ ___

Date of Evaluation (mm/dd/yy): ___/___/___

Week Number (1-12): ___

1. Currently has a sponsor (Y/N): ___

2. Attendance in past week: ___
   A. Refuses to attend B. Less than required C. As required
   D. More than required E. Far exceeds required

3. Current attitude toward self-help group: ___
   A. Positive B. Neutral C. Negative

4. Age at attending first AA/NA meeting: ___

Table Definition

Table Name: Self-help

case	evaldate	visit	sponsor	attend	attitude	agefirst

FIGURE 6. *Form and table definition that violate second normal form*

KEY: AA/NA=Alcoholics Anonymous/Narcotics Anonymous

this table violates the third normal form. If one or more of the fields associated with questions 1 through 5 were updated and the total were not also updated, the data would be inconsistent. For this example, no information would be lost if the total field were eliminated altogether. For analysis purposes, the total can easily be determined by examining the values for questions 1 through 5.

In general, when the tables are constructed in the database, dependences between or among fields must be addressed. In some situations, when dependences are determined, fields can be moved to different tables or eliminated altogether. However, in some cases it may be helpful to maintain these dependences in the database. Consider the following two-part question:

Were you employed in the past month? (Y/N) _____

If Y, enter monthly net income. _____

Form

Treatment Attitude

Subject Case Number: ___ ___ ___ ___ ___

Date of Evaluation (mm/dd/yy): ___/___/___

	Circle One	
1. I frequently disagree with the therapist/counselor.	T	F
2. I expect treatment to be difficult.	T	F
3. I could stop using alcohol/other drugs if I wanted.	T	F
4. I go to an AA meeting at least once a week.	T	F
5. I have friends who support me.	T	F

Total number of true answers (1-5): _____

Table Definition

Table Name: Attitude

case	evaldate	q1	q2	q3	q4	q5	total

FIGURE 7. *Form and table design that violate third normal form*

KEY: AA=Alcoholics Anonymous

Here income is nonnull only when the first question is answered Y. Because this field depends, in part, on the value in the employment field, this is a violation of third normal form. However, it is inconvenient in this case to move the second field to a separate table. The documentation of the table that lists these data should include a note of this dependence. The correctness of the combination of the fields also can be reviewed in periodic validation checks.

DOCUMENTATION

The properties of each table must be clearly documented. Examples of table documentation are shown in figures 2 and 3. The fields that constitute the key should be clearly noted. Possible values that a field can assume should be enumerated. Fields that cannot assume null values should be noted. Validity checks between or among tables also can be documented.

Two Necessary Forms and Tables

For research projects that study subjects over time, forms associated with each subject's start and completion of treatment should be included in the protocol. Minimally, the intake form should include the subject's case number and date of admission. Other information collected at intake can be included (figure 2). Similarly, a discharge form must include the case number and date of termination but also can contain information pertaining to the type of discharge. When these two tables are merged over their common field (the case number) the current status of all subjects can be determined: Those listed in both tables are no longer in treatment; those listed in the intake table only are still active. This concept can be extended to other phases of treatment as well. For example, a form can be incorporated to indicate the termination of the followup phase of treatment.

At any time, the intake form is the minimal set of information included in the database about a given subject. When the database is queried, the table associated with the intake form lists all the subjects currently admitted into the study. Hence, the case numbers in this table can serve as the links to all other tables in the database. This is especially useful if the protocol varies by subject, which implies that not all subjects will be listed in every table. For this reason, it is suggested that, before any other form can be entered for a given subject, a check be made that a row in the intake table exists for that subject.

Validation

Validation of the data can be implemented in several ways and at multiple times. During data entry, the value entered for a particular field can be compared with the list of valid entries. An error message can prompt the data entry technician to correct an invalid entry. Before an entire form's data are appended to its appropriate table, within-form validity checks can be made. For example, if a data field indicates that the subject is a nonsmoker, the field containing the packs-per-day data should be null. In some cases, fields within a form can be compared with data already included in the database. For example, as described in "Two Necessary Forms and Tables" above, a form's case number can be compared with the list of case numbers in an intake table to ensure that intake data have already been entered. If the intake data for that case are missing, an error message can be shown.

Although the data that have been entered can be considered valid, they still may differ from data on the data entry form. When personnel resources are available, double entry of the data can be implemented to

help eliminate this level of data entry error. Here, two different data entry technicians independently enter the data. In a timely fashion, the data are reconciled. An outline of an algorithm that has been used to implement this type of system is shown in figure 8. Another option involves letting a reviewer who is not involved with the data entry compare the data in the database with the corresponding data entry form. With either of these systems, entry errors are noted and promptly corrected.

Regular validity checks should be made on the entire database. This overall check can reexamine the validity of each field, within a form and between or among forms, depending on how the rules are defined for data entry purposes. This is especially important when dependences, as discussed above in "Database Tables," are maintained in the database. In addition, more elaborate checks, either between or among forms, can be made. For example, if a subject at admission indicated she was in the third trimester of pregnancy, forms associated with the birth of her infant should be included in the database within a reasonable amount of time. A check also can be made to ensure that all forms associated with the subject's protocol are entered in the database in a timely fashion; that is, within 3 months of admission, the month 2 Beck data (figure 3) should be entered. Last, when a subject completes the treatment program, there should be an overall check that data from all appropriate forms listed in the protocol, from intake to discharge, have been entered.

SECURITY AND PROTECTIONS

To maintain the integrity of the data, as well as ensure patient confidentiality, security can be incorporated at several levels. Although the implementation of these security procedures depends on the software and computer environment, the guidelines are the same. Only authorized users have permission to access the database. A user who can access the database may then have permission to access all the tables in the database or only a subset of these tables. Certain users may be restricted from accessing tables containing highly sensitive data. The type of access also can be specified for a particular table. Some users may have permission to retrieve, but not modify, data from a given table. Last, but important, a system for periodically backing up the database also must be incorporated and clearly documented. In that way, if there is a computer hardware failure, the data can be restored.

Data initially entered into table 1 and, independently, doubly entered into table 2.
The final table contains only the reconciled data.

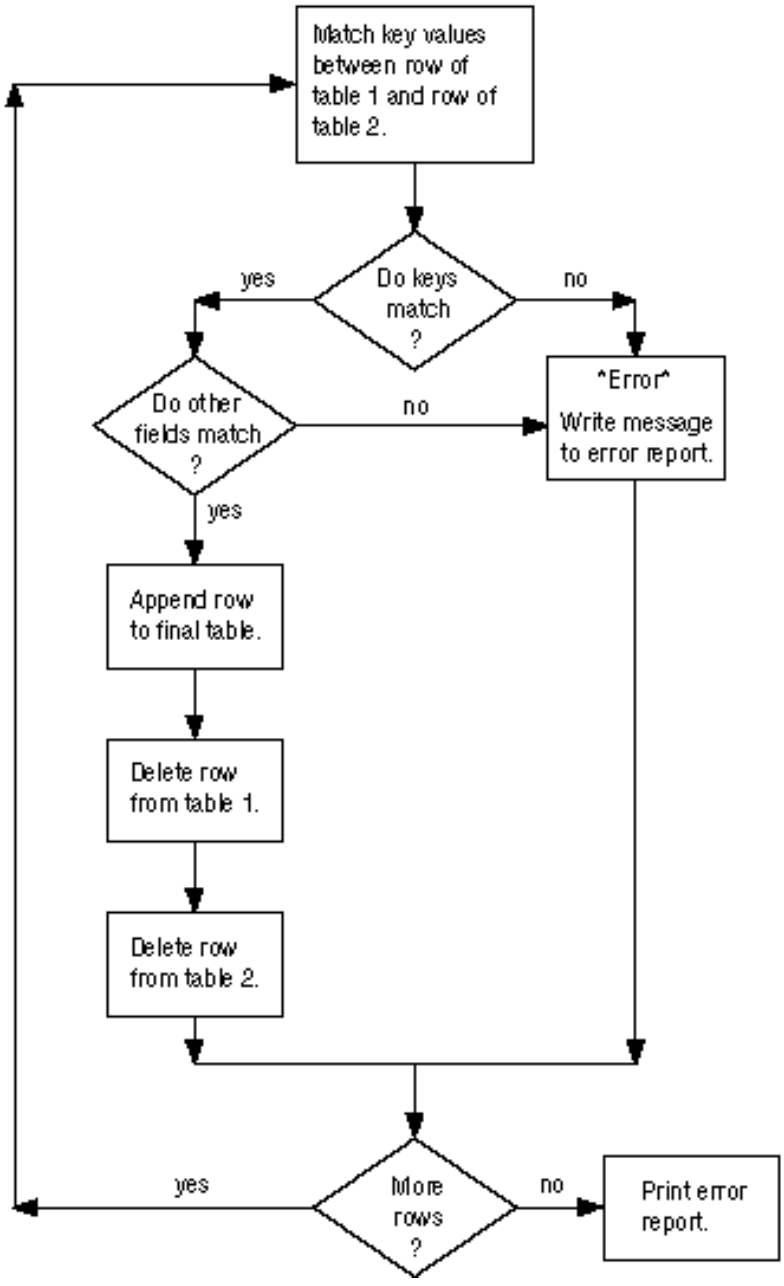

FIGURE 8. *Double-entry reconciliation algorithm*

REFERENCES

American Psychiatric Association. *Diagnostic and Statistical Manual of Mental Disorders: Fourth Edition.* Washington, DC: American Psychiatric Press, 1994. 886 pp.

Beck, A.T.; Steer, R.A.; and Gabin, M.G. Psychometric properties of the Beck Depression Inventory: Twenty-five years of evaluation. *Clin Psych Rev* 8(1):77-100, 1988.

Date, C.J. *An Introduction to Database Systems.* 3d ed. Reading, MA: Addison-Wesley, 1982. 574 pp.

Date, C.J. *An Introduction to Database Systems, Vol. II*. Reading, MA: Addison-Wesley, 1983. 383 pp.

Dawson, K.S., and Parker, L.M.P. From entity-relationship diagrams to fourth normal form: A pictorial aid to analysis. *Computer J* 31:258-268, 1988.

Kent, W. A simple guide to five normal forms in relational database theory. *Commun ACM* 26:120-125, 1983.

McLellan, A.T.; Luborsky, L.K.; Cacciola, J.; Griffith, J.; McGahan, P.; and O'Brien, C.P. *Guide to the Addiction Severity Index: Background, Administration, and Field Testing Results*. DHHS Pub. No. (ADM)88-1419. Rockville, MD: National Institute on Drug Abuse, 1985.

World Health Organization. *International Statistical Classification of Diseases and Related Health Problems. 10th Revision.* Geneva: World Health Organization, 1992.

AUTHORS

Kathryn S. Dawson, Ph.D.
Senior Programmer Analyst
Department of Biostatistics
(804) 828-9824 (Tel)
(804) 828-8900 (Fax)
dawson@gems.vcu.edu (Internet)

Sidney H. Schnoll, M.D., Ph.D.
Professor
Department of Internal Medicine and Psychiatry
Chairman
Division of Substance Abuse Medicine
Medical College of Virginia
(804) 828-9914 (Tel)
(804) 828-9906 (Fax)
sschnoll@gems.vcu.edu (Internet)

Virginia Commonwealth University
1101 East Marshall Street, Box 980032
Richmond, VA 23298-0032

Click here to go to page 272