Relational Database Management: What You Need to Know

Save to myBoK

by Donald T. Mon, PhD

Relational databases offer great flexibility in working with data. Here's an overview of how they work.

Patient care, research, and education are data-intensive, information-rich processes. If data is captured electronically, it is usually stored in a system with an underlying database. One of the most common types of databases is the relational database. What is a relational database, and how does it work? This article answers these questions and tells you why HIM professionals should take note.

Databases: The System's Backbone

Administrative, financial, and clinical data are collected, stored, and manipulated in healthcare for a variety of purposes. These include:

- assisting work flow activities, such as admitting, bed management, orders, results, and billing (where a discrete activity within each process is, in technical terms, a transaction)
- helping clinicians make decisions while they are treating patients (concurrent decision making)
- helping managers and clinicians make decisions based on post-treatment historical data (retrospective decision making)
- developing management reports
- exporting data files to regulatory agencies or benchmarking organizations
- supporting process improvement studies, as well as clinical, health services, and outcomes research

A database is simply a collection of data organized into a structure that facilitates data collection, retrieval, reporting, and analysis. Three common types of databases are hierarchical, network, and relational databases. This article will focus on the relational database, one of the most common types used in healthcare information systems (HIS) and HIM.

Why Relational Databases Matter to HIM

Understanding relational database technology will help HIM professionals enhance their interaction with IT professionals, informaticians, and administrators. It can also help them assume some of the key e-HIM knowledge worker responsibilities needed in the future.

Working with IT staff, end users, and managers, HIM professionals can fully participate in re-engineering work flow and add further value by defining the data to be captured as well as identifying how data quality and patient confidentiality can be enhanced through the relational structure. Such activities can help HIM professionals enhance their roles in maintaining privacy and security and expand into process re-engineering roles at the same time.

In addition, HIM professionals can support concurrent and retrospective clinical decision making, as well as electronic health record (EHR) system selection and implementation, by understanding how the data is currently stored in the paper record, identifying how the data must be defined in a decision support system (DSS) or EHR, recommending how data quality can be enhanced and privacy maintained through the data structure, and extracting the data. These activities enhance customer service, strengthen ties with clinicians, and expand the role of HIM professionals as data managers and data analysts.

Finally, by understanding relational database technology, HIM professionals can assist managers, researchers, and informaticians in developing new or enhancing existing management reports, extracting data to report to regulatory agencies or benchmarking organizations, and conducting internal process improvement studies.

Relational Databases 101

In a relational database, data is organized in a structure of related tables. A table consists of rows and columns, much like a spreadsheet. Each row is simply a record of data on some person (e.g., a patient, a physician), place (e.g., an operating room, the emergency department), or thing (e.g., devices, supplies, costs, charges) within that table.

A table record is not the same as an entire patient record, which, as will be shown later, is spread across many tables in a relational database. Each column is a field that describes an attribute of the record.[1] For example, fields for a table record might include the patient's last name and first name, primary diagnosis, each of the secondary diagnoses, admit date, the attending physician, the consulting physician, and so on. (See "A Flat File Structure".)

a flat file structure

Medical Record Number	Encounter Number	Pt Last Name	Pt First Name	Pt Middle Initial	Pt Date of Birth	Pt Gender	UPIN	Phys Last Name	Phys First Name	Phys Middle Initial	Phys Specialty	License Number	Principal Dx	Sec. Dx1	Sec. Dx2	Sec. Dx3	Sec. Dx4	Sec. Dx5
1234	1	Doe	John	M	8/15/53	M	31200	Jones	John	H	Cardiology	IL-4312	428.0	276.5	287.3	599.0	250.01	412
5678	1	Ward	Susan	J	6/10/78	F	23300	Smith	Elizabeth	A	Gastroenterology	IL-7087	574.71	577.0	401.9	278.00	305.1	
9100	1	Doe	John	M	7/26/62	M	11000	Yee	Xiao	V	Oncology	IL-8702	V58.1	162.9	493.90	272.4	789.1	
1112	1	Public	John	Q	11/11/76	M	60611	Payne	Joseph	F	Infectious Disease	IL-1782	042	136.3	176.0	176.8	311	
1314	1	Young	Sarah	K	3/15/81	F	58001	West	Natalie	U	Obstetrics	IL-6478	645.11	656.31	648.21	285.1	660.41	V27.0
1516	1	Young	Joseph	M	8/25/60	M	48258	Bonecutter	Dwight	H	Orthopedics	IL-6793	717.2	717.6	715.96	250.01		
1718	1	Kong	Li	K	9/22/48	M	48258	Bonecutter	Dwight	H	Orthopedics	IL-6794	816.11	813.18	493.90			
1920	1	Doherty	James	D	12/20/45	M	48258	Bonecutter	Dwight	H	Orthopedics	IL-6795	840.4	726.10				
2122	1	Ramirez	Esteban	A	12/20/45	M	73539	Rex	Ray	B	Radiation/Oncology	IL-4809	V58.0	185	414.01	403.91		
2122	2	Ramirez	Esteban	A	12/21/45	M	73539	Rex	Ray	B	Radiation/Oncology	IL-4810	V58.0	185	414.01	403.91		
2122	3	Ramirez	Esteban	A	12/22/45	M	73539	Rex	Ray	B	Radiation/Oncology	IL-4811	V58.0	185	414.01	403.91		
2324	1	Ritelli	John	L	5/17/53	M	59494	Kirk	James	T	Endocrinology	IL-8102	250.11					
2324	1	Ritelli	John	L	5/18/53	M	43849	Starr	Terry	N	Pulmonology	IL-2395	487.1					
2324	2	Ritelli	John	L	5/19/53	M	19029	Agra	Vi	C	Urology	IL-3939	250.41	583.81				
2324	3	Ritelli	John	L	5/20/53	M	34945	Bradshaw	Bart	D	Neurology	IL-6595	256.61	357.2				
2526	1	Doe	Jane	F	1/10/68	F	59152	Jones	John	N	Rheumatology	IL-3839	727.1	715.97	735.4	754.59	754.52	

It's a common mistake to refer to a large single table as a database. Because a single table has only two dimensions, it is more technically correct to refer to it as a flat file. A relational database, on the other hand, must consist of two or more related tables, with the more complex databases or systems (e.g., a data warehouse or a clinical data repository) containing hundreds to well over a thousand tables.

Designing a relational database structure that best suits user needs is both art and science. It's a science because there are specific rules to normalize the data (i.e., group data into tables and establish relationships between them). And it's an art because defining the data and how the data are perceived to relate to each other is certainly dependent on the functions required—but still often in the eyes of the beholder.

When designing a database, it is quite common for database administrators (DBAs), HIM professionals, and end users to discuss many variations of a database structure before agreeing upon one—there is that much diversity in thought or opinion on these two issues. The following example illustrates the process of designing a simple relational database.

Grouping the Data

The first step in physically designing a relational database, following the aforementioned normalization rules, is to group the data into a set of tables where each table consists of some soon-to-be-derived logical cluster of fields with a descriptive table name (see "A Simple Relational Database").[2]

a simple relational database

Table Name: Patient

Medical Record Number	Last Name	First Name	Middle Initial	Date of Birth	Gender
1234	Doe	John	M	8/15/53	M
9100	Doe	John	M	7/26/62	M
2526	Doe	Jane	F	1/10/68	F
1920	Doherty	James	D	12/20/45	M
1718	Kong	Li	K	9/22/48	M
1112	Public	John	Q	11/11/76	M
2122	Ramirez	Esteban	A	12/20/45	M
2324	Ritelli	John	L	5/17/53	M
5678	Ward	Susan	J	6/10/78	F
1314	Young	Sarah	K	3/15/81	F
1516	Young	Joseph	M	8/25/60	M

Table Name: Doctor

Unique Physician ID #	Last Name	First Name	Middle Initial	Specialty	License Number
19029	Agra	Vi	C	Urology	IL-3939
48258	Bonecutter	Dwight	H	Orthopedics	IL-6793
34945	Bradshaw	Bart	D	Neurology	IL-6595
31200	Jones	John	H	Cardiology	IL-4312
59152	Jones	John	N	Rheumatology	IL-3839
59494	Kirk	James	T	Endocrinology	IL-8102
60611	Payne	Joseph	F	Infectious Disease	IL-1782
73539	Rex	Ray	B	Radiation/Oncology	IL-4809
23300	Smith	Elizabeth	A	Gastroenterology	IL-7087
43849	Starr	Terry	N	Pulmonology	IL-2395
58001	West	Natalie	U	Obstetrics	IL-6478
11000	Yee	Xiao	V	Oncology	IL-8702

Tables are joined by their keys.

Table Name: Visit/Stay

Medical Record Number	Encounter Number	UPIN	Principal Dx	Sec. Dx1	Sec. Dx2	Sec. Dx3	Sec. Dx4	Sec. Dx5
1234	1	31200	428.0	276.5	287.3	599.0	250.01	412
5678	1	23300	574.71	577.0	401.9	278.00	305.1	
9100	1	11000	V58.1	162.9	493.90	272.4	789.1	
1112	1	60611	042	136.3	176.0	176.8	311	
1314	1	58001	645.11	656.31	648.21	285.1	660.41	V27.0
1516	1	48258	717.2	717.6	715.96	250.01		
1718	1	48258	816.11	813.18	493.90			
1920	1	48258	840.4	726.10				
2122	1	73539	V58.0	185	414.01	403.91		
2122	2	73539	V58.0	185	414.01	403.91		
2122	3	73539	V58.0	185	414.01	403.91		
2324	1	59494	250.11					
2324	1	43849	487.1					
2324	2	19029	250.41	583.81				
2324	3	34945	256.61	357.2				
2526	1	59152	727.1	715.97	735.4	754.59	754.52	

For example, a table named "patient" will likely contain fields about patients. The best way to form each table is to state a relation between two table names at a time. The following statements express relations between tables of data typically found in a healthcare setting:

- A patient may have one or more visits and/or stays.
- A patient may be treated by one or more doctors during a visit and/or stay.

If those relationships sound logical, the next question to ask is, "What are the attributes that are associated with patient, doctor, and visit/stay?" The attributes that are derived then become the fields assigned to the associated table. The fields and tables in this example might include those shown in "A Simple Relational Database."

Aside from the obvious fact that there is more than one table, there are other important differences between the two figures. First, no group of data is repeated as the patient data (rows 9 through 15), and doctor data (rows 6 through 11) are in "A Flat File Structure."

Second, each record in each table has a unique identifier (UID), or primary key. In this example, the keys for the patient and doctor tables are the medical record number and the unique physician identification number (UPIN) respectively. In the visit/stay table, the key is a combination of the medical record number and the encounter columns; note how the combination of these two fields form a unique string of characters from row to row.

Third, each field in its respective table is dependent on that table's primary key—which is a technical way of saying that the field must have a logical and primary relationship with the unique identifier, or else the field doesn't belong in the table. For example, the physician specialty field is primarily and more logically related to the UPIN key because it describes the doctor's area of expertise. It does not have as close a relationship to the patient's medical record number key. Therefore, the physician specialty field belongs in the doctor, not the patient, table.

Fourth, the tables can be joined in a variety of combinations to produce the desired information. For instance, by linking the keys between the tables in "A Simple Relational Database" and looking at their respective records, among other pieces of information, the following can be gleaned (without any knowledge of the diagnosis codes):

- The number of, and the specific, patient(s) treated by specialty. For example, only one patient was treated in the oncology department; of the two John M. Does, the one with medical record number 9100 was the specific patient treated
- The number of, and the specific, patient(s) with more than one visit or stay. In this example, there were two such patients, specifically Esteban Ramirez and John Ritelli
- The specific doctors treating the same patient either in the same encounter, or over multiple encounters. For example, Drs. Kirk, Starr, Agra, and Bradshaw treated patient John Ritelli
- The number of patients treated by a specific physician. For example, Dr. Bonecutter treated three patients in the relevant time period

What Are the Advantages of Relational Databases?

Even through this simple example, the advantages of a relational database are clear. Note that the structure of a relational database is independent from any particular application. The data is not necessarily ordered by the way it is captured, displayed on a computer screen, or reported.

For DBAs, software developers, and HIM professionals, this point is one of the most important aspects of relational databases. Such independence provides great flexibility in designing an information system, generating a report, and querying the data, among other things.

Consider such flexibility: For an admitting transaction, a computer screen may be designed to capture patient demographics, as well as some attributes about the stay, and link the patient to an admitting physician, thus using all three tables in "A Simple Relational Database." But to simply look up a patient, only the patient table needs to be accessed. The medical staff office, on the other hand, to update its records on physician license renewals, can simply use the doctor table. However, when measuring outcomes by specialty, the doctor and visit/stay tables can be used.

There are many other advantages as well. By restricting access to certain tables, patient confidentiality can be protected and de-identification can be enforced. For instance, in the outcomes measurement example above, specific patient names, addresses, and other protected health information are not available because the patient table is not accessible to the user. However, should there ever be an instance where it is necessary to identify a specific patient, such identification can be done by joining the patient table.

Further, the relational database structure can increase operational efficiency and enhance data quality at the same time. Remember, relational database groups of data are not repeated. Thus, the data is entered once (presumably correctly), stored in one place, but used for multiple purposes. This feature allows data to be entered as far upstream in the work flow as possible, saving those downstream in the work flow from duplicate, and perhaps inaccurate, entry, but still allowing them to use such data. Moreover, because the data is stored in one place, anyone or any system function accessing the data will receive the most accurate and updated information.

It's to an HIM professional's advantage to understand and make the most of relational databases for these reasons and more. Understanding these databases allows HIM professionals to add value. In doing so, they make important contributions to their organizations, strengthen relationships with customers and colleagues in other departments, gain further visibility, and open new avenues of opportunity for themselves. This path can help add to the worth of the HIM professional in the organization now in the world of paper records and the future in the e-HIM environment.

Notes

1. End users (e.g., managers, clinicians, researchers) may use terms such as “data elements” or “variables,” when they refer to fields. Understanding such nuances in terminology will facilitate communication between HIM professionals and their customers.

2. For brevity’s sake, all of those rules cannot be discussed in this article, nor can each rule be discussed in great detail. For more details, please refer to the wealth of literature that describe the rules more comprehensively, including those noted in the reference list.

References

Codd, E.F. “A Relational Model of Data for Large Shared Data Banks.” *Communications of the ACM* 13, no. 6 (1970): 377-387.

“Codd’s Twelve Rules for Defining a Relational Database.” Available at http://www.cis.ohio-state.edu/~sgomori/570/coddsrules.html.

Halpin, T.A. *Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design*. San Francisco: Morgan Kaufman Publishers, 2001.

Poolet, Michelle. “SQL by Design: Why You Need Database Normalization.” *SQLServer Magazine*, August 1999. Available at http://www.sqlmag.com/Articles/ Index.cfm?ArticleID=4887&pg=1.

***Donald T. Mon** (donald.mon@ahima.org) is vice president of practice leadership/e-HIM products and services at AHIMA.*

Article citation:
Mon, Donald T. "Relational Database Management: What You Need to Know." *Journal of AHIMA* 74, no.10 (November 2003): 40-45.