Chapter 10: Data Storage Design

# Teaching Tips and Strategies *(from Barbara Wixom)*

Some of these topics may have been covered already if your school has a separate class for data management. For example, students at my school already have a pretty decent knowledge of the various types of files and databases. I usually try to cover this chapter in 2 sessions, one session focusing on data storage formats and designing for storage efficiency, and the other session focusing on designing for access speed.

During the first class I briefly go over the various data storage formats, and there are a couple of exercises you may want to try. You can introduce the different file types and then have the students do Your Turn 10-1 in class. This is a good way for them to see how files are still used in a modern context, while also conveying the different kinds of files. When you present the different kinds of databases, you may want to make sure you talk about the current players in each market. You can then have the students do Your Turn 10-2 and talk about when each database would or would not be an appropriate solution for the system described.

Next, I focus on how to optimize data storage formats. We first talk about why it is important to optimize data storage – it helps to throw out statistics for some really large systems to give students some perspective. We talk about how large databases (like Walmart’s) would increase by having redundant fields throughout tables. Then, I review the process of normalization as a way to make sure that data storage is optimized.

I then spend the second class session talking about optimizing storage for speed. You can emphasize the importance of speed by asking the students how they would feel if it took 5 minutes to bring up an email message, etc. It usually is easy for them to understand how critical speed is for user acceptance of systems – because they themselves are impatient users.

After a quick lecture on denormalization, you can have your students try to denormalize the student activity file in the Your Turn exercise. Then I lecture about clustering and indexing. I like to use the grocery store example for clustering and a book index example for indexing – both seem to drive the concepts home for the students.

My students like to do the size estimation as a hands-on exercise – it makes for a good homework assignment. We use Erwin software, which has a volumetrics component. I either give them a model in Erwin, or I have them use the model that they developed for their semester project. Then, they have to develop assumptions and perform volumetrics on the model to estimate the future size of the database. You can actually have them post the results electronically and have the class discuss why the sizes of the group’s databases differ. It becomes clear how important assumptions are, and students also see how factors above and beyond raw data (indexes, administrative overhead) affect the size of the database.

I wrap up data storage by discussing trade-offs between speed and storage efficiency. We talk about how tough it is to balance the two issues. You may want to talk about the popularity of data warehouses. The fact that companies are moving data from the operational systems into decision support repositories demonstrates how systems that update information need to be optimized differently than systems that run read-only, ad hoc queries. You can talk about how a company would optimize its data warehouse versus its transaction processing system.

# War Stories *(from Barbara Wixom)*

## Indexes with Peoplesoft

Even though Peoplesoft is a packaged system, developers still need to worry about indexing. My friend Graig explained some of the issues with the Human Resource module of Peoplesoft:

The HR module is set up so that the indexes exist based on the way the screens are set up. For example, if a screen sorts people by state, then Peoplesoft comes with a state index already created. The problem is that some companies need to change screens or the users may search for people differently than the standard format; some users may want to look people up by zip code or city. In these cases, the system would be *very* slow because there are no indexes that exist to list people in that order.

Graig explains that developers have to be aware of how users will actually *use* the system and then create indexes to make sure the system is built for that use. Ideally, developers should proactively discover the indexes that need to be created before the system is put in place. They can determine this by doing walk-throughs of the system, observing the way users currently perform processes, and talking with users.

Be careful, Graig warns, you can have too many indexes. In fact, in PeopleSoft after the 4th index on a table, you will begin to see performance decrease when users are updating data in the system. Sometimes you have to make trade-offs between slow reports and slow updates. He suggests that developers try to create reports and screens so that they can share indexes whenever possible.

Ideally, you should use test data and test the system before releasing it to users, paying careful attention to how fast the different operations are. Unfortunately, sometimes problem spots don’t surface until after the system is live and data volumes increase. In these cases, developers just need to be aware that index changes may have to be made…

To give you an idea, Graig recently worked on a report that took 5 minutes to run because there was a join involved, and no indexes existed for the join. An index was created, and the same report ran in 15 seconds.

## Multi-dimensional Model for 3M

Here is an example of the way 3M Corporation looks at its data multi-dimensionally. This was taken from a presentation they made at the 1999 Data Warehousing Institute Best Practices competition.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Detail** | **Product** | **Customer** | **Geography** | **Time** |
| Sales | Total 3M | Market  (retail) | All  (global) | YTD/LYTD |
| Margin | Business Unit (office) | Channel  (Mass Merch) | Region  (Americas) | Quarterly |
| Price | Product line (Post-it) | Company  (Walmart) | Country  (Mexico) | January |
| Demand | Product  (pads) | Site  (Store 7878) | State  (Mexico City) | Week 3 |
| % on time | SKU  (3x5 yellow) | Account  (stationery dep) | Sales Territory  (NW Sanchez) | Mon 1/19 |

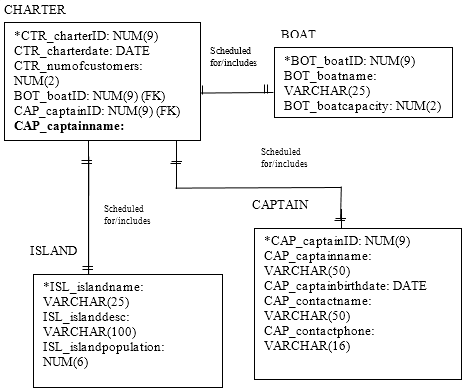
# Answer to Your Turn 10-1: Student Admissions System

Student answers will vary, one solution follows:

* Master file: Student information file which contains student name, address, contact info, degree interest, etc., will be updated as needed.
* Look-up file: State name validation file which contains state names, used to validate application input.
* Transaction file: Student application files containing applications for a specific time period. Will be used to update the master file either as needed or on a scheduled basis.
* Audit file: Contains requested changes to student information file, who requested the change, and when. Files are kept for a pre-determined amount of time.
* History file: Contains student information, name, address, contact info, degree interest, and date of graduation. File is a yearly history, and files over 10 years old are archived to off-site storage.

# Answer to Your Turn 10-2: Island Charters

Student answers will vary based on their response to “Your-Turn 5-7”. When considering how denormalization might occur, students should determine what data, if any, would reduce the time needed to perform queries for reporting or decision making. One example would be to include the Captain’s Name in the Charter table. When accessing information on specific charters, the Captain ID and name would be relevant to determining how often a specific captain was scheduled. For that type of reporting, the Captain’s birth date and contact information would not be needed.



# Answer to Your Turn 10-3: Donation Tracking System

1. Primarily, this can be considered to be a type of Decision Support System (DSS).
2. It might also be considered a transaction system as it intends to solicit and track donations. However, the development officers hope to determine donation patterns in order to improve rates, which indicate that it will be used to support decision making.
3. The data being used includes text, numbers, and date formats.
4. The best choices for a DSS are relational and multidimensional databases that will support queries on an ad-hoc basis.

# Answer to Your Turn 10-4: Denormalizing a Student Activity File

The replies from the students will vary based on the output initially provided by Your Turn 5-7 completed during Chapter 6 activities. This is an opportune time to review the reasons, purposes, and deliverables that may exist when converting logical ERDs to the physical model.

# Answer to Concepts in Action 10-A: Mail-Order Index

Ideally, the requirements for data design will be one of the results of the systems requirement phase. During this phase, users should be consulted as to what type of data, what combinations of data, and the response time needed in order for them to do their job. User needs may not coincide with the ‘standard’ methods of indexing data or retrieving data. In that case, the designer should customize how the data is indexed to optimize retrieval.

# Solutions to End of Chapter Questions

1. *Describe the two steps to data storage design.*

The first step is to select the appropriate format for the data storage. There are several different methods of storing data (files, relational data bases, multi-dimensional databases, object-oriented databases) and the analyst should select the one that will provide the best approach to storing the system data. Second, the data storage must be designed to optimize its processing efficiency, which involves considering how the data will be used, and making the appropriate design decisions.

1. *How are a file and a database different from each other?*

Files are essentially an electronic list of data that have been optimized to perform a particular transaction and the information is changed and manipulated by programs that are written for those purposes. . Typically, fi les are organized sequentially, and new records are added to the file’s end. A databaseis a collection of groupings of information that are related to each other in some way (e.g., through common fields). A database management system (DBMS) is software that creates and manipulates these databases

1. *What is the difference between an end-user database and an enterprise database? Provide an example of each one.*

*End-user DBMS* s such as Microsoft Access support small-scale databases that are used to enhance personal productivity, and *enterprise DBMS*s, such as DB2, SQL Server, and Oracle, can manage huge volumes of data and support applications that run an entire company. An end-user DBMS is significantly less expensive and easier for novice users to use than its enterprise counterpart, but it does not have the features or capabilities that are necessary to support mission-critical or large-scale systems. .

1. *Name five types of files and describe the primary purpose of each type.*

Master files store the business’s or application’s core data. The data in a master file is considered fairly permanent, does not change much, and is usually retained for long periods. Look-up files contain static values used primarily during validation processing. A list of valid code values for a field might be referred to during data entry to ensure that a valid code was entered. Transaction files contain information that will be used to update a master file. These files are usually temporary in nature; they are used to collect transactions, the transactions update the master file, and then the transaction files are archived. Audit files are used to help trace changes made to files. An image of a record before and after a change may be written to an audit file so that the change is documented. History files serve as archives for older information that is rarely used. Typically, the file is stored off -line, yet it can be accessed on an as-needed basis.

1. *Name two types of legacy databases and the main problems associated with each type.*

Hierarchical databases use hierarchies, or inverted trees, to represent relationships. The main problem with this database model is that it cannot be used efficiently to represent nonhierarchical associations. *Network databases* (e.g., IDMS/R, DBMS 10) are collections of records that are related to each other through *pointers.* Both kinds of legacy systems can handle data quite efficiently, but they require a great deal of programming effort. The application programs must understand how the database is built and must be written to follow the structure of the database.

1. *What is the most popular kind of database today? Provide three examples of products that are based on this database technology.*

Relational databases are most popular today due to their ease of use. Examples of relational DBMSs on the market include MS Access, Oracle, DB2, Sybase, Informix, MS SQL Server, and MySOL.

1. *What is referential integrity and how is it implemented in a relational database?*

Most relational database management systems (RDBMSs) support *referential integrity*, or the idea of ensuring that values linking the tables together through the primary and foreign keys are valid and correctly synchronized. For example, if a customer is placing an order, we need to have information on the customer in the customer table. The RDBMS will check to see if there is a record for that customer in the Customer table before it will let an order be entered. Checking for known required relationships helps assure referential integrity.

1. *What is the biggest strength of the object database? Describe two of its weaknesses.*

The biggest strength of object databases is the ability to handle complex data, sucha . (e.g., images, video, and audio). Two weaknesses of object databases are skills are hard to find and the limited acceptance in the marketplace.

1. *How does the multidimensional database store data?*

Multidimensional databases store data using several dimensions. Data may be aggregated and/or detailed, depending upon the access needs of the users.

1. *What are the two most important factors in determining the type of data storage format that should be adopted for a system? Why are these factors so important?*

First, evaluate the type of data that will be stored. Relational databases are the standard for simple data such as numbers, text, and dates. If the data is more complex (video, images, or audio), then object databases may be required. If the data needs to be aggregated, then multidimensional databases are recommended. The second factor is the type of system being developed. Transaction processing systems require rapid update and retrieval capability, and will best be constructed using files, relational databases, or object databases. Decision support types of applications require rapid access to data in ad hoc ways. These types of systems are best implemented using relational or multidimensional databases. These two factors are very important because you must select a data storage format that is suitable for the data the system will include and the uses planned for that data.

1. *Why should you consider the storage formats that already exist in an organization when deciding upon a storage format for a new system?*

This factor is important because the project team needs to be aware of the existing base of technical skills that are available to work with the data storage format. If a data storage format is chosen that is new to the organization, then the team must allocate training and learning time into the project schedule.

1. *What are the differences between the logical and physical ERDs?*

Like the DFD, the ERD contains the same components for both the logical and physical models, including entities, relationships, and attributes. The difference lies in the fact that *physical ERDs* contain references to exactly how data will be stored in a file or database table and that much more metadata is added to the CASE repository to describe the data model components.

1. *Describe the metadata associated with the physical ERD.*

Metadata included in the physical ERD includes information regarding attributes such as data type, field size, format, default values, primary keys, and foreign keys.

1. *Describe the purpose of the primary and foreign keys.*

*Primary keys* are fields that contain a unique value for each record in the file or table. A foreign key is the primary key field(s) from one table that is repeated in another table to provide a common field between the two tables..

1. *Name three ways that null values in a database can be interpreted. Why is this problematic?*

A null value in a field can indicate that there should not be a value in the field (i.e., blank is correct). It can also mean that an error was made, and a value that should have been entered was incorrectly omitted. It can also indicate that a value for the field has been deleted, which may or may not be correct. The difficulty in really knowing why the null exists is the major problem with nulls.

1. *What are the two dimensions in which to optimize a relational database?*

The two dimensions in which to optimize a relational database are for storage efficiency and for speed of access.

1. *What is the purpose of normalization?*

The purpose of normalization is to optimize the data storage design for storage efficiency. Normalization helps ensure that data redundancy and null values are kept to a minimum.

1. *Describe three situations that can be good candidates for denormalization.*

Denormalization is performed to speed up data access. Redundancy is added back into tables in order to reduce the number of joins that are required to produce the desired information. In a normalized Order table, the customer name will not be included; however it may be added back in to the Order table to improve processing speed. This represents a situation in which some parent entity attributes are included in the child entity. Similarly, a look-up table of zip codes and states may be set up in the normalized data model, but could be added back in to the physical model design. Another situation is where a table of product codes lists the description and price. These may also be added back into the physical model to improve application performance. Lookup tables are common candidates for denormalization. Finally, 1:1 relationships may be good candidates for denormalization, since the information may be accessed together frequently.

1. *Describe several techniques that can improve performance of a database.*

Denormalization adds selected fields back to tables in a data model. This adds a little redundancy, but improves the data access speed. Clustering involves physically placing records together so that like records are stored close to each other. Indexing creates small, quickly searchable tables that contain values from the table and indicate where in the table those values can be found. Finally, proper estimation of the data set size is important to assure that adequate hardware is obtained for the system.

1. *What is the difference between interfile and intrafile clustering? Why are they used?*

*Interfile clustering* combines records from more than one table that typically are retrieved together. With *intrafile* *clustering*, similar records in the table are stored together in some way, such as in order by primary key or, in the case of a grocery store, by item type.

1. *What is an index, and how can it improve the performance of a system?*

An index is a small, quickly searchable table that contains values from the table and indicates where in the table those values can be found. System performance is improved with an index because it is no longer necessary to search the entire table for the desired values. The small index table can be quickly searched to reveal exactly where the desired values are stored.

1. *Describe what should be considered when estimating the size of a database.*

The size of the database will be based on the amount of raw data expected, the growth rate of raw data that is expected, and the overhead requirements of the DBMS.

1. *Why is it important to understand the initial and projected size of a database during the design phase?*

Even if you have denormalized your physical data model, clustered records, and created indexes appropriately, the system will perform poorly if the database server cannot handle its volume of data. Therefore, one last way to plan for good performance is to apply *volumetrics*, which means estimating the amount of data that the hardware will need to support. You can incorporate your estimates into the database server hardware specification to make sure that the database hardware is suffi cient for the project’s needs.

1. *What are the key issues in deciding between using perfectly normalized databases and denormalized databases?*

A perfectly normalized database is optimized for storage efficiency, minimizing wasted storage space. This data storage design is not as useful when data must be frequently queried, since the data is spread across many tables that must be joined in processing the query. Access speed will degrade in these circumstances. Therefore, if the data is going to be accessed frequently, it may be valuable to denormalize the design to reduce the number of joins that must be processed in a query.

# Solutions to End of Chapter Exercises

1. *Using the Web or other resources, identify a product that can be classified as an end-user database and a product that can be classified as an enterprise database. How are the products described and marketed? What kinds of applications and users do they support? In what kinds of situations would an organization choose to implement an end-user database over an enterprise database?*

Student answers will vary depending upon the information found on the Internet. Examples of end-user databases include Access and MySQL, examples of enterprise databases include Oracle and Sybase.

1. *Visit a commercial web site (e.g., BestBuy.com, Amazon.com). If files were being used to store the data supporting the application, what types of files would be needed? What data would they contain?*

Master files to store the inventory and the customers. Transaction files to store orders. Lookup files store such things as state names. Audit files might be used to store prices changes to items over time. Archive files store past transactions such as old customers and past orders.

1. *Using the Web, review one of the products listed at the end of this exercise. What are the main features and functions of the software? In what companies has the database management system (DBMS) been implemented, and for what purpose? According to the information that you found, what are three strengths and weaknesses of the product?*

* *Relational DBMS*
* *NoSQL DBMS*
* *Multidimensional DBMS*

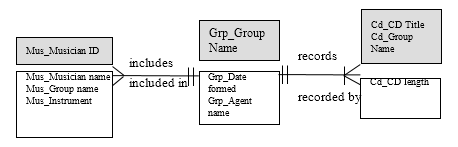
Student answers will vary depending upon the availability of information on the products they find.

1. *You have been given a file that contains fields relating to CD information. Using the steps of normalization, create a logical data model that represents this file in third normal form. The fields include the following:*

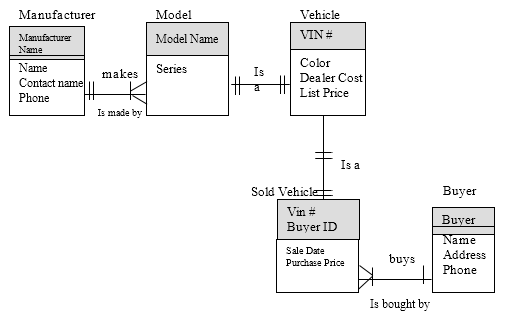
* *Musical group name*
* *Musicians in group*
* *Date group was formed*
* *Group’s agent*
* *CD title 1*
* *CD title 2*
* *CD title 3*
* *CD 1 length*
* *CD 2 length*
* *CD 3 length*

*The assumptions are as follows:*

* *Musicians in group contains a list of the members of the people in the musical group.*
* *Musical groups can have more than one CD, so both group name and CD title are needed to uniquely identify a particular CD.*

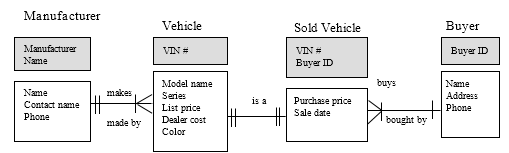


1. *Jim Smith’s dealership sells Fords, Hondas, and Toyotas. The dealership keeps information about each car manufacturer with whom it deals so that employees can get in touch with manufacturers easily. The dealership staff also keeps information about the models of cars that the dealership carries from each manufacturer. They keep such information as list price, the price the dealership paid to obtain the model, and the model name and series (e.g., Honda Civic LX). They also keep information about all sales that they have made. (For instance, they will record the buyer’s name, the car he or she bought, and the amount he or she paid for the car). So that staff can contact the buyers in the future, contact information is also kept (e.g., address, phone number). Create a logical data model. (You may have done this already in Chapter 6). Apply the rules of normalization to the model to check the model for processing efficiency.*

**

1. *Describe how you would denormalize the model that you created in question E. Draw the new physical model on the basis of your suggested changes. How would performance be affected by your suggestions?*

One possibility is to combine the model information into the Vehicle table since this information will probably be accessed together all the time.



1. *Examine the physical data model that you created in question F. Develop a clustering and indexing strategy for this model. Describe how your strategy will improve the performance of the database.*

A cluster that would help performance would be to group the vehicles in the Vehicle table together by model name. Some useful indexes would be to manufacturers, model names, and sales dates.

1. *Investigate the volumetric interface with the computer-aided software engineering (CASE) tool you are using for class. What information do you as an analyst need to input into the tool? How are size estimates calculated? If your CASE tool does not accept volumetric information how can you calculate the size of the database?*

Size of record based on adding the average field size for all fields. An overhead factor from the product vendor. Estimated initial number of records and growth expected.

1. *Calculate the size of the database that you created in question F. Provide size estimates for the initial size of the database as well as for the database in one year’s time. Assume that the dealership sells 10 models of cars from each manufacturer to approximately 20,000 customers a year. The system will be set up initially with one year’s worth of data.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jim Smith Auto Dealership | | | | | | | | | |
| Volumetric Analysis | | | | | | | | | |
| Table Name | Average Record Size | | Overhead Factor | | Total Record Size | Initial Table Size (Records) | | Initial Table Volume | |
| Manufacturer | 50 | | 35% | | 67.50 | 3 | | 203 | |
| Vehicle | 90 | | 35% | | 121.50 | 30 | | 3,645 | |
| Sold Vehicle | 45 | | 35% | | 60.75 | 20,000 | | 1,215,000 | |
| Buyer | 70 | | 35% | | 94.50 | 20,000 | | 1,890,000 | |
|  |  | |  | |  |  | |  | |
|  |  | |  | |  | Total: | | 3,108,848 | |
| After one year: | | |  | |  |  | |  | |
|  | 20,000 sold vehicles |  | | | |  | | 1,215,000 | |
|  | 15,000 new customers | | | | |  | | 1,417,500 | |
|  |  | |  |  | | |  | |  | |
|  |  | |  | Total after one year: | | | | | 5,741,348 | |

1. *How would the following ERD be changed to incorporate the design decision listed next?*

* *The analyst wants to keep track of the user ID of anyone who changes a grade for a course.*

The Grade data store would need to include attributes to capture the User ID and Date changed for grade record modifications. Since there could be multiple changes to a Grade record, a separate data store might be created to hold all the grade changes, and would include the Grade Record ID, User ID, and Date Changed attributes.

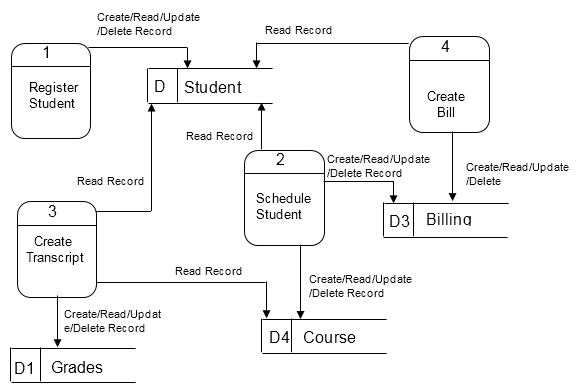
* *A data store is added on the physical DFD so that information regarding the current semester’s courses can be stored temporarily during the add/drop period before the courses become a part of the student’s permanent record.*

Add a data store called Preliminary Transcript with all the attributes of Grade data store. Once the add/drop period is over, the records in Preliminary Transcript are used to create new records in Grade data store and Preliminary Transcript records can be archived.

* *The system would like to archive alumni into a table, once they graduate, so that only active students are stored in the student table.*

At the end of each semester, following recording of all grades, the Student table is searched for all student records that have fulfilled the degree requirements. These records are used to create new records in the Alumni table and then are removed from the Student table.

1. *Draw a physical process model (just the processes and data stores) for the following CRUD matrix.*



# Answers to Textbook Minicases

*1. As shown in the chart below, allow for 6,213,241 characters in the database.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Holiday Travel Vehicles | | | | | |
| Volumetrics Analysis | | | | | |
|  |  |  |  |  |  |
| Table Name | Average Record Size | Overhead Factor | Total Record Size | Initial Table Size (records) | Initial Table Volume |
|  |  |  |  |  |  |
| New Vehicle | 65 | 35% | 88 | 10,000 | 880,000 |
| Trade-in Vehicle | 48 | 35% | 65 | 7,500 | 487,500 |
| Sales Invoice | 76 | 35% | 103 | 16,000 | 1,648,000 |
| Customer | 61 | 35% | 83 | 13,000 | 1,079,000 |
| Salesperson | 34 | 35% | 46 | 100 | 4,600 |
| Installed Option | 16 | 35% | 22 | 25,000 | 550,000 |
| Option | 28 | 35% | 38 | 500 | 19,000 |
|  |  |  |  |  |  |
|  |  |  | Total Initial Volume: | | 4,668,100 |
|  |  |  |  |  |  |
|  |  |  | Growth Rate per Year: | | 10% |
|  |  |  |  |  |  |
|  |  |  | Total Volume in 3 years: | | 6,213,241 |
|  |  |  |  |  |  |

# Supplemental Minicases

1. Jones Legal Investigation Services, Inc., is a growing business that performs legal investigations at the request of attorneys. The company owner, Richard Jones, offers a wide range of investigative services, including audio interviews, video interviews, ground accident scene photos, ground accident scene digital videos, and night vision and electronic surveillance. Using the company’s own plane, aerial accident scene photos and digital videos are also available. When the company was small, Richard could rely on manual filing methods for his case materials, using the computer only for various bookkeeping tasks. However, the company’s primary client, attorney Dan Holm, has such a burgeoning legal practice that the manual filing methods are becoming inadequate. Also, the advances in digital technology make it possible for much of the case material to be stored digitally on computer.

Richard knows that he will need to obtain some expert advice regarding the selection of the appropriate data storage format for a system that will help him manage his case materials. What data storage format do you recommend he consider for this application? Justify your choice. Are there any disadvantages associated with your choice of data storage format?

*Answer: In order to store complex data such as images, videos, and audio as well as text and numeric data, an object-oriented database is recommended. Object-oriented databases have proven to be very capable of handling complex data, and are preferred over any other type of data storage format for this situation. The application proposed here is essentially transaction processing rather than decision making; object-oriented databases are used for this type of application as well. The disadvantage of object-oriented databases is the difficulty of finding qualified, experienced developers to create the databases. These databases tend to be very complex, and there is a steep learning curve associated with using them. A general disadvantage to this system as described here is the huge quantity of storage space that will have to be available to store the graphics, video, and audio files. However, as storage costs continue to plummet, this type of application becomes more plausible.*

1. As you have worked on the system development project team at the Wilcon Company, you have heard the users and project sponsor repeatedly worry about the performance speed of the new order entry system. There has been so much discussion of this issue that you decided to do a little investigation. After a few ‘casual’ conversations around the coffee machine, you’ve learned that a system developed for the Distribution Department several years ago was a major disappointment in the company. Although the system had the features that the users had requested, the system actually ran “as slow as molasses in January” according to one disgusted system user.

Everyone on the team has been concerned about why the users and sponsor were so hung up on system processing speed. Now, you know the source of their anxiety, and you’ve shared what you’ve discovered with your project manager. In response, your manager has asked you to prepare a short memo that discusses, in simple terms, the techniques your team will use to enhance processing performance in the new system. Prepare this memo, explaining the ways in which the project team will try to speed up data access in the new system.

*Answer: As we have developed our database design, we first focused on storing the data as efficiently as possible. We have followed a careful process that produces the most correct data organization. Once that was complete, we then turned our attention to enhancing the speed of access to the system’s data. We have several techniques that we can use to help us with this task.*

*First, we will look at the data organization and the way that you will want to access the data in your reports and queries. When we discover common patterns, such as data items that are always accessed together, we will reorganize our data storage slightly to make it easy to obtain those items as quickly as possible. The technical term for this technique is denormalization. We will also study patterns of record retrieval that will be used. If we find clear patterns or sequences of record access, we will physically structure the records together to speed up access to them. Another technique we make extensive use of is the creation of indexes. An index is a small table that points out the locations of just the records needed to answer a query, for example. We can create many indexes, and they can substantially improve access speed. If necessary, we can place some indexes in the computer’s memory, where they will be executed even faster. A final task we will perform is to carefully estimate the physical storage capacity we will need to accommodate our order entry system to make sure we obtain hardware that is powerful enough for our needs. Even the best system designs will be inadequate if we don’t anticipate the amount of data our system must handle correctly.*

*All of the above techniques are valuable in enhancing the performance of a new database system. You can be assured the team will do everything it can to optimize the performance of the system we are developing.*

# Experiential Exercises

1. Purpose: to understand some practical aspects of data storage design.

Obtain a copy of the physical data model for an implemented information system based on the relational data model. Distribute to the students, and have them identify situations that appear to be deliberate denormalization decisions. Discuss the purpose of the denormalization and how it should affect the system's performance.

1. Purpose: to understand some practical aspects of data storage design.

Invite your campus database administrator to be a class guest (or a database administrator from a local company if possible). Ask the guest to be prepared to discuss and give examples of some of the data access optimization techniques that s/he has employed in 'real' database applications. Also, have the guest be prepared to discuss how s/he estimates the data storage volume requirements for the database applications that s/he oversees.

1. Purpose: to become more familiar with object-relational databases.

Have student work in teams and perform research on the web into database products that encompass both object and relational features. Following the research, have the teams present their findings in a class discussion. Try to identify the common characteristics between products (if any).

1. Purpose: to become more familiar with multidimensional databases.

Have student work in teams and perform research on the web into multidimensional database products. Following the research, have the teams present their findings in a class discussion. Try to identify the common characteristics between products (if any).

1. Purpose: to become more familiar with data warehouses.

Invite a systems analyst or database analyst to class who has participated in the development of a data warehouse. There may be such a project on your campus or at a local company. Have the guest prepare to discuss the process that was used to develop the data model for the data warehouse, the database that was chosen to support the data warehouse, and the process used to build the data warehouse. Students should learn a great deal from the experiences of someone who has actually performed this work in a real setting.