

Databases

Files are used to capture and store data for further processing. With the advent of random retrieval storage devices and the increasing need for immediate access to stored data, the database approach to data administration has become more viable. This section discusses the advantages of databases and presents their more notable features. In addition, this section discusses alternative database structures.

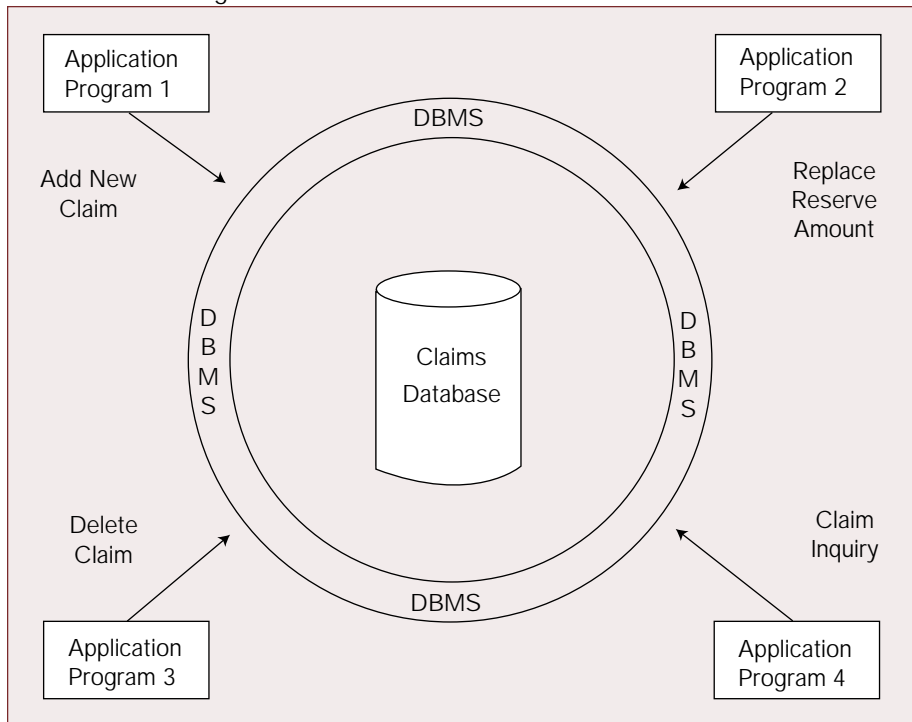
Advantages of Databases

Most organizations purchase a package that can provide a **database management system (DBMS)** suited to their needs. Various database management systems are available, and each has its own advantages. The review and selection of any product should be based on the processing requirements of each company. Some organizations will develop a system in house. The goal of both options should be to provide one common approach to the management of the databases created. Without this, some of the benefits of a database approach will not be realized. A common approach is required because as each database is created and implemented, the application program that accesses the database calls on the services of the DBMS to perform the functions of retrieval, updating, insertion, and deletion of data items on the databases. It is the purpose of the DBMS to coordinate these activities and to provide a consistent approach. An analogy can be made to a standard office filing cabinet. If more than one approach to filing daily reports is used, it would soon become impossible to locate a given file. It would not be possible to share one common file among users. However, a standard approach would make it possible to locate a given file, thus making possible the establishment of a single file that could be used by many people, instead of establishing many files, which can be time-consuming and can introduce the possibility of disseminating different information to the many file users. The database approach depends on the same concept. As shown in Exhibit 4-7, a database management system controls access to a central claims database and allows it to be used by four separate application programs.

Collection of Interrelated Data

One of the most important considerations in the design and implementation of databases is the concept of **subject databases**, or the **enterprise database model**. This approach to database design has been shown to be the most

Exhibit 4-7
DBMS Controlling Access to Databases



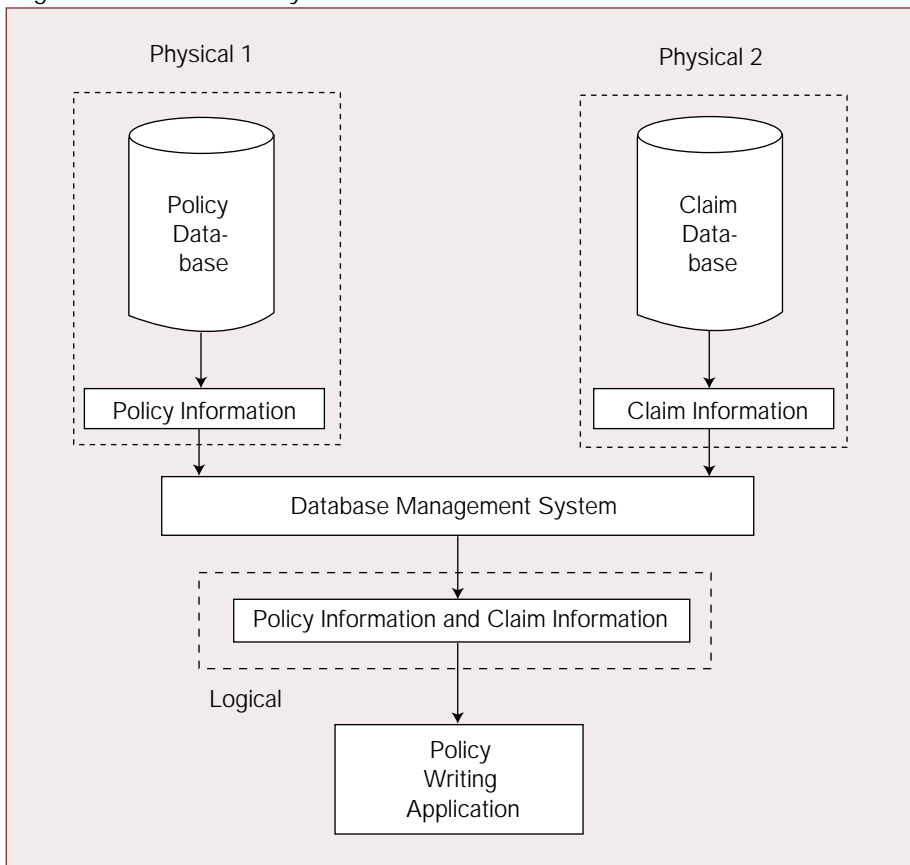
Property of AICPCU/IIA

effective method of meeting current needs while providing the capability to support future requirements. This approach is based on the assumption that data arranged to fit the basic business relations will be much more capable of being used without rearrangement for a much longer period of time. The less attractive alternative approach is to design databases according to the requirements of a given application. An **application database** leads to problems as companies reorganize and new functions have to be added to a system. By developing subject databases, such as claims information, policy information, and billing information, and trying to keep each class of data on its respective database, many more applications can be developed to use this data.

One implication of subject databases is that a given application may have to process multiple databases concurrently. Multiple databases being processed by one application, because of the segregation imposed by subject databases, may at first appear to be an inconvenience to an application development team. However, most database management systems allow the combination of more than one physical database into a logical database. This enables the application development group to view and process what appears to be one

database when, in fact, it would be multiple physical files. For example, a policywriting application may require loss information as well as policy information in order to properly rate a policy. In this case, information regarding losses could be contained on a claims database, and information specifically related to the policies could be contained on a policy database. The application programmers, however, could view these separate databases as if they were one database. Exhibit 4-8 contrasts the logical and physical views of such a database.

Exhibit 4-8
Logical View Versus Physical Database



As shown in Exhibit 4-9, in traditional systems that were developed with the use of files and not databases, a similar application would cause data to be passed from a claims processing application to the policywriting system. As a result, the data would be stored in multiple files. This type of processing not only duplicates data; it also creates inconsistent data. In this example, the transactions sent from the claims system to the policywriting system could be