INTRODUCTION

Over the past two decades, data has become a strategic asset for most organizations. Databases are used to store, manipulate, and retrieve data in nearly every type of organization, including business, health care, education, government, and libraries. Database technology is routinely used by individuals on personal computers and by employees using enterprise-wide distributed applications. Databases are also accessed by customers and other remote users through diverse technologies, such as Web browsers, smartphones, and intelligent living and office environments. Most Web-based applications depend on a database foundation.

BASIC CONCEPTS AND DEFINITIONS

We define a **database** as an organized collection of logically related data.

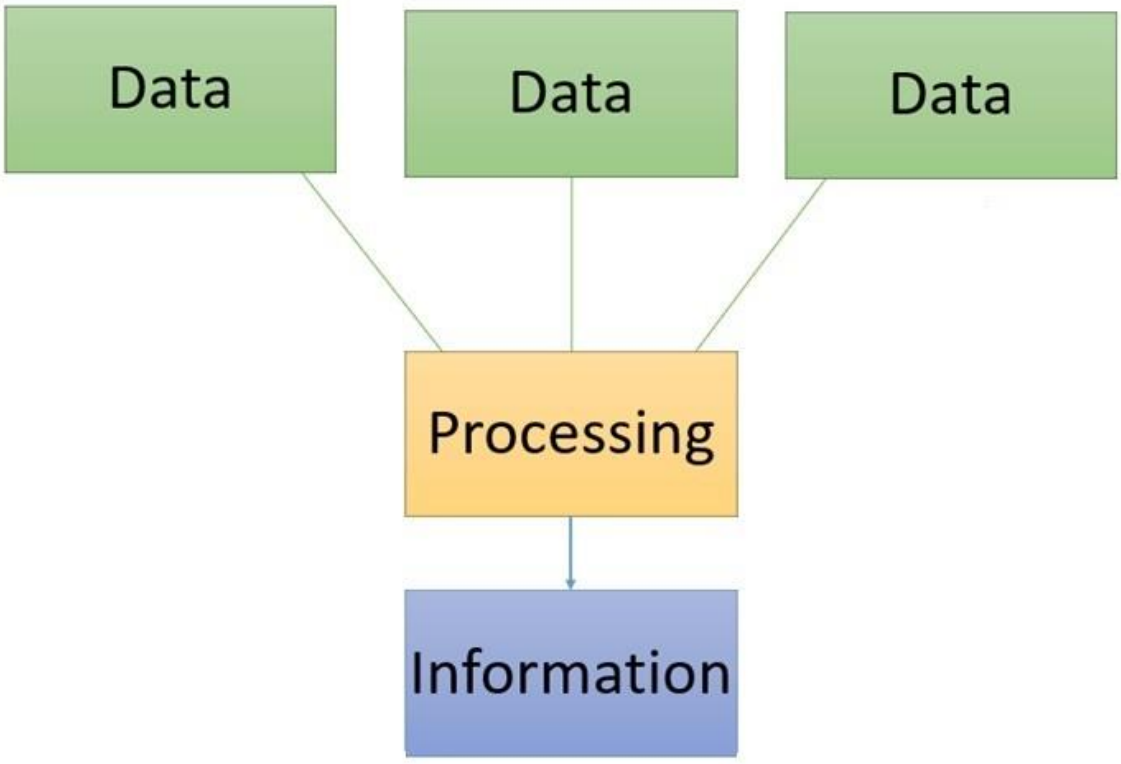
Data

Historically, the term *data* referred to facts concerning objects and events that could be recorded and stored on computer media. For example, in a salesperson's database, the data would include facts such as customer name, address, and telephone number. This type of data is called *structured* data. The most important structured data types are numeric, character, and dates. Structured data are stored in tabular form. The traditional definition of data now needs to be expanded to reflect a new reality: Databases today are used to store objects such as documents, e-mails, tweets, Facebook posts, GPS information, maps, photographic images, sound, and video segments in addition to structured data. For example, the salesperson's database might include a photo image of the customer contact. It might also include a sound recording or video clip about the most recent product. This type of data is referred to as *unstructured* data, or

as multimedia data. Today structured and unstructured data are often combined in the same database to create a true multimedia environment. An expanded definition of **data** that includes structured and unstructured types is "a stored representation of objects and events that have meaning and importance in the user's environment."

Data versus Information

The terms *data* and *information* are closely related and in fact are often used interchangeably. However, it is useful to distinguish between data and information. We define **information** as data that have been processed in such a way that the knowledge of the person who uses the data is increased.



Metadata

Data become useful only when placed in some context. The primary mechanism for providing context for data is metadata. **Metadata** are data that describe the properties or characteristics of end-user data and the context of that data. Some of the properties that are typically described include data names, definitions, length (or size), and allowable values. Metadata describing data context include the source of the data, where the data are stored, ownership (or stewardship), and usage. Although it may seem circular, many people think of metadata as "data about data."

Notice the distinction between data and metadata. Metadata are once removed from data. That is, metadata describe the properties of data but are separate from that data. Metadata enable database designers and users to understand what data exist, what the data mean, and how to distinguish between data items that at first glance look similar. Managing metadata is at least as crucial as managing the associated data because data without clear meaning can be confusing, misinterpreted, or erroneous. Typically, much of the metadata are stored as part of the database and may be retrieved using the same approaches that are used to retrieve data or information.

Data can be stored in files (think Excel sheets) or in databases. In the following sections, we examine the progression from file processing systems to databases and the advantages and disadvantages of each.

TRADITIONAL FILE PROCESSING SYSTEMS

When computer-based data processing was first available, there were no databases. To be useful for business applications, computers had to store, manipulate, and retrieve large files of data. Computer file processing systems were developed for this purpose. Although these systems have evolved over time, their basic structure and purpose have changed little over several decades. As business applications became more complex, it became evident that traditional file processing systems had a number of shortcomings and limitations. As a result, these systems have been replaced by database processing systems in most business applications today. Nevertheless, you should have at least some familiarity with file processing systems since understanding the problems and limitations inherent in file processing systems can help you avoid these same problems when designing database systems. It should be noted that Excel files, in general, fall into the same category as file systems and suffer from the same drawbacks listed below.

DISADVANTAGES OF FILE PROCESSING SYSTEMS

Several disadvantages associated with conventional file processing systems are described briefly next. It is important to understand these issues because if we don’t follow the correct database management practices, some of these disadvantages can also become issues for databases as well.

1. **Program-Data Dependence**

   File descriptions are stored within each database application program that accesses a given file. Because the program contains a detailed file description for these files, any change to a file structure requires changes to the file descriptions for all programs that access the file. It is often

difficult even to locate all programs affected by such changes. Worse, errors are often introduced when making such changes.

2. **Duplication of Data**

Because applications are often developed independently in file processing systems, unplanned duplicate data files are the rule rather than the exception. This duplication is wasteful because it requires additional storage space and increased effort to keep all files up to date. Data formats may be inconsistent or data values may not agree (or both). Reliable metadata are very difficult to establish in file processing systems.

3. **Limited Data Sharing**

With the traditional file processing approach, each application has its own private files, and users have little opportunity to share data outside their own applications. Managers often find that a requested report requires a major programming effort because data must be drawn from several incompatible files in separate systems. When different organizational units own these different files, additional management barriers must be overcome.

4. **Lengthy Development Times**

With traditional file processing systems, each new application requires that the developer essentially start from scratch by designing new file formats and descriptions and then writing the file access logic for each new program. The lengthy development times required are inconsistent with today's fast paced business environment, in which time to market (or time to production for an information system) is a key business success factor.

5. **Excessive Program Maintenance**

The preceding factors all combined to create a heavy program maintenance load in organizations that relied on traditional file processing systems. In fact, as much as 80 percent of the total information system's development budget might be devoted to program maintenance in such organizations. This in turn means that resources (time, people, and money) are not being spent on developing new applications.

It is important to note that many of the disadvantages of file processing we have mentioned can also be limitations of databases if an organization does not properly apply the database approach. For example, if an organization develops many separately managed databases (say, one for each division or business function) with little or no coordination of the metadata, then uncontrolled data duplication, limited data sharing, lengthy development time, and excessive program maintenance can occur. Thus, the database approach, which is explained next, is as much a way to manage organizational data as it is a set of technologies for defining, creating, maintaining, and using these data.