Chapter 4

DATABASE DEVELOPMENT LIFE CYCLE

CHAPTER OBJECTIVES

- Understand the need for a careful and systematic approach to database development
- Examine the steps and tasks in the database development life cycle (DDLC)
- Learn why proper planning is absolutely essential for database development
- Study how requirements are defined and how the design phase is completed
- Observe and appreciate the tasks necessary to complete the implementation and for ongoing maintenance of the database system

At this point, you greatly appreciate the pivotal role of the database system in an organization. The success of modern organizations rests on the effectiveness of their database systems. The database system in an organization provides information critical for achieving the goals of the organization, for fulfilling its core business, and for driving the primary processes. The importance of the organization's database system cannot be overstated.

If the database system of an organization is of paramount importance, then its design and development must be substantially significant as well. If so, how should an organization go about establishing its database system? Obviously, with a lot of care and attention. We will discuss the activities and tasks required to create a sound database system for an organization. We will begin with an overview of the required steps; next, we will walk through each of the steps in sufficient detail. When we have completed the discussion, you will have gained a broader understanding of the design and development process.

Database Design and Development: An Essential Guide for IT Professionals by Paulraj Ponniah
ISBN 0-471-21877-4

MAJOR DEVELOPMENT STEPS

For a moment, think of an organization running its business with file-oriented systems. All the data that support the business reside on sequential or flat files. There is no flexibility, no data sharing. Each user group has its own set of files; you note data proliferation to a great extent. You find disparate islands of data—all of which are meant to support one core business of the organization. The same elements of data may be found in different files, and these duplicate versions of the same data elements could result in gross data inconsistency.

Now imagine the same organization running its business with a database system. This environment is quite the opposite of one with file-oriented systems. First, you observe data integration with completely reduced data duplication. Removal of data redundancies produces a major improvement and projects a uniform image of the company to the outside world. Every document sent to customers and business partners shows the same values for any data element.

Information sharing resulting from a database system is a major change in the organization. Users are no longer confined to their own individual sets of files. Each group of users does not try to protect its data turf. Data in the database system are everyone's data; the integrated whole is for the benefit of the entire organization; the data serve the core business of the organization as a single entity. The transition of an organization from data proliferation to information sharing constitutes a major step.

So how does the changeover from individual data files to an integrated database system occur in an organization? How does the transition from an environment of isolated data islands to one of information sharing happen? With a lot of planning, of course. Companies need to plan and think through all the aspects of the change. The transition requires a systematic and coordinated effort. The development of a database system calls for a proven approach with distinct and purposeful steps. Businesses adopt a life cycle approach to database development. In a life cycle approach, you pay attention to each individual step and make the steps proceed in a standard, productive manner. The approach deals with all the necessary steps in the life of a database system from beginning to ongoing phases.

Starting the Process

Let us go back to when data systems and applications were developed in a file-oriented environment. Consider developing applications for the order processing department in an organization. You study the various functions performed for completing the processing of orders. You gather the data required to complete each task of these functions. Then you put all the required data into one or more sequential files, and these files make up the data system for the department. Here you use a function-oriented approach to developing the data system. In a function-oriented approach, the focus is on the functions. If the functions change or if you need to add more functions, the composition of the data system changes too, sometimes quite extensively.

However, it is different in a database system. This type of data system is designed to cater to current information needs and be able to extend easily to accommodate changes in the requirements and to meet additional ones. You do not focus on the

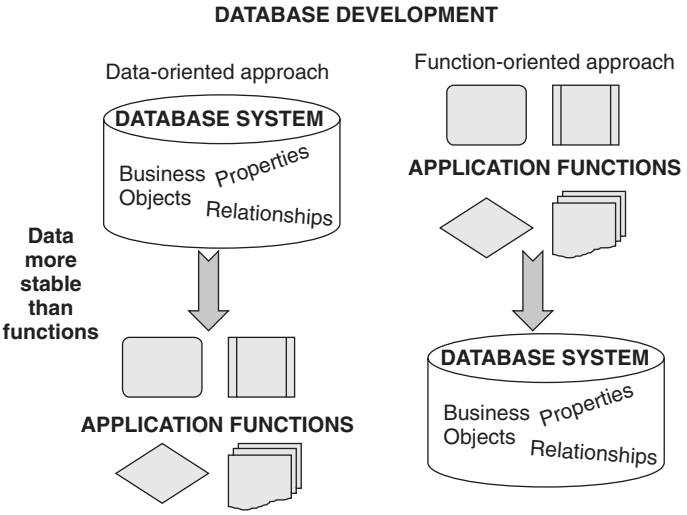


Figure 4-1 Data-oriented versus function-oriented approach.

applications first but look at the overall data requirements of running the business. What are the business objects? What are the relationships among the objects? You store all relevant data about the business objects and then use the data to run the applications. This approach is different—it is data-oriented as opposed to being function-oriented.

Adopt a Data-Oriented Approach

Figure 4-1 illustrates this essential difference. At the very outset, we need to appreciate the significance of the difference in the development of the new data system. The success of database development rests on the adoption of a data-oriented approach.

Establish a Framework for Development

Because of the tremendous importance of the database system in an organization, you must create and work with a structured framework for its development. Figure 4-2 shows a framework for database development. You must adapt and use such a framework.

Begin Development Project with an Initiation Report

A carefully prepared project initiation report defines the scope of the database development, the methodology, the basic data content, and the overall project schedule. The report sets the direction and provides the development parameters.

The following are standard contents of an initiation report for a database development project:

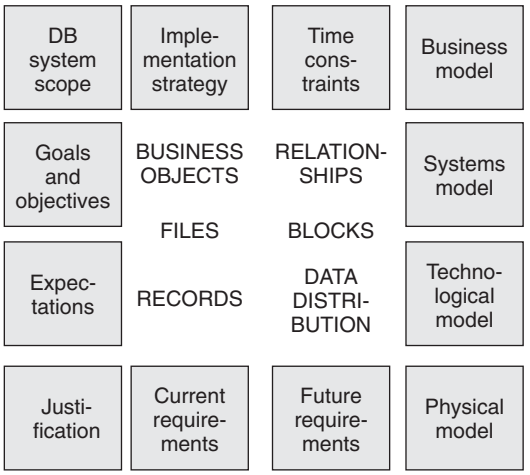


Figure 4-2 Framework for database development.

- Scope of the proposed database system
- Goals and objectives
- Values and expectations
- Justification
- Key business objects for the organization
- Core business and primary processes
- Supporting data elements
- Implementation strategy
- Tentative schedule
- Project authorization

Design and Development

How does a manufacturing company produce a product? The company initiates the process, performs the design, and gets the blueprints ready; then it develops and transforms the design into a finished product. Although it appears simplistic, design and development make up the processes for the manufacture of a product. This metaphor may be extended to the establishment of a database system. After the initiation phase, design of the database system takes place; then the design forms the basis for the development of the system. Figure 4-3 provides an overview of database design and development.

To design the database system, you have to gather the information requirements to support the various processes that make the core business of the organization happen. What are the requirements? What should the data residing in the organization's database be about? Remember that this is a data-oriented approach and the primary focus is on the data requirements. Ascertain the business entities or things of interest to the organization. Is your organization a banking institution? If so, some of its business entities are customer, checking account, savings account, and

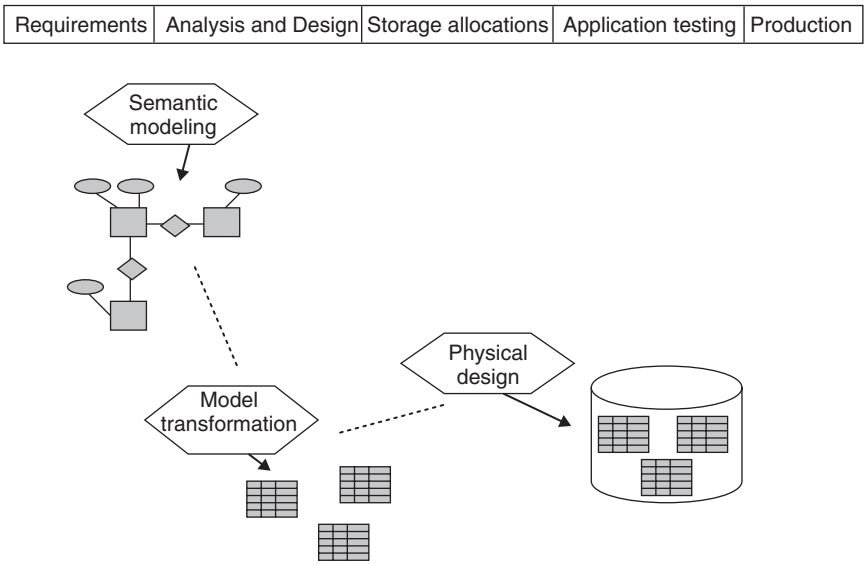


Figure 4-3 Database design and development.

loan account. Is your organization a medical center? Then a few of its entities are patient, physician, diagnosis, and treatment procedures. Whatever the nature of your organization's core business, you list the business entities and determine the information desired about each of these entities. In the case of the banking institution, a customer is related to one or more checking accounts. While considering the business entities, you note such relationships between sets of entities. These relationships must be ascertained to understand how information will be retrieved from the database system and used.

Let us say that you have determined the entities and the relationships. What do you do with them? How do you represent them in the database, and how do you store information about them? How do you design your database system? Design is broken down into two phases based on what actually happens in the design process itself.

Logical Design. True, data is stored in the database as files and records. But what about the structure of the data? What are the pieces of data within each structure? How do the various elements of the structure relate to one another? What are the connections? In the first design phase, you formulate these data structures, their arrangement, and their relationships. You come up with the conceptual view of the database. You use boxes and lines to represent this conceptual view as a logical design diagram. This design is not at the level of how data is stored as files, records, and fields. It is at a higher level that can be understood easily by the users—a level that can be used as a means for communication among the database practitioners and between the database practitioners and the user groups.

Physical Design. Once the logical design is firmed up, it has to be transformed into the design of the physical files and records because that is how data can be in

physical storage. Each component of the logical design is mapped into one or more physical design components. Physical design deals with data storage, data retrieval, data updates, and methods for improving data access performance. Where you want to keep the data, how you intend to lay out the data on the storage media, how you establish the relationships at the storage level—these make up the components of the physical design.

Development and Implementation

When the logical design and physical design phases are completed, you are ready to develop and implement the database. The database administrator has the responsibility of completing the development and implementing the database. He or she defines the data structures, relationships, business rule constraints, storage areas, and performance improvement techniques in the data dictionary.

The database is then populated with data. If the transition in an organization is from a file-oriented system to a database system, then data from the previous files are extracted and converted. Special database utility programs and specially written conversion programs are used to load data in the new database.

Steps and Tasks

The life cycle approach comprises systematic and well-defined steps or phases to complete the design and development of a database system. Each step comprises specific major activities; each activity contains individual tasks. Although the project progresses from step to step, it is not necessary to complete all the activities of one step to proceed to the next. Parts of the steps may be performed in parallel. Sometimes it becomes necessary to repeat and refine some of the steps in an iterative manner.

Figure 4-4 shows the major steps of the database development life cycle (DDLC). Note the sequence of the steps from the bottom to the top. Note how the figure illustrates that the steps may be performed in parallel. While some aspects of requirements definition are still to be completed, the design step may commence. Also, observe the indication of the iterative nature of design and development. When you complete a portion of the design, you may want to go back to the requirements definition phase to review and refine a few aspects of the data requirements.

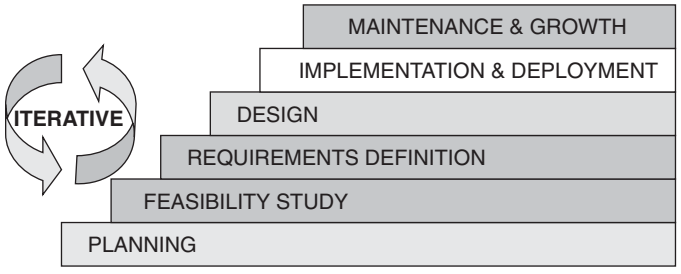


Figure 4-4 DDLC: major steps or phases.

In the rest of this chapter, we will be studying each step or phase of DDLC in sufficient detail. However, at this point, let us highlight the objective for each step.

Planning. Review the organization's long-term plan; plan specifically for the database system.

Feasibility study. Study the state of readiness; establish the costs and benefits.

Requirements definition. Define the business objects or entities; determine data requirements.

Design. Complete logical design; transform it into physical design.

Implementation and deployment. Populate database; train users; get database ready for applications.

Maintenance and growth. Perform ongoing maintenance; plan for growth of the database system.

Roles and Responsibilities

Who are the people responsible for database design and development? Who are the people doing the design and implementation phases? Who gathers the data requirements? What are their roles? What types of specialized skills are needed in the major design and development effort? To a limited extent, we have already looked at the roles of database practitioners and database users. Now, we will define the roles more precisely. First, a few points about the specialized skills that are necessary in a database project.

Specialized Skills For a company, the launching of a database system for the first time is a major undertaking. The transition is from a file-oriented data system to a completely different approach to storing, accessing, and using data. The company needs to modify its very attitude and practices about data usage. The people responsible for designing and developing the database system need a whole new set of skills for proceeding through the DDLC phases. Database design and development demands different techniques and poses greater challenges.

Here is an indication of specialized skills needed for the various phases:

Planning. Ability to interpret the organization's long-term plan and apply it to the database plan.

Feasibility study. Skills to assess the resource requirements, calculate costs, and determine tangible and intangible benefits.

Requirements definition. Analytical skills to identify business objects and related data, knowledge of interviewing techniques, and ability to collect and present requirements in a complete and correct way.

Design. Knowledge of data modeling; training in CASE tools, in-depth understanding of selected DBMS, and knowledge of system software, data storage methods, and media.

Implementation and deployment. Experience in data conversion and storage space allocation; knowledge of remote and on-site deployment techniques.

Maintenance and growth. Knowledge and experience in data security, data integrity, and data recovery.

User Roles The participation of users in the database design and development project must extend throughout the project. In each phase, users have important roles to fulfill. No database project can succeed without the intimate participation of users. Database practitioners and users must work in close cooperation in every phase. Although database practitioners take primary responsibilities in the design and implementation phases, users have definite tasks in these phases as well as the other phases.

Figure 4-5 indicates how and where users must be made to participate in a database project. Carefully note the roles and responsibilities in each phase. We must emphasize that database design and development is not the sole responsibility of

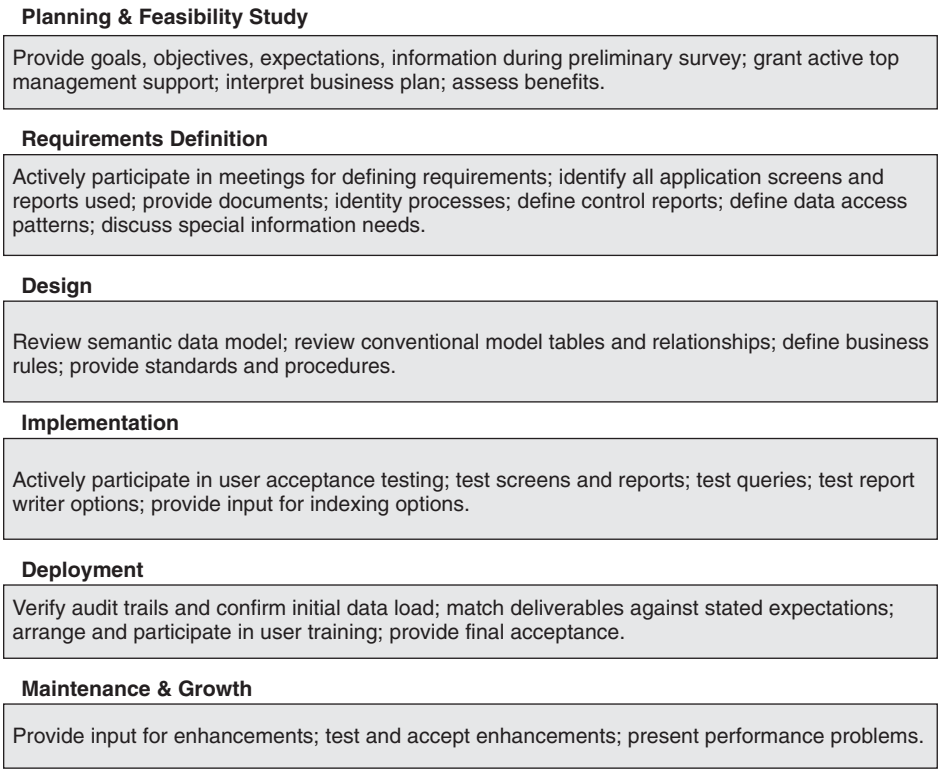


Figure 4-5 DDLC: user participation.

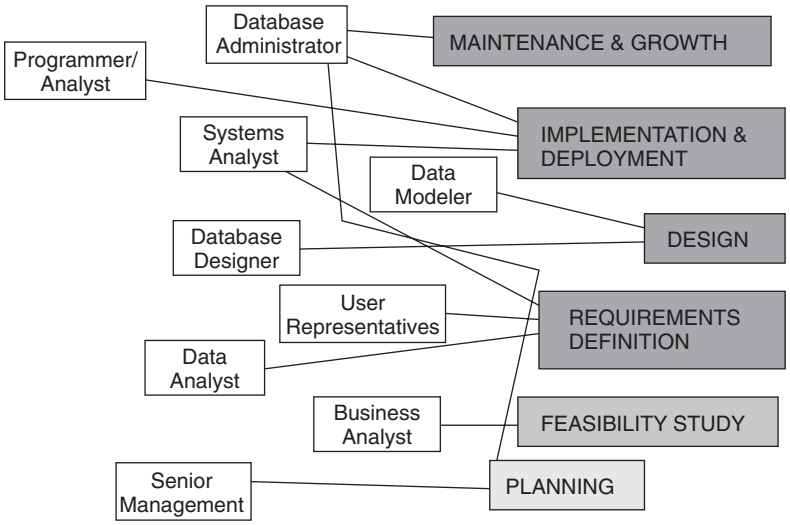


Figure 4-6 DDLC phases: participation of users and practitioners.

the information technology department of an organization. User participation, involvement, and acceptance of responsibility will result in a successful database system.

Roles of Database Practitioners Now let us bring out the cast of characters. Who are the database practitioners and what are their roles? What are the phases in which they perform the tasks of design and development? These questions are best answered by means of two figures.

First, Figure 4-6 relates database practitioners and users to the phases where they are most active in fulfilling assigned tasks. Note how some practitioners participate in more than one phase.

Next, Figure 4-7 separates out design and development into data and process areas. In each of the two areas, practitioners and users assume specific roles. Note the types of activities in the two areas and some of the techniques adopted for performing the activities.

Management and Control

We have discussed information as a key resource for modern organizations. You have understood that the success and sometimes the very survival of many organizations rest on the information obtained from their database systems. Almost all of today's organizations depend on their database systems to run their day-to-day business. Because the database system is an indispensable resource, it has to be managed with utmost care and attention. Organizations must manage and control how their database systems provide storage of information, how they are accessed, and who may access data.

In the effort to manage and control this key resource, organizations face a few new challenges. These challenges are new and different because the management

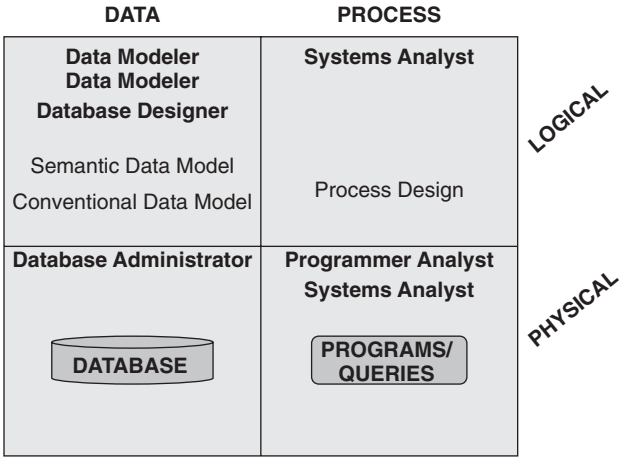


Figure 4-7 Data and process: design and development roles.

of earlier file-oriented systems was much simpler. Here are some of the major challenges:

- The database is more open to everyone in the organization than the separate files in the earlier data systems. Openness and information sharing necessitate better control of security. The database system must be made secure by preventing unauthorized access by insiders and outsiders. Only authorized users must be allowed to use relevant portions of the database system.
- In a database environment, many users may be attempting to access and update the same data elements simultaneously. This is likely to cause erroneous updates and compromised data integrity. Preserving data integrity in a database system is a big challenge.
- If the database system is crucial for an organization, then it should be available to users at all times when they need information. The database system must provide uninterrupted service with no unscheduled downtime. Whenever hardware or software malfunctions occur, the database must be restored to a stable state and become available for use quickly.
- In a dynamic business environment, changes happen all the time. The database system must be amenable to such changes. If business conditions warrant newer elements of data, the database must accommodate changes and additions to the data structures. If the number of database transactions increases and more users begin to access the database, the system must be able to scale up.

Who is responsible for management and control of the database system? Although user representatives and database practitioners share the responsibility to some extent, the database administrator bears the primary responsibility. In smaller organizations, one person may assume the role of database administration. In larger organizations, an entire group of professionals perform the database administration function. Here is a brief list of the management and control functions carried out by the administrators:

- Grant or revoke database access privileges to users.
- Arrange periodic backups of database content.
- Manage storage allocation.
- Install software upgrades.
- Collect data access statistics and monitor usage.
- Fine-tune the database as necessary.
- Oversee data integrity provisions.
- Recover database from hardware and software malfunctions.

Chapter 14 extensively discusses the administration functions in a database environment. You will learn a great deal more about the functions of the data administrator and the database administrator in that chapter.

PLANNING FOR THE DB SYSTEM

Before we begin a database project, we have to assess how the management and users understand data systems in the context of the organization. What do they think about the utility of data systems to support the core business and the business processes? If they are from a file-oriented environment, their views are likely to be narrow and limited. The planning process broadens their views about data systems and moves them forward to appreciate the benefits of information sharing. Planning involves senior management and therefore provides a means for them to express their support and sponsorship.

Planning sets the direction and scope for the database project. The project will be a new type of project for those organizations launching a database for the very first time. Even for those companies that are expanding their database systems or migrating from one commercial database system to another, the project may offer new challenges. The planning process identifies the challenges; it also brings out the opportunities afforded by database systems.

We can think of the planning process as containing two distinct subphases. In one subphase, you study and interpret the impact of the organization's long-term plan on the proposed database system. If the company is planning for expansion into three new territories within the next two years, what types of data must be included in the database? The other subphase of planning relates to the actual planning for the database system based on the current requirements. The two subphases are important to complete the planning because you must combine current and future requirements in the design.

The planning process, naturally, is the first phase or step in DDLC. Although some limited portion of the next phase may be started while planning is being completed, generally organizations tend to complete the planning phase before proceeding to the other phases. Planning sets the tone and direction of the entire project, and this is important for the conduct of the other phases. How long should planning last? It depends on the situation in each organization. If your organization is one for which a database project is completely new, then the people doing the planning must carefully review all relevant issues and take the time to complete the

planning process. Nevertheless, as a general rule, the planning phase should not last longer than four to six months.

The planning phase ends with a clear report that can be used and referred to in later phases. This planning document serves as a guide and reference throughout the project. It articulates the management's view and understanding of a database system; it states the expectations; it communicates the opportunities. The planning document, as soon as it is completed, confirms the management's commitment to the database project.

Scope of Overall Planning

Planning a database system translates into planning for the current and future information requirements of the users in several organizational units performing different business functions and residing at various geographic locations. They need information about business objects that are relevant to the organization's goals and objectives. The organization's long-term business plan provides input for the planning phase. Again, the business plan is grounded in the company's core business. The blend of all these planning objects is the overall database plan for the project.

Figure 4-8 illustrates the overall planning process. Note how the organization's business plan provides input for the planning process. Observe the elements of the core business that go into the planning. Note how the business plan and the factors of the core business lead into the user requirements and how all of these are tied together in the database plan.

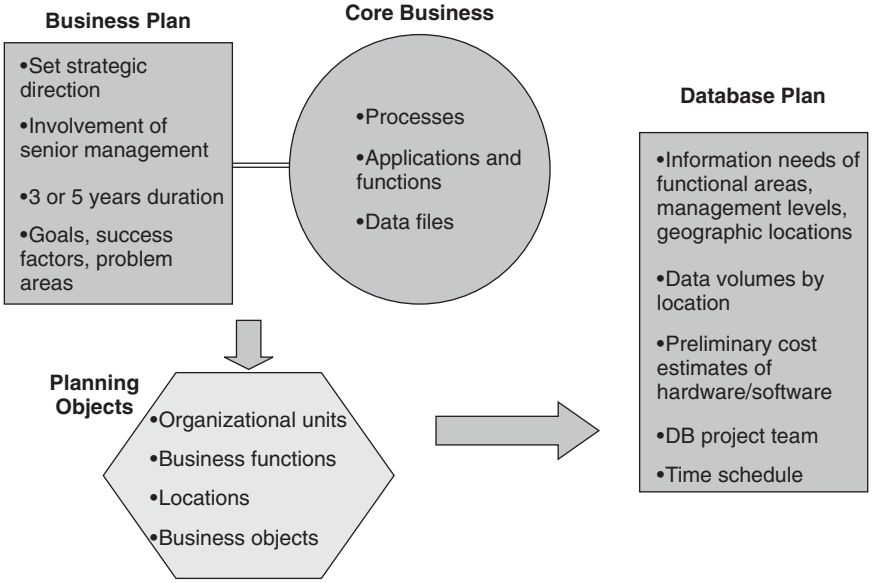


Figure 4-8 Overall planning.

Who Does the Planning?

The senior executives of an organization think through the future opportunities and challenges. Every year most organizations come up with a long-term plan for 3–5 years; some even prepare 10-year plans. The senior management takes the primary responsibility for producing the long-term plans. The CEO of the organization is intimately involved and usually directs the planning. Some institutions have separate planning departments to collect data for the planning process and coordinate the overall planning effort.

What has the long-term business plan to do with the database project? The proposed database for the organization is not meant to be static. Once implemented, the database system will have to grow and accommodate requirements relating to the foreseeable future. It should be able to conform to the direction and goals set forth in the long-term business plan. It must be able to anticipate the data requirements for the next three to five years.

We have already noted that one subphase of project planning deals with interpreting the organization's long-term plan and applying it to the database project. The other subphase is the consolidation of all requirements and preparation of the database plan. Generally, organizations carry out these two subphases by assigning responsibilities as noted below.

- Senior Managers of the IT Department review the long-term plan, estimate the data that would be required to support the long-term goals and objectives, and highlight the business areas that would need the support from the proposed database system. The CIO and CTO are likely to be part of the group of the organization's senior executives who created the long-term plan. Therefore, these executives contribute greatly in the interpretation of the long-term plan for the purpose of relating it to the database project.
- Usually, companies form a special team within IT to prepare the database plan. This team consists of the heads of data administration, database administration, database design, and application development. Some others with specialized skills in relevant hardware and software may also be included in the planning team.

Impact of the Business Plan

We have emphasized the significance of applying the long-term business plan to the database project. How exactly does the business plan impact the proposed database system? First of all, what do long-term business plans contain? We will briefly review the normal contents of a long-term business plan and then see how each element in the plan can affect the planning for the database.

Among other components, here are the major contents of a long-term business plan:

- Scope and duration of the plan
- General goals
- Specific objectives

- Special opportunities
- Challenges, internal and external
- People responsible for executing parts of the plan

Now let us consider each of these major components and see how it applies to planning the database project.

Scope and duration. These set the boundaries in terms of geographic and logical extents as well as in terms of time. You will know who will need data in the future, what types of data, and when.

General goals. These are broad statements of what the organization wishes to do in the future. For example, an organization may intend to strengthen certain product lines and drop a few lines in the future. Or an organization may express a general theme of cost cuts. It may emphasize an overall improvement in product shipments. It may underscore progress in customer service. General goals do not come with definable targets. What can we do about general goals in the database systems? Use general goals to identify areas of special emphasis and ensure that the database covers these areas. If improvement of customer service is emphasized as a general goal, make sure that the database would contain data elements to support this business function.

Specific objectives. These take the form of directives with set targets. For example, a specific objective of an organization may be increasing the market share by 5 percent within three years or a 6 percent increase in overall sales within five years. Specific objectives tend to be aggressive and definite. Defined action items also accompany specific objectives. For example, the objective of increased market share may be accomplished through expansions in the western and central regions. Look for specific objectives in the organization's long-term business plan. These are not like general goals. Those formulating the database plan must take the specific objectives seriously and ensure that the proposed database system supports these specific objectives completely. Pay special attention to the action items. If an expansion in the market in the central region is set forth as an action item, then plan to have the database include additional stores and offices in that region.

Special opportunities. The long-term business plan may detect certain trends and spot opportunities that are likely to result from these trends. For example, if your organization is a financial institution, the lowering of interest rates is expected to increase the number of loans that can be offered. Assess the expression of these opportunities in the business plan and then plan to accommodate the data requirements in the database system. If the number of loans is expected to increase, plan to include all data elements that will make loan processing easier and faster.

Challenges. The proposed database system can help the organization in overcoming some of the challenges the organization faces or is likely to confront in the near

future. For a pharmaceutical company, conforming to the requirements of the Food and Drug Administration (FDA) and providing mountains of data for product approvals may pose a great challenge. Your proposed database system can be robust and flexible enough to store, sort out, and provide data for FDA approvals.

People. These are the people in the organization responsible for carrying out the directives in the long-term business plan. When you interpret the business plan, you might need additional information as to the proposed data requirements. You then go to the people who are charged with the execution of parts of the business plan.

The Database Plan

The database plan covers what must be done in the remaining phases of DDLC. It sets the tone for the project; spells out the key activities necessary for the success of the project, and provides a planning document to guide the rest of the project. Study the following highlights of the activities for preparing the database plan and for suggestions on its content.

- Classify your users by responsibility levels, business functions, and geographic regions. Include the overall, very general nature of the data requirements for each user group.
- Adopting a data-oriented approach, consider the business objects and intrinsic data about them. These are the data that will be stored in the proposed database system.
- Make preliminary estimates of data volumes. These will help you determine storage capacities and access paths.
- On the basis of broad assumptions, prepare very general estimates of hardware and system software requirements.
- Take input about future data requirements from the long-term business plan.
- Gather data requirements on current functions, applications, and files.

The issuance of a database plan report marks the conclusion of the planning phase. This planning document gathers the discussions and decisions of the phase; it sets the tone and momentum for the remaining phases. Here is a typical outline for a database plan report:

- Introduction
- Scope
- Specific Objectives
- Key Issues and Assumptions
- Expectations
- Overview of Data Requirements
- General Implementation Strategy
- Tentative Schedule

Critical Planning Issues

While preparing the database plan, the staff preparing the plan must be continuously watchful for special challenges and critical issues. Database design and development is not a common or easy project. Several new challenges will arise, and sufficient awareness and planning for the contingencies are necessary. Let us briefly go over a few of the major issues.

Data Ownership If your organization is making the transition from a file-oriented data system to a database system, then you will be faced with resolution of data ownership conflicts. Even when a database project consists of conversion from one type of database system to another, some data ownership conflicts remain. Who owns the data in the database? Which user groups are responsible? You know that the database is for information sharing. No one group has exclusive ownership. Then who is responsible for data quality? For example, many user groups would be adding and updating customer data. Which user group owns customer data? Your planners must recognize potential data ownership conflicts and review general principles applicable to your organization.

Specialized Skills As already mentioned, implementation and maintenance of database systems require specialized skills. You need people trained in data administration, database administration, data modeling, database design, database management systems, and programming in a database environment. The database plan must consider issues relating to recruitment and training of personnel throughout the entire DDLC.

Project Vulnerability For most organizations, when they undertake a database project they are breaking into new ground. They have not faced many of the issues and challenges posed by a database project. Possibilities exist for failures, delays, and cost overruns in such a novel project. Planners ought to recognize the possibilities for failures and suggest preventive measures and remedial action. For example, if project failure is possible because of the spread of user groups over many locations, planning must specifically address how to bring about cooperation and collaboration.

New Overhead Costs Be prepared for additional overhead costs. Some of these costs may be for additional hardware infrastructure; some may be for enhancing the operating system software; a large chunk of additional overhead costs may be for people with new specialized skills. Plan for such new overhead costs.

Issues for Large Projects Large projects pose particular challenges. A large database project covers numerous user groups, extremely big data volumes, a huge number of anticipated database transactions, and data about several business objects. Many database professionals and user representatives participate in a large project. When the size and scope of a project is huge, communication among the designers, developers, administrators, and user representatives may be a source of potential problems. Coordination and control could be difficult. Staff retention for the entire duration of the project is another area for concern. Recruitment and

retraining may delay the project. Planners ought to be aware of the special problems and challenges related to large projects and indicate methods to address and contain the problems. They may also suggest how the project may be divided into manageable parts in each phase of DDLC.

FEASIBILITY STUDY

The database project has been initiated, the database plan has been completed, and the plan report has been issued. Are you then ready to jump into the requirements and design phases? Should you simply proceed into each phase? Are there any more essential activities before going ahead full steam?

The project initiation denotes the formal beginning of the project—an announcement of the intention for a database system. The planning phase defines the scope of such a database system and lays down the planning issues. However, your organization does not yet know how ready it is for a database system.

Before proceeding with the project, you need to assess this readiness. No organization is completely ready for a new database system—the hardware and software platform may need revisions and enhancements, sometimes very extensive ones. People would need to be trained in specialized skills. Therefore, between the planning phase and the other design and development phases, there must be definitive activities to assess readiness. This set of intervening activities makes up the feasibility study phase of DDLC.

Purpose and Extent

What happens during the feasibility phase of a database project? Usually, business analysts with assistance from database administrators conduct the feasibility study phase. Sometimes, the responsibility for the phase is assigned to the project manager who runs the database project with a few IT professionals assisting him or her. As the name of the phase indicates, this is a study phase. They look at the organization as a whole, the IT resources, and user readiness to assess what more has to be done before a database system could be implemented in the organization. This phase is critical because getting immersed in a database project without proper readiness assessment may result in project failures. Some organizations may even decide to put off implementing a database system until they are ready for it.

We will now walk through the objectives, activities, and crucial issues of the feasibility study phase.

Objectives Assessment of readiness for a database system implies taking inventory of the resources already available in the organization. The business analysts must evaluate each resource and determine whether it is adequate for the proposed database system. If the resource is lacking in functions and features, then the analyst must estimate what it will take to enhance the resource. For example, if the server hardware needs upgrades, the types of upgrades must be estimated and specified.

When an organization implements a database system, many new types of resources become necessary. The feasibility study phase reviews the available resources and lists the new types of resources the organization will have to procure.

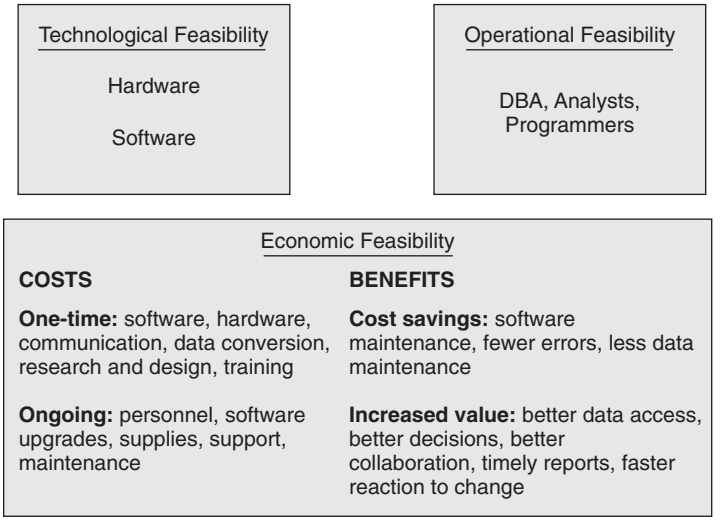


Figure 4-9 Feasibility study phase: three areas.

Broadly, the objectives of this phase include specifications of the enhancements to the already available resources and recommendations on the purchase of additional resources. Also, the study phase lists the action items for accomplishing the recommendations to improve the readiness of the organization.

Three Types of Feasibility We may think of the feasibility study phase as studies in three areas—technological, operational, and economic. This helps break down the activities of the phase into convenient and natural sets. To complete the activities in each area, business analysts rely on the help of different groups of database practitioners and user representatives.

Figure 4-9 presents the three areas for feasibility study. Note the components that analysts will have to study in each area. Analysts use assistance from the information technology department for the study of technological feasibility and operational feasibility. Analysts work with IT personnel and user representatives for the study of economic feasibility. Outside agencies also provide information to complete the feasibility study phase.

Action Items Out of the feasibility phase comes a comprehensive list of action items. Business analysts may adopt the division of action items to correspond to the study in the three areas—technological, operational, and economic. Whatever format the list may take, it is important to include all the needed enhancements to existing resources and all the additional resources to be obtained.

Timing The listing of action items must be accompanied by recommendations on when each action item must be completed. For example, if one of the action items recommends adding two data modelers to the team, this action will be useless unless the data modelers are recruited or trained before the design phase commences. Indication of the timing for each action is absolutely essential.

Technological Infrastructure

Business analysts first look at the technological infrastructure to assess its adequacy for the proposed database system. Technological infrastructure includes both hardware and systems software. This is the platform on which the database system will operate. Business analysts size up the proposed data volumes and the estimated number of transactions to study whether the available infrastructure can support the database system. The technological feasibility study involves taking an inventory of existing hardware and systems software; the study usually results in recommendations for additional hardware and upgrades of the systems software.

Hardware Consider the hardware environment for the proposed database system. The database will usually reside on a database server machine. Database management system (DBMS) software will run on the server machine. How big and powerful the server machine has to be depends on the size and use of the database. If the data content were large, you would need ample storage and a choice of better media for storing the data. If rapid data access were critical, you would need faster storage media. In a database environment where information is shared, keeping the database available at all times becomes crucial. Therefore, storing data in fault-tolerant storage media is desirable. Most database systems use RAID (redundant array of inexpensive disks) storage technology. This technology provides duplicate versions to be stored with mirroring techniques. We will focus more on this technology later, but for now just note that many aspects and options must be considered for database storage.

What about the applications for the users to draw on the wealth of data stored in the proposed database? Applications in a database environment tend to be more sophisticated than those in a file-oriented environment. Database applications multiply quickly. Because these applications are more standardized and uniform, their spread to many user groups becomes easier. Where should the applications run? On the client machines? If the proposed applications in your organization are expected to be more involved, large, and widespread, the business analysts doing the feasibility study may consider running the applications in the middle tier on separate application servers. This option will also make deployment of the applications easier.

Now turn your attention to client machines. How many additional client machines does your organization need? What about the speed and capacity of each machine? The business analysts need to take an inventory of the client machines in each user department and assess the number and types of additional machines required.

Study of the hardware configuration also includes review of the communication network links. This is very important if the user groups are spread out in multiple geographic regions domestically or internationally. Will the network configuration be adequate to deliver data from the proposed database? What about the public carriers? What about the sufficiency of the bandwidth? Will the Internet be a delivery medium in your database environment? These are some of the major questions the business analysts must consider.

Systems Software Database management system software (DBMS) interacts with the operating system for accessing data from the database. Every query or

application data request gets executed by the DBMS, which in turn depends on the operating system for input and output operations. The operating system must be robust enough and compatible with standard commercial database management systems. In a database environment, security, reliability, and scalability are important considerations in the choice of operating systems. The business analysts, with the help of other database practitioners, ascertain the necessary capabilities of the required operating systems. The analysts make recommendations for version upgrades to the existing operating system or for a new and more powerful system.

The business analysts must also look closely at the existing network operating system. Now with a database system, network traffic patterns and data transfer volumes are likely to change drastically. This is especially true in a distributed database environment. Will the current network operating system be up to the task in the proposed environment? Can it be upgraded or augmented with third-party tools? Or should the current network operating system be replaced with a better and more suitable system?

Skills Review

Review of skills and listing of recommendations form the key activities of the operational feasibility study. Business analysts must proceed systematically to list the necessary skills demanded by the database project. Next, they must take an inventory of the skills currently available in the organization. On the basis of the information gathered in these tasks, the business analysts make recommendations for actions to make up the deficiencies.

Necessary Skills The first part of the operational feasibility study entails making a list of all the skills necessary for the successful completion of the project. The business analysts must seek the help of others, including the project manager and the database administrator, to determine the types of skills needed for each phase of DDLC. If the database administrator function is not fully staffed yet, this fact goes on the list first. Each phase needs different kinds of special skills. You need people with different types of experience and training in the various phases. At this point, do not make a list of the types of persons needed but only the types of skills needed. For example, in the design phase, you would need data modeling skills. In the requirements definition phase, you would need interviewing skills.

After determining the types of needed skills, the business analysts can translate and relate these skills to the qualifications of the types of people needed for the project. Sometimes, you may be able to find people with multiple skills who can be assigned to more than one type of activity. A person well versed in logical database design may also be able to lay out the physical files and records.

People Inventory What is the availability of people for the database project in your organization? Do you have analysts who are experts in interviewing techniques? Are there data modelers? Do you already have database administrators? Do you have programmers experienced in writing data conversion programs? The business analysts have the task of noting the skills and experience of each person already assigned to the project. This must be done in sufficient detail so that the right person may be given the right assignment.

If some people possess more than one skill that can be used in the project, this can be effective in a project of reasonable size. In large projects, allocation of different assignments to the same person may not be practical.

Matching Up Once the business analysts have a list of the skills necessary for the project and an inventory of the available personnel, the next task is one of matching up the skills with the people. If the project manager is not directly involved in the feasibility study, this task calls for his or her participation. The project manager and the business analysts determine how the skills and the people match up best and estimate the deficiencies.

Wherever people are not present to handle specific skills, suitable people must be found for the project. Here are the three common options for filling the deficiencies:

Training insiders. First look at the prospects inside the organization. Some qualified candidates within IT or within the rest of the organization would welcome a transfer to the database project. If these candidates can be tested for their aptitude, they could be trained before being assigned to the project. Some programmers without any database experience can be trained easily and quickly. If the project schedule is flexible enough, selecting and training loyal insiders can rank as the best and easiest option.

Recruiting outsiders. What do you do if there are not enough qualified people inside the organization who can be trained for the database project? Then, of course, you have to recruit from outside. If this is a viable option for your organization, sufficient lead time for recruitment must be built into the project schedule.

Hiring consultants. Consultants can always fill the gap wherever suitable talent is not available from within or outside the organization to be included as employees on the project. Nevertheless, consultant assignments must be very specific in terms of tasks and time. While the particular tasks are carried out, some employees may be trained for those tasks. The assignments of the consultants terminate as soon as the specific tasks are done.

Estimation of Costs

While conducting the technological feasibility study, the business analysts estimate the additional hardware and software requirements. The operational feasibility study produces a list of what is needed for training, recruiting, and assigning people to the various phases of the object. In effect, these two activities in the feasibility study phase result in a list of technical and people resources. How much is it going to cost to get these resources? How long in the project are these resources likely to be used? Which of the costs are one-time costs, and which of these are ongoing?

The first part of the economic feasibility study determines the major costs of the project. Every database project has one-time costs incurred before the implementation of the database system. Costs for some items such as personnel continue for ongoing maintenance of the system even after implementation. It is easier to estimate the costs by separating the costs into one-time and ongoing costs. By so

dividing up the costs, you are not likely to omit any costs. Some items like database software have an initial one-time cost to purchase the item and then ongoing cost to maintain the item after implementation.

One-Time Costs Let us make a list of the common one-time costs. Although most of the items on the list are standard for every database system, the project in your organization may include a few new items. Review the list and adapt it to suit your database system.

Hardware. Include all necessary upgraded servers—additional memory, storage, and processor power. Add enhancements to client machines. Estimate additional communication equipment and cabling. Use quotations from vendors to prepare cost estimates.

Systems Software. Most of the time, later versions of existing operating systems will be able to handle the requirements of a database system. Estimate version upgrade. If your organization needs to go to another more robust operating system, then systems software costs could be substantial. Also, include the costs of any specialized utility tools such as backup programs or file reorganization software.

Database Software. This is perhaps the largest portion of one-time costs, almost in the same range as costs for major enhancements to hardware. Estimate costs for database software by obtaining quotations from a few vendors of leading DBMSs. Also, include cost of any necessary CASE tools in this category.

Communications Software. A database system in a distributed environment presupposes a strong communications system. If this is the case for your organization, allocate costs for upgrades to the existing network operating system or even for a more sophisticated replacement that can control potential increase in communication traffic.

Training. Training costs relate to training any IT personnel being transferred to the database project and to training other people from other departments suitable for the project. It would cost more to train people from other departments, but these training costs would offset recruitment costs if people would otherwise have to be recruited from outside.

Recruitment. Include search and recruitment fees for getting qualified people for the database project. People with specialized skills in database techniques and concepts attract larger recruitment fees.

Studies. In addition to the initial feasibility study, other studies become necessary in a database project. If your system needs elaborate data modeling because of its enormous size, then you would need studies to assess and purchase a good CASE tool. Another study may be warranted to select a new network operating system. In all projects, a special study is needed for examining the options and selecting the proper DBMS.

Materials and Supplies. Do not overlook costs for supplies and materials. Make a provision for these costs in your estimation.

Ongoing Costs Once a specific item is purchased, that item attracts ongoing costs. Hardware will have to be maintained; software upgrades will be essential; personnel costs continue. Ongoing costs begin during the various phases of the project itself. When the project is completed and the database system is deployed, many of the costs continue. Personnel costs to administer and manage the database system continue beyond the implementation phase.

Estimate ongoing costs for the following items:

- Hardware maintenance
- Software upgrades
- People
- Support system
- Training
- Materials and supplies

Assessment of Benefits

Not all of the benefits of a database system are derived directly. In other words, it is hard to estimate many of the intangible benefits although such benefits empower the users and produce great positive impact on the business. Unlike the estimation of costs, assessment of benefits poses difficulties. The general method consists of coming up with estimated numbers for tangible benefits and explicitly expressing every intangible benefit in the feasibility study report.

A database system results in cost savings in several areas. Make a list of the cost savings. Compare your current data environment with the proposed environment and assess all possible cost savings. How will the proposed database system make collaboration among the various departments easier? What synergies will result from information sharing? How will the proposed system improve customer service? How easy will it be to get all information from one data source? Consider such questions to produce a list of intangible but significant benefits.

Intangible Benefits

Data consistency. A database system contains consistent data values. For a particular data item, the same data value exists irrespective of who retrieves the data. Your organization will present consistent data values in the documents sent to customers and business partners.

Data integrity. It is possible for a database system to enforce edits and checks for data entered into the database. The database system expects to store more reliable and correct data.

Data availability. Data backups and recovery procedures are more standardized in a database environment.

Stricter controls. You can control data access better in a database system. Database management systems provide better methods for granting and revoking access privileges.

System flexibility. Changes to business conditions occur rapidly and unexpectedly. If your organization can react to changes quickly and take appropriate action, it will have a competitive edge. A database system provides flexibility for adapting to changing business conditions.

Information sharing. A database system promotes information sharing and results in closer collaboration among users.

Better standards. In a database system, it is easier to establish standards on data names, edit rules, data types, field lengths, data access patterns, and so on.

Timely information. A database system provides timely and up-to-date information on business objects.

Improved productivity. Because of standardization and ease of use, you will notice improved productivity in the way users are able to access and use data from a database. Many data operations are provided through query and reporting tools that are part of the database management system. This eliminates the need to write programs for these purposes. Programmer productivity increases in a database environment. Sometimes it is possible to estimate cost savings relating to improved programmer productivity.

Better management decisions. Integrated and accurate data in one place in the organization's database, allowing easier and faster data access, enable better management decisions. Managers and supervisors can obtain timely reports of data needed for managing and controlling their resources.

Cost Savings

Minimal data duplication. As you know, hardly any duplication of data exists in a database management system in comparison with file-oriented data systems, where you observe a proliferation of duplicate data. Elimination of data duplication means less overall storage space for storing an organization's data content. Estimate the cost savings from using less storage space.

Fewer error corrections. Minimal data duplication also results in less effort to correct errors or updates to data. If the same data, say a customer's name and address, are duplicated across many applications, any changes to the name and address must be applied in several places. However, in a database system, the name and address of a single customer is stored only once; therefore, any corrections to this name and address need to be made only once in only one place. Try to come up with a reasonable estimate on the possible savings due to reduction in error corrections.

Less programming. Application programs do not require elaborate coding for data retrieval and update operations; the database management system will interpret simple data access coding and complete the operations. Moreover, separate programs are not needed for writing most queries and reports. Query tools and report writers will do the job.

Reduced training costs. As we have seen above, applications can be standardized in a database environment. So it becomes easier to train users across applications. This brings about a general reduction in training costs.

Weighing the Options

Before the business analysts reach the end of the technological and operational feasibility study activities, they collect enough material to list and evaluate the options. Consider physical hardware, especially the database server machines. Should you upgrade the existing servers, or switch to a new brand, more powerful but also more expensive? Weigh the available options at each decision point and make recommendations.

The feasibility study phase concludes with the publication of a comprehensive report. The study phase not only answers whether the organization is ready for a database system at all but also makes an assessment of the organization's preparedness. Make suggestions on the available options. Show the costs and benefits for each option.

Here is a list of standard contents for a feasibility report:

- Introduction and scope
- How costs and benefits are estimated
- Assumptions derived from planning
- Hardware
  - Current status
  - Proposed enhancements
- System Software
  - Current status
  - Proposed upgrade
- Database Management System
  - Major features needed
  - Options and vendors
  - Recommendation
- Costs
  - One-time
  - Ongoing
- Benefits
  - Intangible
  - Cost savings

- Final Recommendations
  - List of options
  - Suggested recommendations

REQUIREMENTS DEFINITION

Proper definition of information requirements drives the design and deployment of the database system. The earlier planning phase sets the tone and direction for the project. The feasibility study phase provides the justification and examines the options for the project. But it is the requirements definition phase that lays the foundation for the system. The importance of requirements definition as a driving force cannot be overemphasized. Every organization launching a database system must pay special attention to how the requirements definition phase is conducted.

In this phase you observe the overall business, you understand the core business, you examine the primary processes, you find out who needs what data to run which processes, and you determine the complete information requirements. You study the data volumes and data access requirements to estimate storage space and the exact type and features of the database software. Note the following summary list of major tasks during the requirements definition phase:

- Study overall business operations.
- Observe business processes.
- Understand business needs.
- Interview users.
- Determine information requirements.
- Identify data to be collected and stored.
- Establish data access patterns.
- Estimate data volumes.
- Document the information requirements.

Who is involved in requirements definition? Practically, everyone on the project may have some input, but usually one or more data analysts, ably supported by a team of systems analysts, initiate and control this phase. In every task, the user representatives must participate actively. Technical know-how will have to come from the database administrator.

Data analysts and systems analysts working on requirements must have had training and experience in gathering requirements and documenting requirements according to accepted standards. They have to work closely with user groups and understand their information needs; this requires training in interviewing and documenting techniques. They must know how to work with users to collect the information needs and have the gathered type of information requirements confirmed by the users. Representatives for users at different responsibility levels, in various functional units, and at several geographic locations—all must make the phase successful. These analysts must also have access to existing systems and documents.

Although information requirements are derived by observing business functions and processes, the approach must be clearly data-oriented. In other words, the emphasis is not on how the functions and processes are carried out but on what data are required for performing the business functions and processes. This is a new mind-set for analysts who had previously worked on designing and deploying applications. In applications, you are designing the "how" of the processes. But in a database system, you are designing the "what" of the data requirements. So keep requirements definition for a database system data-oriented. Come up with the elements of data required for each process and for each function.

At the end of the phase, the analysts pull all the data requirements together and consolidate them. This aggregation step eliminates any duplicate statements of requirements and aggregates all data required for individual business objects. A requirements definition document is produced to conclude the phase.

Requirements Gathering Methods

It may appear that the requirements definition phase is a formidable effort because of its extreme importance. Many project teams think of the project as mainly consisting of design and implementation. However, requirements drive these later phases, and without properly defined requirements the phases that follow cannot be successful. If requirements definition is so significant, are there standard methods that can be applied universally?

Undoubtedly, each organization will have its own special requirements and these special requirements may warrant innovative techniques. Nevertheless, most requirements gathering methods follow a set pattern. You now know the purpose of this phase and what is expected at the end of the phase. But how do you reach the desired end and document the information requirements?

A summary list of the major standard methods follows. In the subsequent subsections, we will elaborate on a few of the important ones.

Interviews. This is a common technique. Personal interviews cannot be replaced by other methods. Still, one-on-one interviews produce the best results.

Group sessions. When it is convenient and desirable to meet a user group as a whole, group sessions work well. Here you save time by not interviewing the people in the group one by one.

Observation of processes. The combination of all processes within an organization makes up the entire business. So if you review each process beginning to end, you will be able to get a handle on what information is needed at the several points along the way.

Review of applications. The current computer applications are used to run the business now. These applications store, retrieve, and update data. The storage, retrieval, and update techniques may not be satisfactory. That is one of the main reasons for moving to a database system. However, much can be learned from the data elements used by the existing applications.

Study of documents. Existing documents give ample indications as to the type and content of the data needed for the organization to run the business. Planning and policy documents indicate long-term data needed for achieving the company's business goals. Reports printed from current applications form a valuable source for requirements gathering. Layouts of the current files are good starting points to determine the information requirements. Procedure documents for individual processes indicate what data are required to perform the processes.

Conducting User Interviews

The first question is, Whom do you interview? Who are the users of the proposed database system? Broadly, we may classify the users as follows:

Senior executives and division heads
Departmental managers
Operational staff
Business analysts
IT staff maintaining computer files

Senior executives will give you a sense of the direction and scope for the database system. You can derive information requirements for future expansions and changes in business strategies. Departmental managers can provide information about the type of control and exception reports they would need and the data content of those reports. Interviews with operational staff will produce the bulk of what you need in this phase. That is where most of the emphasis must be placed. Business analysts can tell you the types of data they use for conducting studies and producing results for the executives. A lot of data requirements will be available from IT people such as systems analysts, programmers, and operations staff who maintain the current applications and their computer master files and transaction files.

When you gather data requirements from each group, separate the requirements by business objects. The requirements you collect will comprise of the following:

- Data elements
- How and who uses the data elements
- When and where the data elements are used
- Business rules governing the data elements
- Data volumes

You may consider the interview process as consisting of the following steps:

- Decide on the list of persons to be interviewed.
- Prepare for interviews.
- Conduct interviews using appropriate techniques.
- Document interview findings.
- Follow up and confirm interview write-ups.

You will have to go to various user groups and IT personnel to gather information requirements. Analysts adopt two basic techniques for meeting with people and gathering requirements, (1) interview, one-on-one or in small groups, and (2) joint application development (JAD) sessions. Note the following thoughts on individual interviews and group sessions. We will consider JAD sessions in a separate subsection.

Interviews

- Up to two or three persons at a time
- Easy to schedule
- Good approach when details are many and intricate
- Some users prefer one-on-one interviews
- Need adequate preparation to be effective
- Always require preinterview research
- Users also must be encouraged to prepare for the interview

Group Sessions

- Groups of 20 or fewer persons at a time
- Useful only after obtaining a baseline understanding of requirements
- Not good for initial data gathering
- Effective for confirming requirements
- Sessions need to be well organized

Interview Techniques Interview sessions, when properly conducted, take up a lot of the project time. Therefore, interviews must be planned, organized, and managed well. Before the analysts begin the interview process, the project team must make sure that the following major tasks are completed:

- Provide further training in interviewing techniques to the selected analysts.
- Assign specific roles to the members of the interviewing team (lead interviewer, scribe, and so on).
- List the persons to be interviewed.
- Prepare a broad schedule of interviews.
- Note the expectations from each set of interviews.
- Complete preinterview research.
- Prepare interview questionnaires.
- Prepare users for interviews.
- Conduct a brief kick-off meeting of all users to be interviewed.

To a large extent, the effectiveness of the interviews rests on the preinterview research. This task provides you with the necessary background information on which the requirements definition may be built. Here is a list of a few topics for preinterview research before interviewing a specific business unit:

- History and organization of the business unit
- Number of employees and their roles and responsibilities

- Locations of the users
- Preliminary purpose of the business unit in the organization
- Contribution of the business unit to the overall core business
- Secondary purpose of the business unit
- Relationship of the business unit to other business units

Here are some tips on the types of questions for the interviews:

Major business processes. What are the business processes for which your unit is completely responsible (example: completely responsible for the process of preparing invoices to customers)? What are the business processes that your unit is not completely responsible for but participates in with other business units? Briefly describe each process your unit is involved in.

Current information sources. Do your processes generate all the information needed by your unit? If not, what are the sources of information you need for your business processes? Which reports do you use for running your business?

Timeliness of information. Do your processes require real-time data? What are they? Which are the on-line data types? Which types of data used are a day old, a week old, or a month old? Does your unit need historical data? If so, for what purposes?

Major computer applications. What are the computer applications used in your unit? Which are the important reports? Which are the input screens? What are the types of data input by your unit? Which display screens are important to your unit?

Data volumes. How much of the data your unit generates is stored in the computer files? Give volume estimates. How much of the data your unit uses, but does not generate, is stored? Give volume estimates.

Document each interview with a write-up adopting the following general outline:

1. Profile of the person or persons interviewed
2. Background and objectives
3. How the unit is managed and controlled
4. Business processes
5. Business objects
6. Data requirements related to each business object
7. Current computer screens used
8. Current computer reports used
9. How success of the business unit is measured

JAD Methodology If you are able to collect a lot of baseline information up front from documents, group sessions may be a good substitute for individual interviews. In this method, you are able to get a number of interested users meet together in groups. On the whole, this method could result in fewer group sessions than

individual interview sessions. The overall time for requirements gathering may prove to be less and therefore the project may be shortened. Also, group sessions may be more effective if the users are dispersed in remote locations.

Joint application development (JAD) techniques were successfully used to gather requirements for application projects in the 1980s. Users of computer systems had grown to be more computer savvy, and their direct participation in the development of applications proved to be useful.

As the name implies, JAD is a joint process with all concerned groups getting together for a well-defined purpose. It is a methodology for developing computer applications jointly by users and IT professionals in a well-structured manner. JAD centers on discussion workshops lasting a certain number of days under the direction of a facilitator.

JAD consists of a five-step approach:

Project Definition

Complete high-level interviews.
Conduct management interviews.
Prepare management definition guide.

Research

Become familiarized with the business units and systems.
Document user information requirements.
Document business processes.
Gather preliminary information.
Prepare agenda for the sessions.

Preparation

Create working document from previous phase.
Train scribes.
Prepare visual aids.
Conduct presession meetings.
Set up venue for sessions.
Prepare checklist for objectives.

JAD Sessions

Open with review of agenda and purpose.
Review assumptions.
Review current applications.
Review data requirements.
Discuss timeliness of required information.
Examine data volumes.
Resolve all open issues.
Close sessions with lists of action items.

Final Document

Convert the working document.
Map the gathered information.
List all current data sources.
Identify all future data requirements.
Assemble and edit the document.
Conduct review sessions.
Get final concurrence from users.
Establish procedure to change requirements.

The success of a project using the JAD approach depends on the composition of the JAD team. The size and mix of the team will vary based on the scope and extent of the proposed database system. The typical composition, however, must have certain roles present on the JAD team. For each of the following roles, usually one or more persons are assigned.

Executive Sponsor Person controlling the funding, providing the direction, and empowering the team members

Facilitator Person guiding the team throughout the JAD process—usually the project manager or lead analyst

Scribe Person designated to record all decisions

Full-Time Participants Everyone involved in making decisions about the database system

On-Call Participants Persons affected by the project, but only in specific areas

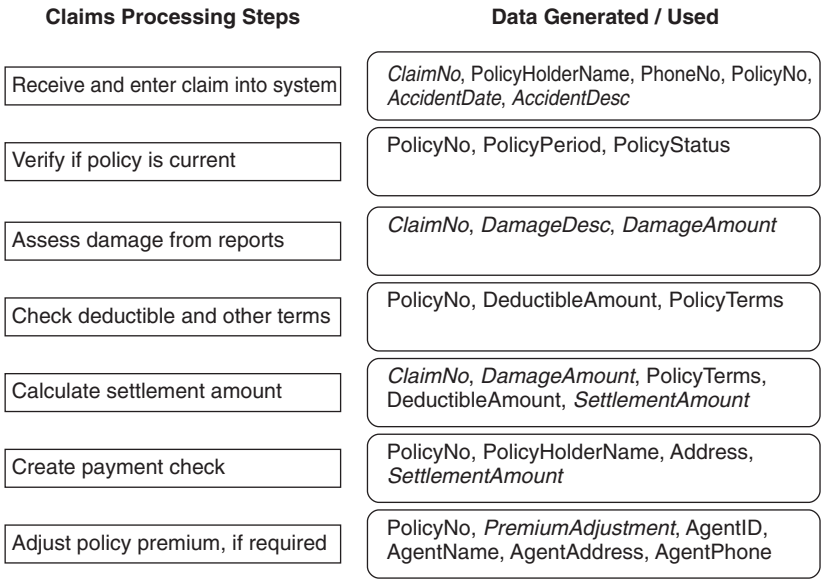
Observers Persons who would like to sit in on specific sessions without participating in the decision making

Observation of Processes

Almost all of the data requirements will be collected through one-on-one interviews or through group sessions. This step of observing the business processes serves two purposes. First, this step enables you confirm the data requirements gathered through interviews. Second, the step provides an opportunity for you to collect any requirements that were missed during the interviews.

You know that every business process within the organization generates data that must be stored for use by the same process itself or by other processes. Furthermore, most processes make use of data generated by other processes. What exactly do these two concepts mean—generating data and making use of data by a process?

Consider the process of settling an automobile claim by an insurance company. This is a primary process for the claims processing business unit of the company. The process begins with the receipt of an automobile claim from one of the policyholders. The person or persons in the company responsible for processing the claim get some data from outside and enter them into the computer system. These are the data generated by the process to be stored in the database. At the same time, the



NOTE: Data generated by this process shown in *italics*.

Figure 4-10 Claims processing: data requirements.

process needs to make use of some data elements generated by some other business unit. Figure 4-10 illustrates data generated and used by this process. Note each step of the process and observe the data generated and the data used at each step. Also, see how the data requirements are tied to business objects that are relevant to the company.

When you finish confirming the requirements through the observation of business processes, your requirements definition solidifies. Any vagueness in the requirements determined through interviews is removed. We have two more steps that accomplish similar confirmation of requirements: review of applications and study of documents. Let us go over these two steps before finishing the requirements definition phase.

Review of Applications

During the interviews, you would have completed a list of all the current computer applications that support the various business processes. In fact, you would have noted all the applications reports and screens that are especially important to the user groups. Now in this step you will use the compiled list of applications, reports, and screens to review the data requirements. This step is another step to augment the requirements gathering through interviews.

While reviewing the current applications, you are not interested in how the users navigate through the menus and screens. That goes with the design of applications. In DDLC, you are finding out what data are required to support the applications.

You are not generally concerned with the application structure. Review the applications not for functions, but for data requirements.

Review of applications consists of examining the following for data content and relationships:

- Input screens
- Screens only for display of information
- Reports
- Business rules covering relationships among business objects
- Edit rules for individual fields

Study of Documents

Again, this is another step to confirm the requirements gathered through interviews. For the interviews you need the involvement and participation of the user groups. You have to work around their daily responsibilities and to schedule time for the interviews. However, you can study existing documents without too much involvement from the users. Scheduling the study of documents involves only the analysts performing the activities of the requirements definition phase.

Documentation from User Groups What information can you obtain from the existing documents? In many organizations with old computer systems, not much documentation is readily available. You will have to piece together several individual documents to the complete the picture. If you have documents recording the screen formats and report layouts, study those in detail. These documents will complete the information about the data currently used by the various departments. If you have additional documents on the rules and edits, go through those as well.

Documentation from IT Go to the people in IT responsible for maintaining the files of the current applications. These are the technical support persons who are responsible for backing up the files and expand space for files, and so on. Collect the layouts of all the files. Study of the layouts will yield valuable information about the data that support the current applications. In IT, programmers and systems analysts are assigned to support specific applications. If you have an order processing application, find the analysts and programmers supporting this application. Obtain all relevant documents for your study.

Consolidating Information Requirements

We have reviewed a few methods for collecting and confirming data requirements for the proposed database system. You understand that interviewing users and IT professionals is still the best method for gathering requirements. Now we are at the stage of consolidating everything that has been collected and formalizing the data requirements. In a large project in which many analysts are working on the activities of the requirements definition phase, they would have accumulated many pieces of information. How do you aggregate and make sense of everything that has been collected?

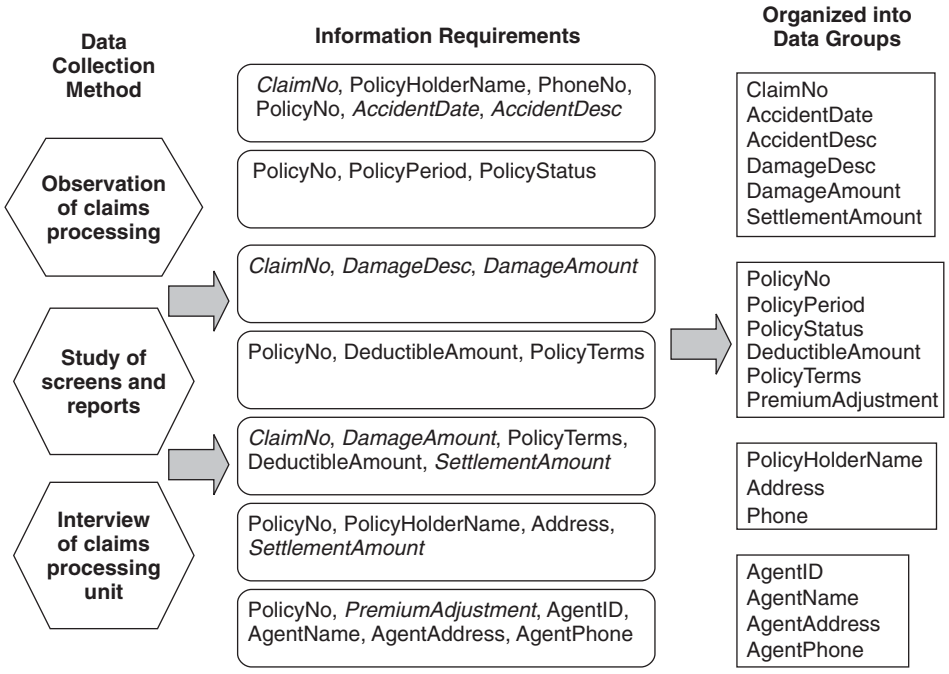


Figure 4-11 Claims processing: organizing data groups.

The consolidation step is intended to organize the collected requirements in a way that would make sense and be useful for the design phase. So what are the tasks in this step? Let us say that you have collected requirements from the interviews with the users in the claims processing unit and also gathered requirements by studying the screen formats and the report layouts of the claims processing application. You would have collected most of the information through the interviews and additional information through the study of the screens and reports. How do you consolidate and record what you have collected? Figure 4-11 illustrates the method for organizing the information collected and recording these by business objects. Observe how the data groups are formed for the process of settling claims.

The outcome of this step is the organized data groups. Each data group relates to a single business object, and each data group has been derived from the information gathered for a single process through the standard methods.

Requirements Definition Document

Publishing a requirements definition document signals the end of the requirements definition phase. The analysts combine all the information collected through different methods and prepare the document. After a draft is prepared, it must be reviewed and confirmed by the users. This is absolutely essential to ensure that you have not overlooked any data requirement. Also, the review and confirmation task corrects any misunderstandings and clarifies ambiguous requirements.

Here is a suggested outline for the requirements definition document. Amend and adapt the outline to suit your database project.

1. **Introduction.** State the overall scope and purpose of the project. Include broad project justification. Provide an executive summary for each subsequent section.
2. **Requirements collection methods.** Describe the methods used for collecting requirements. List departments and business units interviewed. List documents studied and applications reviewed.
3. **Interview summaries.** Include summaries for all key interviews. If you used JAD methodology, attach JAD documentation.
4. **Data groups.** Provide information requirements in the form of data groups compiled in the previous consolidation step. Pay special attention to this section as it forms the foundation for the design phase.
5. **Future information requirements.** Include the requirements gleaned from those interviewed and inferred from the planning documents.
6. **Other information requirements.** In this section, put down other requirements such as data volumes, remote locations, data access patterns, data conversion needs, and business rules.
7. **User expectations.** Record all special expectations in terms of data timeliness, data integrity, problems, and opportunities.
8. **User review and sign-off.** Describe the procedure used for getting the requirements reviewed and signed off on by the users. Also, indicate how users may request changes to the requirements definition and how the changes would be accepted for inclusion.

THE DESIGN PHASE

As noted above, the outcome of the requirements definition phase drives the design phase and the subsequent phases. Very specifically, the design phase transforms the requirements collected in the previous phase into a design for the proposed database system. All the materials and information gathered in the requirements definition phase are used to produce the design. The product of the design phase generally takes the form of a design diagram together with descriptions and explanations.

In the design phase, you find answers to questions such as, How is the data required by the users to be organized and structured? How is the data going to be stored in physical storage? What are the file layouts? What are the data structures? When you produce the design for a database system, you do so at two levels—one at the level of physical storage and the other at a higher level showing just the data structures. As you know, this type of separation of the design provides data independence. Changes at one level need not propagate into changes at the other level.

Later chapters deal with the design of a database system in more detail. Chapters 5, 6, and 7 elaborate the topic of data modeling that is part of the design process. Chapters 8, 9, and 10 deal with design specifically with regard to the relational model, which is a major focus of this book. Chapters 11 and 12 also relate to the design phase. Therefore, at this stage, when we are discussing the entire database development life cycle (DDLC), we will be fairly brief about the design phase. Enough will be said later. So just try to note the general concepts of design here.

Design Objectives

Before we briefly describe the activities of the design phase, let us understand the main objectives of this phase. We want to represent the information requirements in a format that can describe how data structures are perceived, how the structures are related, and how data are to be stored. Data are stored for access and usage. The ability to navigate through the structures and retrieve data must also somehow be portrayed in the design. The design must express the data content.

Here is list of major objectives for the design, both at the level of physical storage and at the higher level indicating the data structures:

- Show data content in detail.
- Represent data relationships.
- Indicate how data structures may be accessed easily.
- Denote information sharing.
- Introduce data independence.
- Secure and safeguard data.
- Establish data integrity and consistency.
- Allow easy data management and administration.

Logical Versus Physical Design

As noted above, the design phase produces two levels of design. Design at the two levels represents two levels of data abstraction. At one level, you design the data structures and arrange them in a manner that expresses their relationships. If your database consists of data only on customers and orders, then the data structures in one level of abstraction will represent the logical view of the data. This is the logical design. At the other level of the design, you will indicate the files, blocks, records, fields, data types, field lengths, keys, indexes, and so on. This level of design that relates to physical storage is the physical design.

The following summarizes the distinction:

Logical design. This is a high-level view of data contained in the entire database. This is a conceptual view of the whole database. It consists of data structures and their relationships. Consider how each user group looks at the data in the database. A particular user group may be interested in only a few data elements from customer data and most of the data elements in the order data. Then, from outside, the view of the database for the user group consists only of the required data elements from customer and order data. This set of data elements is the external view for that user group. Therefore, in the design process, we come up with sets of external views for the various user groups. The combined external views constitute the logical design.

Physical design. The logical design shows the data structures; the physical design represents how data relating to these structures are stored on physical storage. The physical design takes the design closer to hardware. While performing physical design, you have to consider physical hardware features. The physical design step is the transformation of the logical design into a design in terms of hardware.

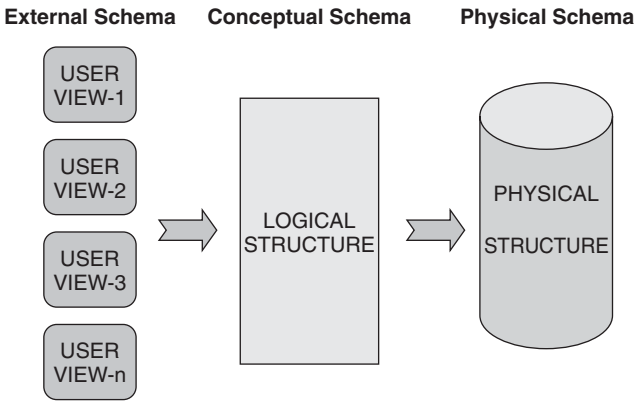


Figure 4-12 Logical versus physical design.

Figure 4-12 indicates the output from the logical design and physical design steps. "Schema" means a way of indicating the output of the design phase. "External schema" constitutes the set of external views of various user groups. "Conceptual schema" represents the structures of the database as a whole. Finally, "physical schema" consists of the files and records as stored in hardware. Creation of the external schema and the conceptual schema are generally reckoned as logical design activities; whereas production of the physical schema is the physical design step.

Go back to the example of the process of settling an auto insurance claim. Imagine that your proposed database just has the data required to perform this process. We will describe the external, conceptual, and physical schema for the database of an auto insurance company in the next three subsections. For the sake of keeping it simple, we will deal with only a few data elements.

The External Schema

Try to understand what exactly "external schema" means and why it is noted as *external*. Imagine that some user group is looking at the data structures represented in the database from outside, from a standpoint external to the database. What would that user group like to see in the database? The group is interested in several data elements inside the database, most probably not all of it. When that group looks at the database, for the group the database is a collection of data elements that are important to it; that set of data elements is the group's view of the database.

The external schema is the collection of such views of different user groups. Figure 4-13 clarifies the concept further with an example from the database of an auto insurance company. Note the separate user views that make up the external schema.

The Conceptual Schema

This is another level of the design. Let us step back to the concept of external schema. We described the external schema as the collection of data views of

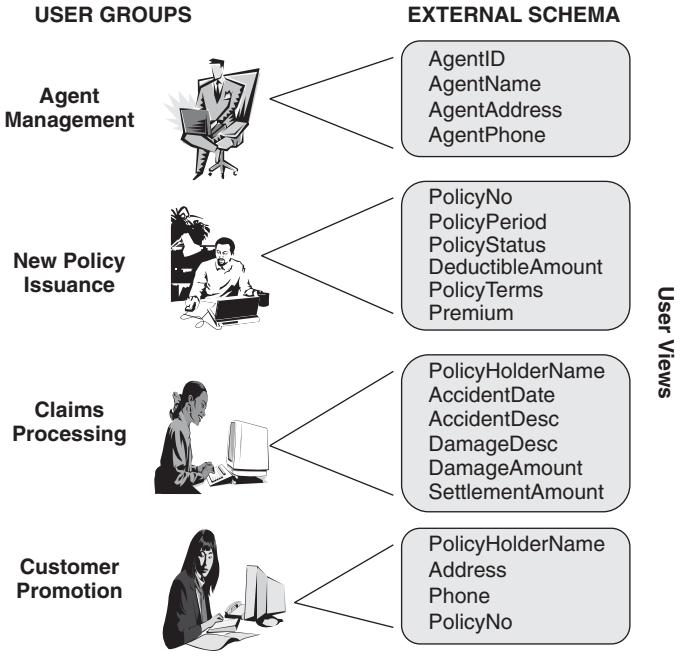


Figure 4-13 Auto insurance database: external schema.

individual user groups looking at the database from outside. Now let us ask the questions. What if there is a user group interested in every data element in the database? What will be the data view of such a user group? Of course, the view will comprise the entire database. The view comprising the whole database represents the concept of the complete database. It is known as the conceptual schema.

Now compare the two notions of external schema and conceptual schema. The external schema comprises individual views of the database by separate groups of users; the conceptual schema encompasses the whole database.

You must have already guessed it: the conceptual schema may be derived from the external schema by combining the various data views of individual user groups. It turns out that aggregation of the external schema is a good method for designing the conceptual schema.

Figure 4-14 shows the conceptual schema for the insurance database. Observe closely how the conceptual schema evolves out of the external schema. See clearly the similarities and differences between the two levels of design.

The Internal Schema

If the external schema and the conceptual schema represent views of the database from outside, the internal schema may be thought of as looking at the database from inside. From outside, the database is perceived as a collection of data structures and their relationships. How does the database look on the inside? If you look inside the database as it resides on physical storage, how does it appear? For this reason, the internal schema also goes by the name of physical schema.

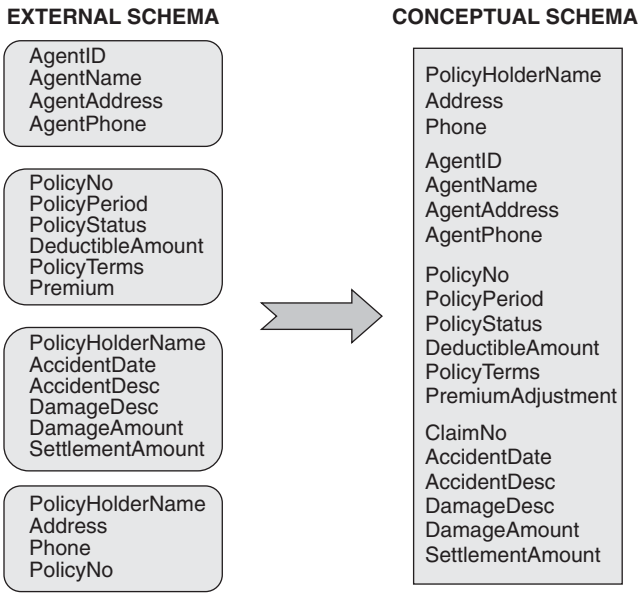


Figure 4-14 Auto insurance database: conceptual schema.

Figure 4-15 shows the components of the internal schema for the insurance database. Note that the internal or physical schema is made up of data files, index files, records, fields, and so on. Also, note how the internal schema is a translation of the conceptual schema to the level of physical hardware. In physical storage, data exist as files and records.

IMPLEMENTATION AND DEPLOYMENT

Before you can implement the database of your organization and deploy it for use, you must know what you are implementing. This knowledge comes from the design. We have noted the two types of design necessary for a database—logical and physical. We have also recognized the need for design at these two levels. We have seen how the external schema and the conceptual schema for the database result from the logical design process. Transformation of the conceptual schema into the internal schema results from the physical design process. Once you have the design at the two levels, you are ready for implementation and deployment of the database. Essentially, implementation consists of getting the data storage areas and files ready and placing initial data in these storage files. Deployment relates to making the database available to the users once the initial data is in.

Database administrators perform most of the activities in the implementation and deployment phase. Analysts also participate in this phase, but the primary responsibility rests with database administrators. Figure 4-16 indicates the major activities in this phase. Note how the responsibilities for the activities are shared.

Part V of this book covers topics on implementation and deployment of relational databases in greater detail. Therefore, at this stage of enumerating DDLC

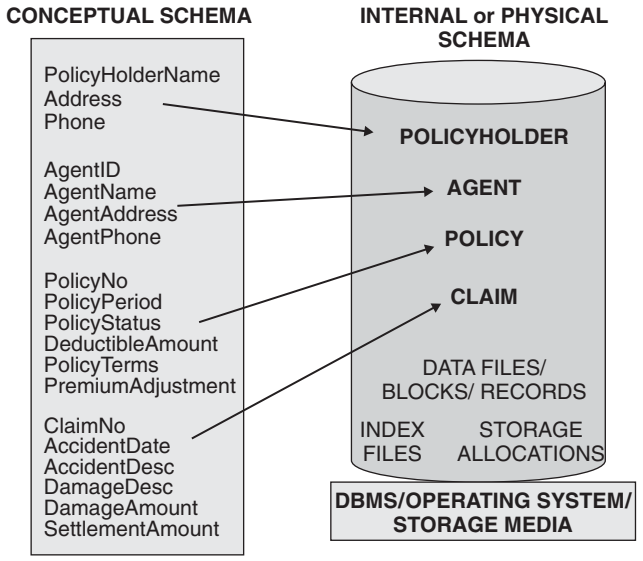


Figure 4-15 Auto insurance database: internal schema.

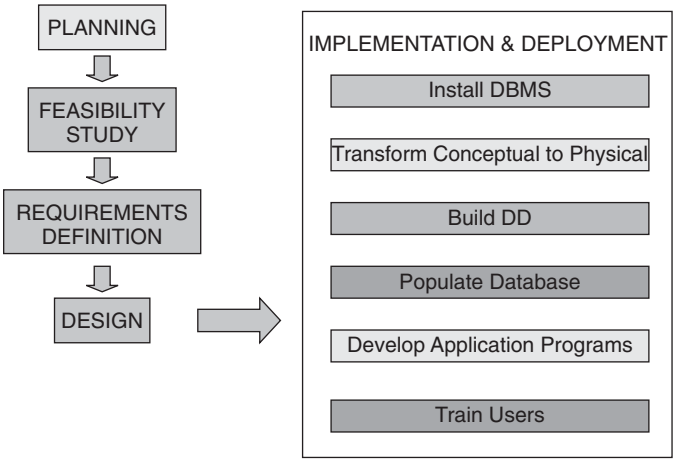


Figure 4-16 Implementation and deployment phase.

phases, we will just highlight the major activities. This will prepare you to appreciate the flow of the phases from requirements to design and then to implementation and deployment.

Conceptual to Internal Schema

Implementation of a database takes place at the physical level. When you implement a database, you are preparing for its usage at the level of physical storage and hardware. As you know, the internal schema represents the database at this level. Implementation of a database is, in effect, establishing the physical files in

accordance with the internal schema. How do you get to the internal schema? You get to the internal schema by transforming the data structures of the conceptual schema into the components of the internal schema.

The internal schema deals with the database at the level of hardware, operating system, and database software. Therefore, you need to take into account the following while formulating the internal schema:

- Features of the selected database management system (DBMS)
- Interface mechanism of the DBMS with the operating system
- Hardware configuration, especially the features of the storage devices

Refer back to Figure 4-15 and note the mapping between the conceptual schema and the internal schema. Also, observe how other factors influence the design of the internal schema.

DBMS Installation

Several considerations go into the selection of the proper DBMS and the right version that suits the needs of the organization. To select the appropriate DBMS, you check for flexibility, you make sure you can scale up, you verify compatibility with the operating system, and you examine features relating to easy data access. Once the selection is made, the project team procures the database software and the database administrator installs the DBMS.

A list of major installation tasks follows. These tasks are elaborated in later chapters.

- Ensure compatibility of DBMS with operating system.
- Obtain installation support from database vendor.
- Determine options to be installed (base product, language options, distribution options, etc.).
- Verify availability of disk space and memory for chosen options.
- Set installation parameters.
- Follow installation menu options and install DBMS.

Building the Data Dictionary

The data dictionary is a set of files to record the structure of the database. Before you can make use of the database, you must define the structure of the database in the data dictionary. The database software continually refers to the data dictionary for data manipulations. If you submit a query for customer names and addresses, the database software refers to the files in the data dictionary to find the field structure for customer name and address and also to find where customer names and addresses are stored. So before you can use the database, you must build the files that make up the data dictionary.

The database administrator defines the structure of the database in the data dictionary. The DBMS itself provides tools and procedures to enter this information into the data dictionary. The database administrator uses the conceptual schema to define the overall structure of the database, the external schema to define individ-

ual data views, and the internal schema to define the storage spaces and to tie the data structure to physical storage.

Populating the Database

What has taken place so far in the database development life cycle? The requirements definition was completed, and the requirements drove the subsequent phases. The logical design activities were finished. The logical design was translated into the physical design. The data dictionary files have been built. The storage space is ready. The data dictionary has all the entries necessary to define the data structures and relationships. What next? Next comes the activity of placing data in the database so that users can make use of the data.

You place data in the database in two stages. Is your database being converted from some other data system? Are you making the transition from a file-oriented data system? Or are you converting to a later database model? In any case, you will have to convert the data over to the new database system. This is the initial loading of data. Initial loading is usually done through utility programs provided in the DBMS package. Some organizations may opt to load initial data by using third-party tools. Generally, populating the database consists of the following major tasks:

- Extract data from the old data system and create extract files.
- Verify extract files for correctness and data consistency.
- Prepare data in the extract files by sorting and merging as needed.
- Create load files from the sorted and merged extract files to conform to the target database structures.
- Use DBMS load utility or third-party utility programs to populate database with initial data.

Developing Application Interfaces

Loading initial data is the first activity for placing data in the database. This is just the initial data from the old system. Once the database is up and ready for use, day-to-day business transactions can begin to store data in the new database. Ongoing transactions may add, update, or delete data from the database. This is the next activity in populating databases. Users now take over the task of manipulating data.

However, users need applications designed and built to be able to access data from the database. The applications needed to run the business processes must have interfaces to connect to the new database. Although applications may be thought of as part of the overall database environment, design and development of applications fall outside the scope of DDLC. The DDLC phases cover the design and development of databases; application design and development is usually accomplished through a similar life cycle approach—system development life cycle (SDLC).

MAINTENANCE AND GROWTH

The database development life cycle (DDLC) does not end with deployment of the database system and the users beginning to use the system. The development life

cycle continues beyond deployment. The database has to be kept up—24/7 in many organizations—and continually maintained. The database administrator continues to monitor the performance of the database and initiates measures to tune the database whenever necessary. He or she continues to check space utilization and the usage of other resources such as processors and communication lines. After the deployment, the project enters a continuous phase of maintenance and growth.

Compare the maintenance and growth stage of a database system with that of an earlier data system. Maintaining the database system and coping with its growth require more effort and specialized skills. The types of maintenance activities are different and more involved. In the earlier phases of DDLC, you find many database professionals taking the various responsibilities. During this phase, however, the roles of analysts, designers, and programmers are diminished; the database administrator takes the primary responsibility. During this phase, the following are the major functions of the database administrator:

- Ongoing monitoring of the database with the aid of usage statistics
- Tuning the database for performance
- Planning and executing tasks related to the growth of the database
- Managing changes to the structure and physical characteristics of the database

Here we have just highlighted the functions during the maintenance and growth phase. Chapter 17 is completely devoted to ongoing maintenance, and we will discuss more details in that chapter.

Administration Tools

You have noted that during the maintenance and growth phase, the database administrator and a few other database professionals are charged with many responsibilities. It is a dynamic environment, and these professionals must stay on top of so many facets of the database system. How are they able to perform their intricate functions? Fortunately, modern database management systems (DBMSs) provide appropriate tools to carry out these functions. This feature of the modern DBMS is even referred to as the toolkit feature. Today's commercial DBMSs come equipped with adequate toolkits. Even where the DBMS is lacking a specific tool, you can find third-party providers offering the missing tools.

DBMSs come with a complete set of DBA commands for the database administrator to evoke the functions of the toolkit. Commonly, the toolkit component of a DBMS contains tools and utility programs to perform the following functions:

- Load additional data as needed
- Import data from outside data sources
- Export data to outside data systems
- Create log files for recovery
- Supply usage statistics
- Backup database files
- Recover database after any type of malfunction

- Manage users and provide authorization for data access
- Manage disk space
- Create indexes for faster data retrieval or drop indexes that are no longer necessary
- Tune the database

Ongoing Monitoring

Monitoring the database is a primary daily function for the database administrator. Every day, the database administrator checks the space utilization reports produced through database utilities. The administrator must make sure that none of the database files is close to running out of space. If a file runs out of space during normal daily operations, users will be prevented from adding records to the file. For those users, the database becomes useless until the situation is rectified. Database administrators try to prevent such mishaps by keeping watch over space utilization through database statistics.

Database administrators continually monitor usage by the various user groups. They constantly monitor usage patterns to determine how each data structure is being accessed. Is there a need to establish new and faster paths for retrieval? Are there peak times when excessive usage overloads the system? Is there a steady increase in the number of users?

Another aspect of ongoing monitoring relates to database security. Database administrators must continue to prevent unauthorized access. They must monitor against illegal attempts at data retrieval and guard against hackers. This is especially significant where the database is open to the organization's business partners or even to the general public. Database administrators constantly scrutinize who uses the database and which files they normally use.

Database administrators continually monitor the database by analyzing database statistics produced by the DBMS utilities. These statistics enable them to take the necessary steps to tune the database whenever necessary and to facilitate smooth and uninterrupted growth of the database.

Performance Tuning

If the database project is successful and the implementation satisfactory, you will notice a marked increase in the number of users. More people will understand the ease with which they can obtain information from the organization's database. They will realize how information sharing could be effective in each individual process. Enthusiasm will increase rapidly.

More users mean more database transactions. The number of daily transactions accessing the database usually doubles within the first six months. The increase in the number of transactions is produced through standard applications as well as ad hoc queries for information from the database. At the outset, power users may begin with simple queries; but, as they become adept at using the database, they will execute more and more complex queries.

An increasing number of users, a rising number of transactions, and an increased query sophistication—all of these slow database access down over time. Retrieving

information from the database becomes difficult and time-consuming. The task of fine-tuning recognizes the performance problems of the database and sets out to improve performance. The database administrator, in coordination with the applications analysts and programmers, adopts measures to fine-tune the database. Essentially, the purpose of tuning the database is to make it faster and easier to retrieve information from the database.

Database administrators have a variety of options and methods for tuning a slow and lethargic database. They examine statistics collected by the DBMS and take appropriate action. The following is a list of the major methods:

- Create or drop index files to suit the changes in data access patterns.
- If there are multiple index files for the same data file, force the system to use the right index for retrieval.
- Revise memory buffer management as number of transactions increase.
- Revise how records are blocked within a file.
- Place records that are retrieved together closer to one another in storage.
- Change storage medium if necessary.

Managing Growth

Growth in the database system is inevitable in an organization. The database environment is dynamic. You must anticipate more users and more database transactions with the passage of time. Databases grow in size. That is, you will need more storage space. Databases tend to grow in complexity. As changes in business conditions occur, existing data structures will be revised and newer data structures will be added to your organization's database. Databases tend to get distributed. As users disperse to newer locations, data must be provided to them at the additional locations.

Managing database growth occupies the continued attention of the project team. All IT professionals responsible for the upkeep of the database environment share the responsibility to plan and manage the growth. Here are a few tips to manage growth of the database system:

- Be prepared for growth. Immediately after the requirements definition phase, start planning for growth.
- Establish a small team of IT professionals charged with the responsibility of monitoring growth and making recommendations.
- Archival of old data frees up storage space for current data. Have a regular schedule for archival of infrequently used data.
- Reorganize disk files at regular intervals to consolidate fragmented empty spaces.
- Wherever feasible, use data compression techniques.
- Adopt parallel processing hardware and software techniques, especially if your database environment has to support a large volume of ad hoc queries.

- At regular intervals of about 12 months, review your processor capacity to handle the increased number of transactions. Upgrade if necessary.

CHAPTER SUMMARY

- Because of the importance of database systems, organizations must build them with a lot of care and attention.
- Right at the outset, adopt a data-oriented approach and establish a framework for the development.
- The database development life cycle (DDLC) consists of the following phases or steps, not necessarily in sequential order: planning, feasibility study, requirements definition, design, implementation and deployment, maintenance and growth.
- Database practitioners from IT and user representatives have distinct roles in database development.
- Proper definition of requirements is critical because it drives the design and the other phases.
- The design phase consists of producing the logical design (data structures, relationships, etc.) and physical design (files, records, etc.).
- Database administrators play key roles in database implementation, deployment, and maintenance.

REVIEW QUESTIONS

1. What do you understand about a data-oriented approach for database development?
2. Name the phases or steps of DDLC. Describe very briefly the types of tasks in two of these phases.
3. Who normally does the planning for the database system in an organization? What are the planning responsibilities?
4. What are the three general areas for feasibility study? Why are they important?
5. Describe the types of costs associated with a database system.
6. List any six major tasks in the requirements definition phase. Describe any one of these tasks.
7. What is the distinction between logical and physical design? Why are these two levels of design necessary?
8. What are some of the tasks for populating the new database with initial data? Describe briefly.
9. "DDLC does not end with the initial deployment of the database." Do you agree? Explain.
10. What is tuning the database for performance? Why is this necessary?

EXERCISES

1. Indicate whether true or false:
   A. The DDLC approach enforces a systematic method for database development.
   B. In every database project, the steps of DDLC must be followed strictly in sequence.
   C. The company's business plan indicates future information requirements.
   D. The feasibility study phase defines the requirements.
   E. JAD is an appropriate methodology for a small company with a few users.
   F. The logical design deals with how data are actually stored in the database.
   G. The conceptual schema is an aggregation of the external schema.
   H. The data dictionary is built after the database system is deployed.
   I. Fine-tuning the database is an exclusive responsibility of the database administrator.
   J. Disk space management is an ongoing administrative task.
2. You have been hired as the database administrator for a medical center that is implementing a new database system from scratch. Consider each phase of the development life cycle and describe your expected role in each phase.
3. A company manufacturing parts for appliances proposes to move from a mainframe hierarchical database system to a client/server relational database system. As the project manager, examine briefly your main challenges in this major transition.
4. If you are the chief user representative for the database development effort in a large global organization, what contributions are you expected to make in the requirements definition phase?
5. You are the senior database designer in the database project for a local bank. What are your responsibilities in the design phase?