Abstract - The trend of producing databases which do not satisfy organizational requirements can be attributed to the fact that the database designer is required to simultaneously consider many design constraints and does not have a comprehensive methodology to follow. A new decomposition of the logical database design process in which each level satisfies a single design objective is developed. In this decomposition, the requirements process and logical database design process are integrated to ensure that organizational requirements are satisfied. A structured logical database design methodology called LDDM that is based on this decomposition has been developed. LDDM consists of six design steps: Requirements Step, Entity Step, Relationship Step, Entity Structure Step, Refinement Step and DBMS Accommodation Step. LDDM including its constituent steps and techniques is presented. The utilization of LDDM will result in increased designer productivity, better documentation and a high quality logical database.

I. Introduction

Designing a database is a difficult, complex and time-consuming process. Unfortunately, inadequate databases result because they cannot satisfy the present or future organizational requirements. This can be attributed to the lack of a usable logical database design methodology that is integrated with the requirements process. Thus, a formal structure is needed to be the foundation for the logical database design methodology.

A structured logical database design methodology called LDDM which will produce "good" logical databases has been developed. Before this methodology can be described, a "good" logical database will be defined and various approaches to structuring the logical database design process will be presented.

The primary goal of the logical database design process is to produce a logical database that satisfies organizational requirements. There are four aspects of satisfying organizational requirements. First, the organizational information requirements must be satisfied. Information requirements refer to the availability of all information needed by the organization. Second, the organizational processing
requirements must be met. This includes adequately satisfying the time constraints of processing. In contrast to information requirements, processing requirements are concerned with the efficient access of data. Another characteristic of a good database design is that it is represented in a simple and easily comprehensible form. This form should be usable by many different individuals from diverse levels and functions within the organization. Finally, the logical database should be flexible enough to accommodate future requirements by being easily adapted. The first three aspects can be verified. However, it is difficult or perhaps impossible to demonstrate compliance with the last aspect. A flexible database design has a high degree of data independence. Therefore, an analogous goal of producing an adaptable and flexible design is to produce logical databases with a high degree of data independence especially logical data independence.

II. Approaches to Logical Database Design

The purpose of the logical database design process is to transform real world requirements into a "good" logical database. There are a number of alternatives for accomplishing this transformation. One extreme is to map the real-world requirements represented in a single statement (in a single step) into a logical database. The other extreme is to have numerous statements and an extremely large number of steps. Experience has shown that neither of these is a practical solution. A realistic solution is to divide this transformation into a series of step-wise mappings, each of which is more easily understood and more easily accomplished by the designer. Thus, the logical database design process can be viewed as two mappings. The first is mapping real-world requirements into an information structure and the second is transforming the information structure into a logical database. This view of the logical database process is shown in Figure 1. The information structure (user's view) is problem oriented and is independent of implementation. The process of deriving the information structure from real-world requirements is usually ignored in performing database design. The logical database (application program's view) is a type of data structure and is dependent on data representation and the database management system (DBMS).

The requirements document must be in a usable form to serve as input to the logical database design process in order for the ensuing logical database to satisfy organizational requirements. Traditionally, the logical database design process begins with an information structure as input. Whether this information structure is a global representation or that of one individual is not of concern to the designer. If the database design process is not based on a global information structure, then the ensuing database design cannot satisfy organizational goals and requirements. To