FORMALISMS OF OBJECT-ORIENTED DATABASE

A Thesis
Presented to the
Faculty of the Computer Science Graduate Program
College of Computer Studies
De La Salle University

In Partial Fulfillment of the
Requirements for the Degree
Master Of Science in Computer Science

by

MARY ANN G. NAVALTA

May, 1990

ABSTRACT

The database can be viewed as a collection of tables (relational model), as a collection of record types and sets (network model) or as a forest of tree structures (hierarchical model). The logical model of the database is the correct level for database users to focus on. Considering the performance of these databases is highly dependent upon the database design, many of them have been over designed to ensure adequate performance. The performance of the database depends primarily on the efficiency of the data structures used to represent the data in the database and on how efficiently the system is able to operate on these data structure.

The database models mentioned above came into existence because of a need for a tool to make data processing professionals and end users of their services more productive. To date, most of the research work is focus on the other database model which is object-oriented model.

Object-oriented modelling represents a successful unifying paradigm in various areas of computing specifically databases. In an object-oriented environment the world is viewed as a collection of active objects. Each object has a set of variables which determine the state of the object. Each object can understand a set of predefined messages. Objects communicate with each other by sending messages. Sending a message to inquire about the state of an object is like invoking a function and sending a message to change the state of an object is like an update operation. Updates can be

generated because sending a message to an object can cause more messages to be sent to other objects. Similar objects are defined by classes. Classes are organized as class/subclass hierarchies. Subclasses inherit the messages of their superclasses. This inheritance and the class/subclass relationships support generalization abstraction.

This research is directed towards formulating the formal formalisms of object-oriented database. This study shows that object-oriented increases the productivity of the design systems by providing modelling facilities which closely mirror the proofs that object modelling improves the structure of the data in promoting the availability of the data. Thus, the study substantiates its significance in providing a better set of constructs for modelling database. Moreover, the study proves that object paradigm helps the semantic gap between databases and applications.

TABLE OF CONTENTS

LIST OF FIGURES

LISTINGS