CHAPTER 1

INTRODUCTION

Exploration is a process of discovery. In the database exploration process, an analyst executes a sequence of transformations over a collection of data structures to discover useful information, or knowledge. The analyst may or may not have a preconceived exploration plan, and often uses the result of one data transformation (not necessarily the previous transformation) to contribute to the formulation of a follow-up data transformation. The database exploration process is richly varied and complex, requiring a high degree of technical and perceptual skill on the part of the data analyst. Understanding the database exploration process is therefore a key challenge towards building supportive software environments, and this research addresses several fundamental issues.

1.1 Problem Statement and Motivation

Knowledge discovery in databases (KDD), or “data mining”, has become a popular technological phrase of the 1990s. Not only may a large database be summarized once its internal knowledge is discovered, its embedded knowledge can be exploited by a data analyst or decision maker (who can be, for example, a marketing manager, scientific researcher, policy planner, or financial analyst).

The problems associated with KDD are numerous. The root of the problem is that database size can approach the terabyte range, and data dimensionality can approach thousands of distinct attributes. As database size and complexity increases the embedded information becomes more difficult to extract. Another fundamental issue is that the analyst often has no idea of what he is looking for, and must actively explore the data by manually orchestrating the application of data transformations. Associated issues range from data storage and access to knowledge extraction algorithms to data presentation to user interface configuration.

Adequate software tools are required to support the data analyst in extracting knowledge from databases. Current software tools - database management systems, data analysis systems and data visualization systems - address specific portions of the database exploration problem. Database management systems excel at formal data modeling, integrity, concurrency and data selection. Data analysis systems determine statistical data properties, perform algorithmic knowledge extraction (clusters, rules and patterns in the data), and display data graphically. Data visualization systems, a subclass of data analysis systems, use advanced rendering techniques to display massive amounts of multidimensional data graphically, thereby harnessing the human perceptual capabilities to discriminate patterns in the data where algorithmic approaches fail. A key challenge for software system design is to support all three of these database exploration components, either in a single system or in an integrated environment.

Providing an integrated database exploration environment poses many additional challenges for the software architecture. Integrating the component systems requires translating between data models and their data manipulation styles. Database systems primarily use a relational data model for transactional data processing. Data analysis systems primarily use multidimensional arrays and matrix operations for analytic

processing. Data visualization systems use a multitude of representations for batch processing of data through dataflow networks. Emphasizing *user-centric* aspects (data manipulation and presentation paradigms) of the integration places burdens on the *data-centric* (i.e., systemic) aspects when translating interactions over visual data representations to data manipulation operations over the appropriate data objects.

But how is a database explored? From the above, data-centric description of the problem, the ability to explore databases for hidden knowledge requires *data selection* techniques to isolate data and reduce complexity, *data analysis* techniques to determine quantitative data properties and *data visualization* techniques to display data volumes in some intuitive manner and to guide the exploration. Interaction with data at multiple levels of abstraction and resolution must be supported in the integration of these three components.

We know many of the techniques that are employed, but we know little about the database exploration *process*, the manner in which the data transformations are applied over the data to extract knowledge, their sequencing and relationships. The transition from the database to knowledge is a data refinement process, as “nuggets” of useful information are uncovered. The database is subjected to numerous data transformations to produce entities that are reduced in size, but contain the essence of the database: a summary report, a mathematical model, a visualization, etc., that are of great interest to the data analyst. The refinement process takes a large, unknown, and unorganized database (with respect to knowledge structures, since databases usually have well-defined data structures), and produces smaller, known, and highly organized knowledge structures.

The refinement process does not follow a straight and narrow course, however. Of the numerous possible data transformations, some may refine data usefully, while others may deviate from a particular path of refinement. Furthermore, a particular refinement

path could become useless as it is further extended. Thus, there are numerous, interconnected “threads of inquiry” into the database that comprise an exploration session.

We have little knowledge of such intricacies. Empirical studies have identified common tasks that are undertaken during scientific data analysis, but little has been done to model the dynamic nature of database exploration. Understanding the database exploration process therefore requires the process to be modeled and quantified.

1.2 Thesis Goals

There are two primary goals of this research, the first is data-centric and the second is user-centric.

Goal 1. Analyze systems issues related to supporting static database exploration tasks.

(a) Analyze current systems used for database exploration with respect to the data interaction capabilities they provide the user.

(b) Based on the analysis in (a), design and implement a software architecture that maps between the data models and interaction styles of a database system and a visualization system, realizes a simple database exploration model and allows the tracking and storage of user/data interactions.

This data-centric goal actually takes a user-centric approach to generate requirements for the software architecture. Subgoal 1(a) establishes a baseline description of user/data interaction capabilities available in commercial software environments used from database exploration. Subgoal 1(b) applies this user-centric analysis towards the realization of a software architecture that permits the study of the

system integration issues. It also establishes a context for the research embodied in the second primary goal.

Goal 2. Develop a dynamic database exploration process model.

(a) Design a general, data-independent database exploration model that accounts for the dynamic nature of exploration interactions.

(b) Develop a number of descriptive measures for the general database exploration model.

(c) Apply the general model and measures towards database and visualization domains.

(d) Show how the model captures various database exploration scenarios.

Subgoal 2(a) defines all of the fundamental components of a database exploration model, devoid of any data domain semantics, thereby capturing only the dynamic aspects of the process. Subgoal 2(b) enables the model to express useful information about the database exploration process. Subgoal 2(c) applies the model to database and visualization data domains, making the general model a concrete model. Subgoal 2(d) is a validation of the model.

1.3 Thesis Contributions

This thesis approaches database exploration in a unique manner by addressing both data-centric (systems) issues and user-centric (interaction) issues with equal weighting. We feel that this is an essential, more complete approach to the database exploration problem, as database exploration is really a synthesis of several other domains, including database, visualization, artificial intelligence, statistics and user interfaces.

The data-centric contribution is the systems integration model, a task-based solution to the impedance mismatch problem inherent in applying database management systems to different data domains. It introduces and validates a new integration model that is a departure from previously proposed models. Its design revolves around supporting fundamental database exploration tasks derived from the literature. The most notable aspect of the integration model is that it elevates the importance of communicating the visual interactions from the user interface of the visualization system to the DBMS, and not only matching the data structure models for visualization purposes. It proposes a framework (the VisualizationView hierarchy) to model this communication, and an implementation, *Exbase*, to realize this communication. The systems integration model also shows that database management capabilities can indeed be built into a visualization system.

The user-centric contribution is a generalized data exploration (GDE) model, composed of *data entities* and *data derivations*, that defines the database exploration process in a general fashion, i.e., irrespective of the underlying data model and data manipulation domain. The GDE model is also general in that it accommodates two views of the data exploration process, at the individual data entity level, and at the set-of-data-entity level. Thus, it can accommodate multiple data entities being transformed in a single transformation.

The GDE model defines four types of metadata that describe data objects: *identification*, *structural*, *process* and *knowledge*. Its emphasis on process metadata is unique, as the model describes a process. The GDE model's two fundamental representations of data exploration sessions, *data derivation sequences* and *data derivation graphs*, show different aspects of the session. The sequence representation emphasizes the sequential, linear nature of the process, while the graph emphasizes the

derivational nature of the process. Both are used in analyzing sessions and the process in general. Several variants of data derivations, sequences and graphs are developed that have different degrees of expressiveness.

An important and unique contribution of the GDE model is the definition of *metrics* (process metadata) over the model that can be used to describe the data exploration process in a general fashion. The metrics categorize data derivation graph vertices and paths based on their *derivational structure*. A calculus of interactions is developed on the model (using both sequence and graph representations) to describe the process in even greater detail. Along the way to defining metrics, the concepts of *data exploration space*, *scope*, and *continuity* are established.

The final contribution of this thesis is in the application of the GDE model and its metrics with the concrete data exploration scenario that initiated the systems integration model. In accomplishing this, we developed several canonical *interaction patterns* that are prevalent in both database and visualization exploration domains.

1.4 Thesis Overview

Chapter 2 defines database exploration and reviews applicable previous research. It is lengthy but necessary to highlight the many facets of database exploration. Chapter 3 describes database exploration systems issues and how the research prototype, *Exbase*, is a representative solution to those issues. Based on open questions posed in Chapter 3, Chapter 4 defines a Generalized Data Exploration (GDE) model. Chapter 5 refines the GDE model by creating metrics to describe database exploration sessions, developing a calculus of interactions to quantify the changes occurring in a session, and describing the concept of *continuity* with respect to database exploration. Chapter 6 applies the GDE model to database and visualization domains and develops several important interaction

patterns expressed by the GDE model. Chapter 7 summarizes the research and describes future work.