Chapter 7
Fuzzy Querying Capability at Core of a RDBMS

Ana Aguilera
Universidad de Carabobo, Venezuela

José Tomás Cadenas
Universidad Simón Bolívar, Venezuela

Leonid Tineo
Universidad Simón Bolívar, Venezuela

ABSTRACT

This chapter is focused on incorporating the fuzzy capabilities into a relational database management system (RDBMS) of open source. The fuzzy capabilities include connectors, modifiers, comparators, quantifiers, and queries. The extensions consider a more flexible DDL and DML language. The aim is to show the design and implementation details in the RDBMS PostgreSQL. For this, a fuzzy query processor and fuzzy access mechanism have been designed and implemented. The physical fuzzy relational operators have been defined and implemented. The flow of a fuzzy query through the different modules (parser, planner, optimizer, and executor) has been shown. Some experimental results have been included to demonstrate the performance of the proposal solution. These results show that the extensions have not decreased the performance of the RDBMS.

INTRODUCTION

The language used by human beings is imprecise, but for the computer the opposite stands true. Scientists are continually investigating how to create an intermediate point to enhance the communication between these different worlds. Moreover, the capacity to store large quantities of data is growing in an exponen-
tially form, we now dispose of technology that is normally interconnected to support many terabytes or petabytes of data, consequently we need systems that are able to efficiently manipulate this data.

The problem resides in the fact that human reasoning is very different from the logic involved in software. In order to narrow this gap, there exists a branch in the scientific community which studies different techniques to model the human mind. This branch is known as soft computing (Zadeh, 1994). The aim of this work is to describe, in a general form, how to render the constraints of the queries on a database more flexible, therefore making a Database Management System (RDBMS) flexible without affecting in a considerable way its performance to manipulate large quantities of data.

Natural language is imprecise, vague and inexact; for example, when we say *Juan is a young person* or *Milton is very tall*, we are expressing imprecise terms like *young* and *very tall*; how does the computer interpret these terms? Normally, we have to employ precise terms: a person is young if she/he is 30 years old or less; but, what about the person who is 31 years old? One day I am young (I was 30 years old), and the day after my birthday, Am I no longer young?

Furthermore, Juan is 50 years old and he considers himself to be young. That is true for him but false for others. If Milton is 1.80 meters, do you think that he is very tall? A lot of persons will say no, but if Milton is being selected to ride horses as a jockey? Then he is very tall; truthfulness or falseness is relative, and depends on many factors. People have different points of view (preferences) and the role of context is also important.

In a computer system, objects are clearly divided into white and black, but in the real world there is an infinite grey area between the two and most things normally fall in the grey area. Morin (1999) states that “Reality is not easily legible. Ideas and theories are not a reflection of reality they are translations, and sometimes mistranslations. Our reality is nothing more than our idea of reality” (p. 44). Morin, a French philosopher and sociologist, recognizes ambiguity and uncertainty as the hallmark of science and of human experience.

Morin’s approach, complex thought, is in harmony with a culture of uncertainty. Morin (1999) asserts “But in life, unlike crossword puzzles, there are boxes without definitions, boxes with false definitions, and no neat framework to define the limits” (p. 45). How can we manipulate these boxes into the computers?

An example is the evaluation of young ladies for top models. There are many requirements to be fulfilled and nobody satisfies all of them. However, *Lolita* was disqualified because she is 1.79 meters tall and there exists a clear (*crisp*) condition stipulating a minimum height of 1.80 meters. But what about the other prerequisites met? What happen if *Lolita* was the best candidate based on the rest of them? This system may be very unfair, due to the rigidity of precise conditions.

We have proposed the enhancement of an RDBMS to support flexible queries and help users to define preferences depending on the context by using fuzzy conditions in their SQL queries.

The contribution of this chapter is to describe the design and implementation of flexible queries (using fuzzy logic) in a Relational Database Management System (RDBMS) without affecting its performance to manipulate large quantities of data in a considerable form. The approach is Tightly Coupling, with modification in the source code on the query processor engine in order to process fuzzy queries.

**BACKGROUND**

Traditional RDBMS suffer from rigidity: data are considered to be perfectly known and query languages do not allow natural expression of user preferences. In order to provide flexible queries over databases,