A Demonstration of FlexPref: Extensible Preference Evaluation Inside the DBMS Engine

Justin J. Levandoski  Mohamed F. Mokbel  Mohamed E. Khalefa  Venkateshwar R. Korukanti
University of Minnesota Department of Computer Science and Engineering

Need for Preference Functionality in the DBMS

DBMS Implementation Approaches

Preferences Methods

Many Multi-Objective Preference Methods

Quick Exercise
1. Go to Google Scholar
2. Search for papers on preference evaluation methods
3. How many results do you get back?

The list goes on and on...

Extensible Preference Evaluation Inside the DBMS Engine

Preference Methods

Existing Approaches

The Layered Approach

DBMS is a "black box" for the preference method

Severe performance limitations:
1. Evaluate SQL query
2. Evaluate preference function

Simplicity: easy to implement

Limited Efficiency: cannot interact with DBMS internals; no query optimization

The Built-In Approach

Custom implementation for each method.
Several thousands of lines of code for each method

Efficient: methods tightly coupled with DBMS

Infesible: cannot provide custom implementation for every preference method

Writing Queries

Severe performance limitations:
1. Evaluate SQL query
2. Evaluate preference function

Efficiency of the built-in approach: methods tightly coupled with DBMS

FlexPref Architecture

Modify query processor only once

"Plug-in" preference method semantics

FlexPref in Action

Define Preference Method

We demonstrate how to implement the Top-K Domination preference method using FlexPref functions and macros

Functions implemented in file outside DBMS codebase and compiled into FlexPref

Once compiled, preference method is available for use in defining preference profile in web-based application

Querying with FlexPref

Users press "find restaurant/hotel" button.

Answers displayed on Google Maps.

Answers generated by FlexPref executing in PostgreSQL using chosen preference method

Users can view SQL query extended with preference syntax that was executed in underlying PostgreSQL database

Users can also issue ad-hoc FlexPref queries, and view query plans, using graphical client connected to PostgreSQL

FlexPref offers generic query processing support for the following operators, each operator written in generic fashion with reference to the specific macros and functions for specific preference method semantics

Environment

Define Preference Method

We demonstrate how to implement the Top-K Domination preference method using FlexPref functions and macros

Functions implemented in file outside DBMS codebase and compiled into FlexPref

Once compiled, preference method is available for use in defining preference profile in web-based application

Querying with FlexPref

Users press "find restaurant/hotel" button.

Answers displayed on Google Maps.

Answers generated by FlexPref executing in PostgreSQL using chosen preference method

Users can view SQL query extended with preference syntax that was executed in underlying PostgreSQL database

Users can also issue ad-hoc FlexPref queries, and view query plans, using graphical client connected to PostgreSQL

FlexPref offers generic query processing support for the following operators, each operator written in generic fashion with reference to the specific macros and functions for specific preference method semantics

Web-based restaurant and hotel finder application implemented in Google Maps with real Indianapolis restaurant/hotel data

Data stored in underlying PostgreSQL database

FlexPref implemented inside query processor of PostgreSQL