



# FlexPref: A Framework for Extensible Preference Evaluation in Database Systems

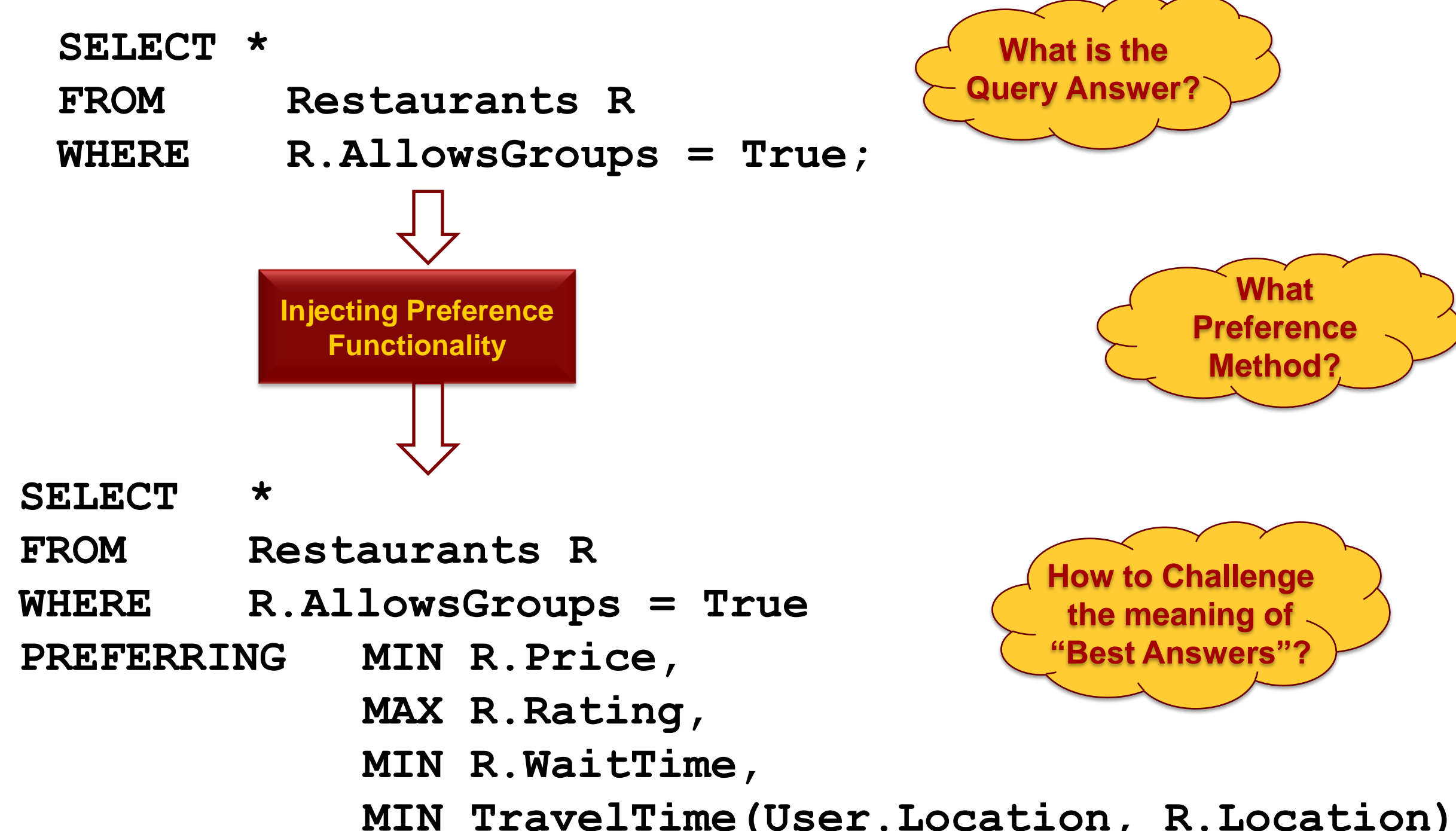
Justin J. Levandoski

Mohamed F. Mokbel

Mohamed E. Khalefa

University of Minnesota Department of Computer Science and Engineering

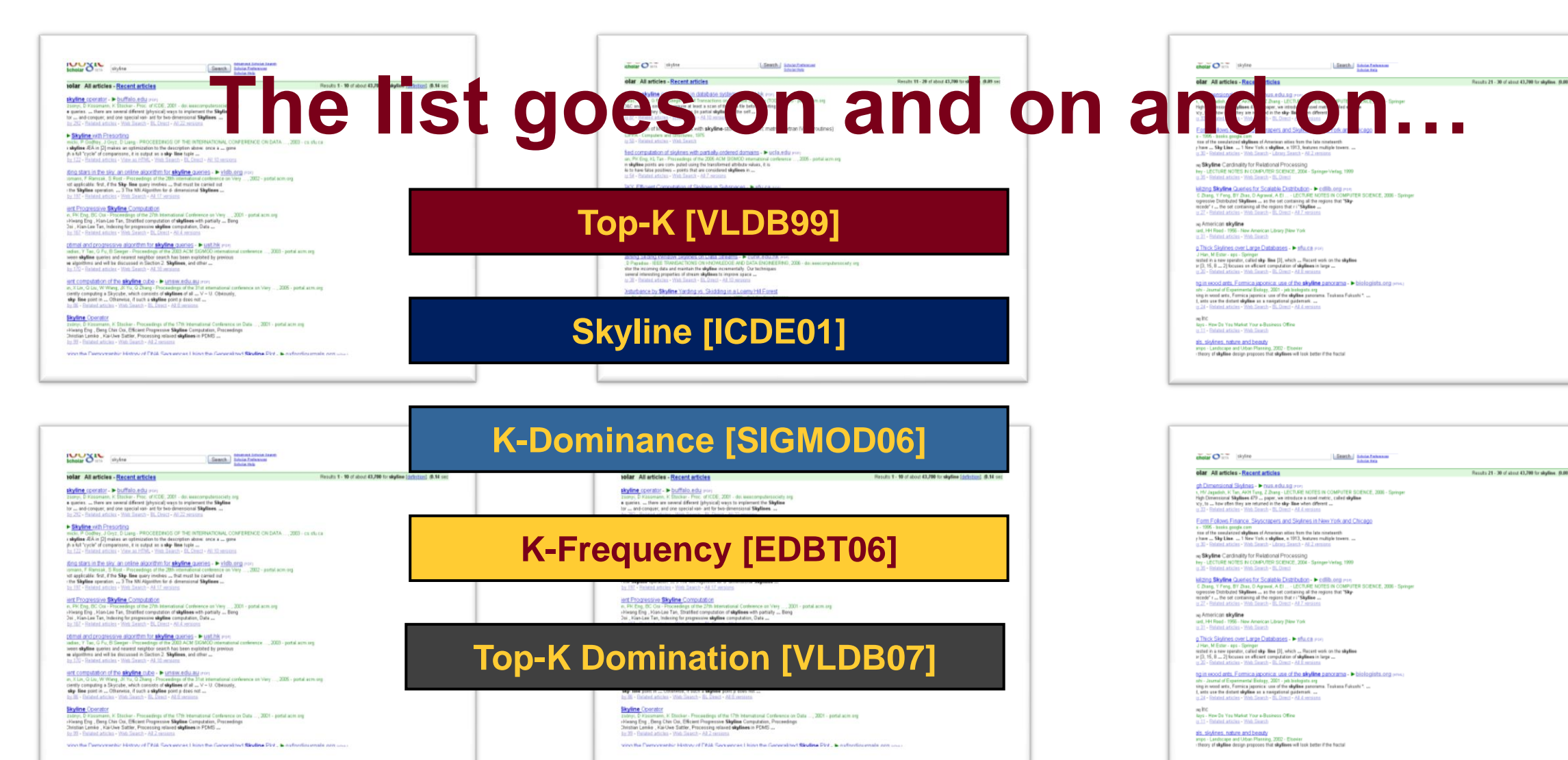
## Need for Preference Functionality Inside the DBMS



## Many Multi-Objective Preference Evaluation Methods

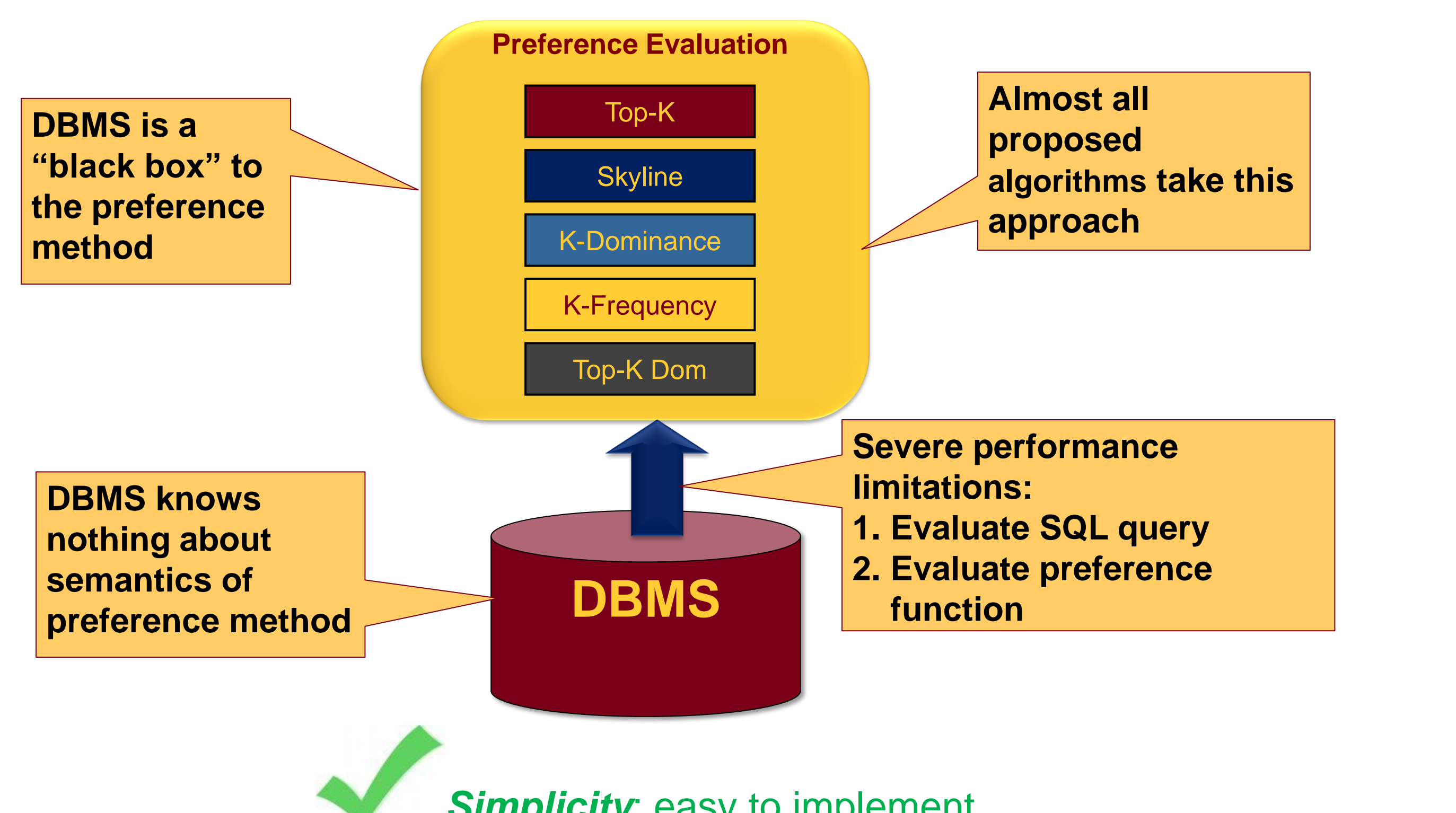
### Quick Exercise

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



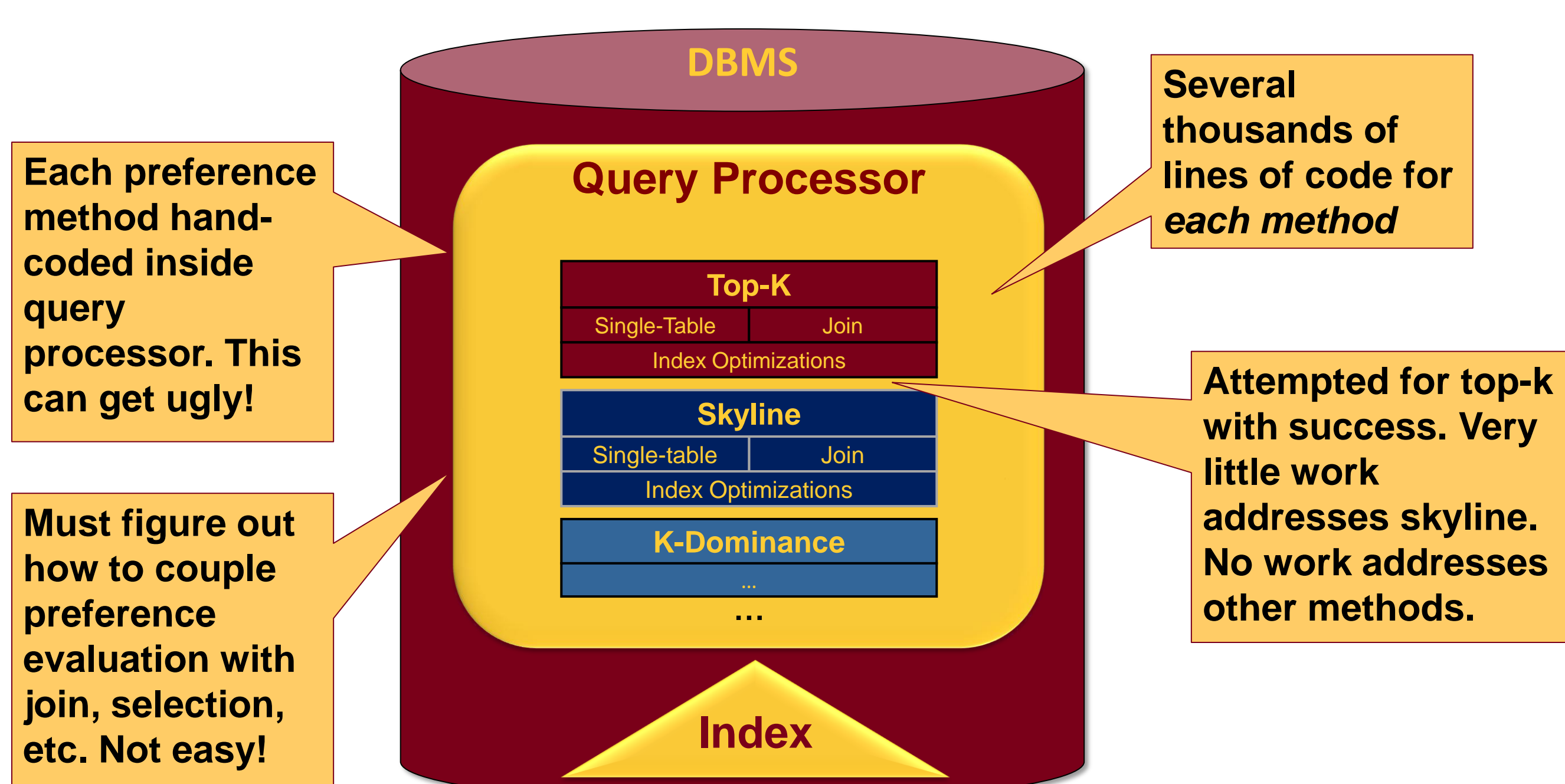
## Implementing Preference Functions in a DBMS: Existing Approaches

### The Layered (On-Top) Approach



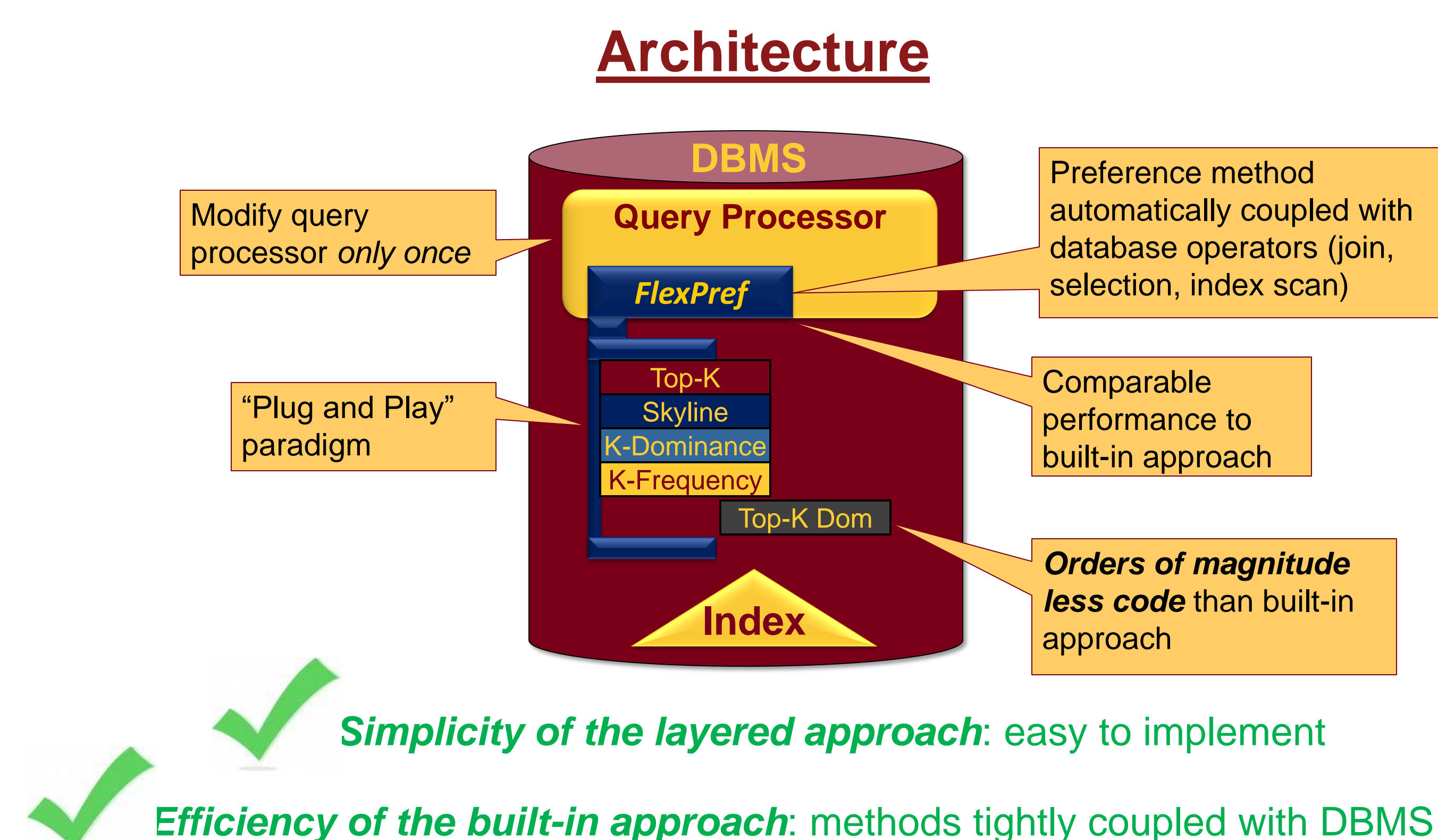
**Limited Efficiency:** cannot interact with DBMS internals, thus no query optimization

### The Built-In Approach



**Infeasible:** cannot provide custom implementation for every preference method

## The FlexPref Approach



### Writing Queries

```

SELECT *
FROM Restaurants R
WHERE [Where clause]
PREFERRING [Attribute List]
USING MyPref Objectives [Preference Objectives]
    
```

```

SELECT * FROM Restaurants
WHERE ...
PREFERRING Price P, Distance D, Rating R
USING Skyline OBJECTIVES MIN P, D, MAX R
    
```

```

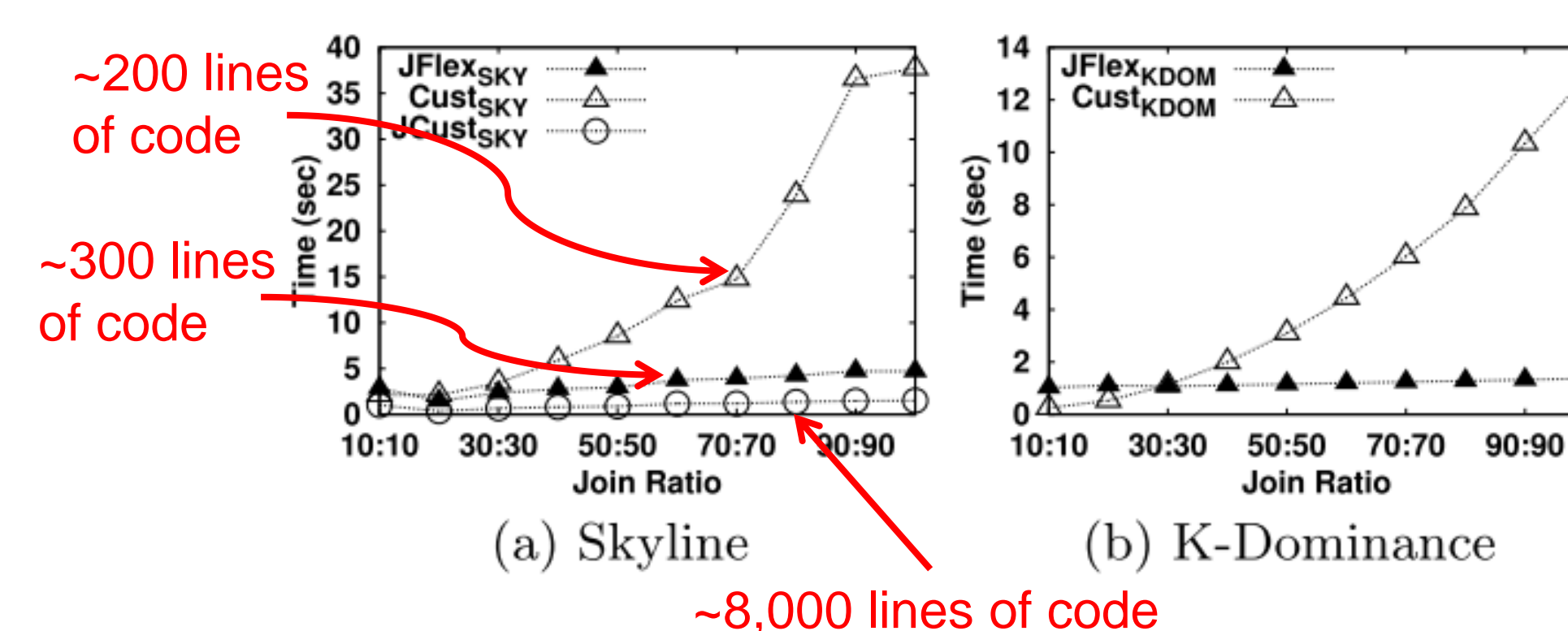
SELECT * FROM Restaurants
WHERE ...
PREFERRING Price P, Distance D, Rating R
USING TopKDom WITH K=5 OBJECTIVES MIN P,D, MAX R
    
```

```

SELECT * FROM Restaurants
WHERE ...
PREFERRING Price P, Distance D, Rating R
USING TopK WITH K=5 OBJECTIVES MIN Func(P,D,R)
    
```

### Experimental Analysis

Preference evaluation with binary join  
FlexPref Join v. Built-in Selection v. Built-in Join



### Preference Implementation

1. Define *two macros* and *three functions* in separate "MyPref.c" file outside DBMS/FlexPref

```

#define DefaultScore 0
Default score assigned to each object
#define IsTransitive 1
Whether preference function is transitive or not
PairwiseCompare(Object P, Object Q)
INPUT: Two objects P and Q
ACTION: Update the score of P
RETURN: 1 if Q can never be a preferred object
        -1 if P can never be a preferred object
        0 otherwise
IsPreferredObject(Object P, PreferenceSet S)
INPUT: A data object P and a set of preferred objects S
RETURN: True if P is a preferred object and can be added to S
        False otherwise
AddPreferredToSet(Object P, PreferenceSet S)
INPUT: A data object P and a set of preferred objects S
ACTION: Add P to S and remove or rearrange objects from S
    
```

2. Compile into FlexPref using command: DefinePreference MyPref with MyPref.c

### Query Processing

FlexPref offers generic query processing support for the following operators, written in terms of three functions and two macros

