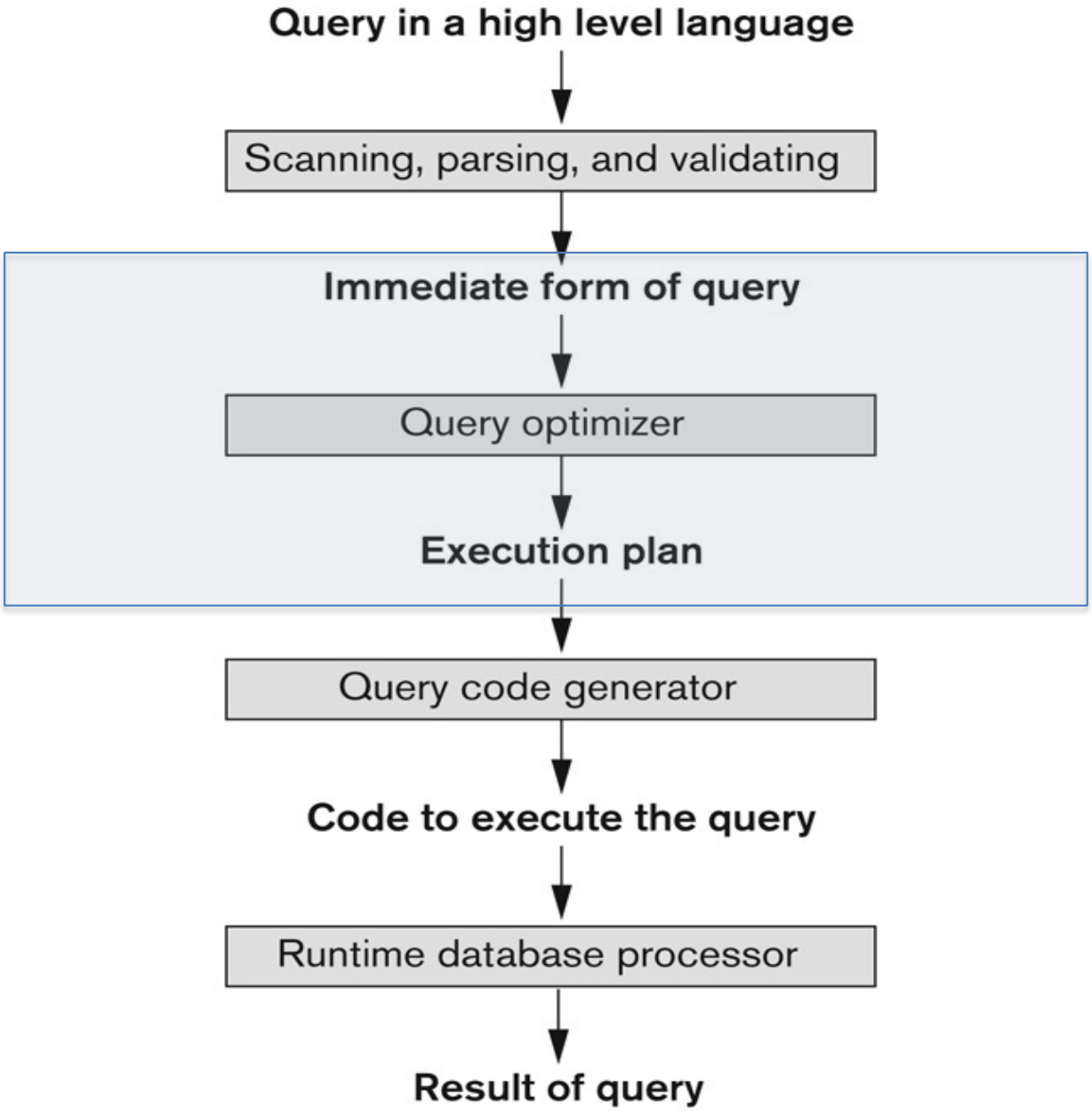
INFS7901
Database Principles

Relational Algebra and Query Processing and Optimization

Rocky Chen

Introduction to Query Processing



Conceptual Procedural Evaluation Strategy

1. Compute the cross-product of *relation-list*.
2. Discard resulting tuples if they fail *qualifications*.
3. Delete attributes that are not in *target-list*.
4. If DISTINCT is specified, eliminate duplicate rows.

Example of Conceptual Procedural Evaluation

SELECT Name

FROM MovieStar M, StarsIn S

WHERE S.StarID = M.StarID AND MovieID = 276

join selection

MovieStar **X** StarsIn

(StarID)	Name	Gender	MovieID	(StarID)	Character
1273	Nathalie Portman	Female	272	1269	Leigh Anne Touhy
1273	Nathalie Portman	Female	273	1270	Mary
1273	Nathalie Portman	Female	274	1271	King George VI
1273	Nathalie Portman	Female	276	1273	Nina Sayers
…	…	…	…	…	…

Query Optimization

- This strategy is probably the least efficient way to compute a query!
- A query typically has many possible execution strategies. The process of choosing a suitable one for processing a query is known as **query optimization**

The term optimization is actually inaccurate because in some cases the chosen execution plan is not the optimal (best) strategy. It is just a reasonably efficient strategy for executing the query.

To perform query optimization, we must first translate SQL queries to relational algebra.

Basic Relational Algebra Operations

Advanced Relational Algebra Operations

Implementation of SELECT and Join Operations

From Queries to Optimization

Relational Query Languages

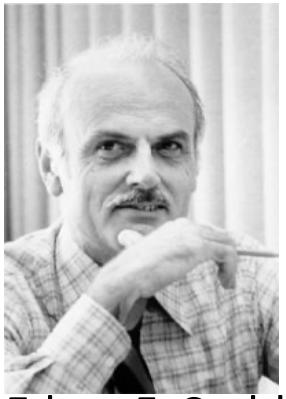
- Allow data manipulation and retrieval from a DB.

- Query Languages != Programming Languages
  - QLs provide easy access to large datasets.
  - Users do not need to know how to navigate through complicated data structures.

- Relational model supports simple, powerful QLs:
  - Strong formal foundation based on logic
  - Allows for much optimization via *query optimizer*

Relational Algebra: A formal relational query language that forms the mathematical foundations for SQL.

Relational Algebra (RA)


Edgar F. Codd

- All operators take one or two relations as inputs and give. a new relation as a result
- operations:
  - *Selection* (σ): Selects a subset of rows from relation.
  - *Projection* (π): Deletes unwanted columns from relation.
  - *Set operations*(∪, ∩ , -): Use Union, intersection, and set difference to select a subset of rows from two relations.
  - *Cartesian-product* (x): Combines two relations.
  - *Rename* (ρ): Assigns a (another) name to a relation or a column.
  - *Join* (⋈):Combines two relations with some constraints.
  - *Assignment*(←): Assigns the result of an expression to a temporary relation.
  - *Division* (/): Allows for expressing queries that include a "**for all**" or "**for every**" phrase.

Example Movies Database

Movie(MovieID, Title, Year)

StarsIn(MovieID, StarID, Character)

MovieStar(StarID, Name, Gender)

Example Instances

Movie:

MovieID	Title	Year
1	Star Wars	1977
2	Gone with the Wind	1939
3	The Wizard of Oz	1939
4	Indiana Jones and the Raiders of the Lost Ark	1981

StarsIn:

MovieID	StarID	Character
1	1	Han Solo
4	1	Indiana Jones
2	2	Scarlett O'Hara
3	3	Dorothy Gale

MovieStar:

StarID	Name	Gender
1	Harrison Ford	Male
2	Vivian Leigh	Female
3	Judy Garland	Female

Selection (σ (sigma))

- Notation: $\sigma_{\mathbf{p}}(r)$
- Defined as:

$$\sigma_p(r) = \{t \mid t \in r \textbf{ and } p(t)\}$$

Set of tuples of r satisfying p

- **p** is called the selection predicate defining the selection condition in propositional logic.
- **The Result:** Selects rows that satisfy selection condition
- **Schema:** Identical to schema of input relation.

Selection Example

Movie:

MovieID	Title	Year
1	Star Wars	1977
2	Gone with the Wind	1939
3	The Wizard of Oz	1939
4	Indiana Jones and the Raiders of the Lost Ark	1981

$\sigma_{year > 1940}(Movie)$

MovieID	Title	Year
1	Star Wars	1977
4	Indiana Jones and the Raiders of the Lost Ark	1981

Selection Example #2

Find all male stars

Movie (MovieID, Title, Year)
StarsIn (MovieID, StarID, Role)
MovieStar (StarID, Name, Gender)

$$\sigma_{\text{Gender} = \text{'male'}}(\text{MovieStar})$$

StarID	Name	Gender
1	Harrison Ford	Male

Projection (π (pi))

- Notation: $\boldsymbol{\pi_{A1, A2, \ldots, Ak}(r)}$
  where *A1, …,Ak* are attributes (the projection list) and *r* is a relation.

- **The result:** Deletes attributes that are not in projection list.
- **Schema:** Exactly the fields in the projection list, with the same names they had.

- Duplicate rows removed from result (relations are sets)

Result relation can be the input for another relational algebra operation!

Projection Examples

Movie

MovieID	Title	Year
1	Star Wars	1977
2	Gone with the Wind	1939
3	The Wizard of Oz	1939
4	Indiana Jones and the Raiders of the Lost Ark	1981

$\pi_{\text{Title, Year}}$ (Movie)

Title	Year
Star Wars	1977
Gone with the Wind	1939
The Wizard of Oz	1939
Indiana Jones and the Raiders of the Lost Ark	1981

π_{Year}(Movie)

Year
1977
1939
1981

What is $\pi_{\text{Title,Year}}(\sigma_{\text{year} > 1940}(\text{Movie}))$?

Title	Year
Star Wars	1977
Indiana Jones and the Raiders of the Lost Ark	1981

Projection Example #2

- Find the IDs of actors who have starred in movies.

Movie (MovieID, Title, Year)
StarsIn (MovieID, StarID, Role)
MovieStar (StarID, Name, Gender)

$\pi_{\text{StarID}}(\text{StarsIn})$

StarID
1
2
3

Clicker Projection Example

- Suppose relation R(A,B,C) has the tuples:

A	B	C
1	2	3
4	2	3
4	5	6
2	5	3
1	2	6

- Compute the projection $\pi_{C,B}(R)$, and identify one of its tuples from the list below.

A. (2,3)
B. (4,2,3)
C. (6,4)
D. (6,5)
E. None of the above

Clicker Projection Example

- Suppose relation R(A,B,C) has the tuples:

A	B	C
1	2	3
4	2	3
4	5	6
2	5	3
1	2	6

- Compute the projection $\pi_{C,B}(R)$, and identify one of its tuples from the list below.

A. (2,3) Wrong order
B. (4,2,3) Not projected
C. (6,4) Wrong attributes
D. (6,5) right
E. None of the above

Selection and Projection Example

- Find the ids of movies made prior to 1950

MovieID	Title	Year
1	Star Wars	1977
2	Gone with the Wind	1939
3	The Wizard of Oz	1939
4	Indiana Jones and the Raiders of the Lost Ark	1981

$$\pi_{\text{MovieID}} (\sigma_{\text{year} < 1950} (\text{Movie}))$$

MovieID
2
3

Union, Intersection, Set-Difference

- Notation: $r \cup s$ $r \cap s$ $r - s$
- Defined as:

$$r \cup s = \{t \mid t \in r \textbf{ or } t \in s\}$$
$$r \cap s = \{t \mid t \in r \textbf{ and } t \in s\}$$
$$r - s = \{t \mid t \in r \textbf{ and } t \notin s\}$$

- For these operations to be well-defined:

  1. *r, s* must have the *same arity* (same number of attributes)
  2. The attribute domains must be *compatible* (e.g., 2nd column of *r* has same domain of values as the 2nd column of *s*)

- **The result:** Union, intersection, or difference of the inputs.
- **Schema:** Identical to schema of the first input relation.

Union, Intersection, and Set Difference Examples

MovieStar

StarID	Name	Gender
1	Harrison Ford	Male
2	Vivian Leigh	Female
3	Judy Garland	Female

Singer

StarID	SName	Gender
3	Judy Garland	Female
4	Christine Lavin	Female

MovieStar ∪ Singer

StarID	Name	Gender
1	Harrison Ford	Male
2	Vivian Leigh	Female
3	Judy Garland	Female
4	Christine Lavin	Female

Attributes compatible!

MovieStar ∩ Singer

StarID	Name	Gender
3	Judy Garland	Female

MovieStar – Singer

StarID	Name	Gender
1	Harrison Ford	Male
2	Vivian Leigh	Female

Set Operator Example

MovieStar

StarID	Name	Gender
1	Harrison Ford	Male
2	Vivian Leigh	Female
3	Judy Garland	Female

Singer

StarID	Name	Gender
3	Judy Garland	Female
4	Christine Lavin	Female

Find the names of stars that are Singers but not MovieStars

$$\pi_{Name}(Singers - MovieStars)$$

Name
Christine Lavin

Basic Relational Algebra Operations

Advanced Relational Algebra Operations

Implementation of SELECT and Join Operations

From Queries to Optimization