Chapter 2 The Relational Model of Data

Objectives

- Understand what is the relational model and database design basing relational model.
- Conceptualize data using the relational model.
- Understand what basic relational algebra operators under set semantics.
- Express queries using relational algebra.

Contents

2.1 An Overview of Data Models

- **Data model**: a collection of concepts for describing data, including 3 parts:
  - Structure of the data
    - Ex: arrays or objects
  - Operations on the data
    - Queries and modification on data
  - Constraints on the data
    - Limitations on the data

2.1 An Overview of Data Models

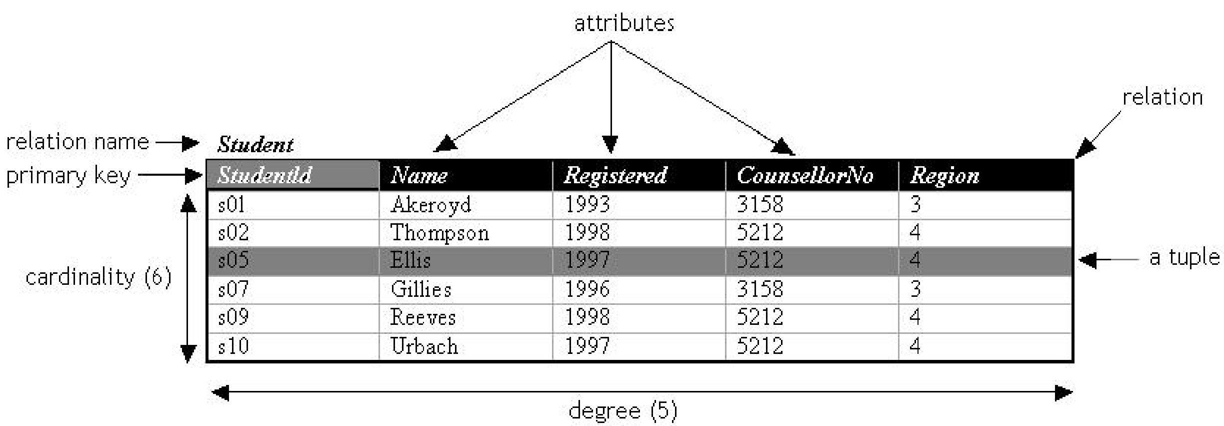
- The relational model, including object-relational extensions
- The semi-structured data model, including XML and related standards
- Semi-structured data resembles trees or graphs rather than tables or arrays
- XML, a way to represent data by hierarchically nested tagged elements
- Operations involve following paths in tree from an element to one or more of its nested sub elements, and so on
- Constraints involve the data type of values associated with a nested tag

2.1 An Overview of Data Models

```
<?xml version="1.0"?>
<!DOCTYPE PARTS SYSTEM "parts.dtd">
<?xml-stylesheet type="text/css" href="xmlpartsstyle.css"?>
<PARTS>
   <TITLE>Computer Parts</TITLE>
   <PART>
      <ITEM>Motherboard</ITEM>
      <MANUFACTURER>ASUS</MANUFACTURER>
      <MODEL>P3B-F</MODEL>
      <COST> 123.00</COST>
   </PART>
   <PART>
      <ITEM>Video Card</ITEM>
      <MANUFACTURER>ATI</MANUFACTURER>
      <MODEL>All-in-Wonder Pro</MODEL>
      <COST> 160.00</COST>
   </PART>
   <PART>
      <ITEM>Sound Card</ITEM>
      <MANUFACTURER>Creative Labs</MANUFACTURER>
      <MODEL>Sound Blaster Live</MODEL>
      <COST> 80.00</COST>
   </PART>
   <PART>
      <ITEM⊠ inch Monitor</ITEM>
      <MANUFACTURER>LG Electronics</MANUFACTURER>
      <MODEL> 995E</MODEL>
      <COST> 290.00</COST>
   </PART>
</PARTS>
```

2.2 Basics of the Relational Model

- **Relational model**
  - A relation is made up from 2 parts:
    - Schema: specifies name of relation, name of attributes and domain/type of one's.
      - Ex: Student(StudentID: string, Name: string, Registered: int, CounsellorNo: int, Region: int)
    - Instance: a table with rows and columns
      - Rows ~ cardinality; columns ~ degree/arity
  - **A simple thinking: a relation as a set of distinct rows or tuples**



Student

StudentId	*Name*	*Registered*	*CounsellorNo*	*Region*
s01	Akeroyd	1993	3158	3
s02	Thompson	1998	5212	4
s05	Ellis	1997	5212	4
s07	Gillies	1996	3158	3
s09	Reeves	1998	5212	4
s10	Urbach	1997	5212	4

2.2 Basics of the Relational Model

- Database schema: a set of schemas for the relations of a database
- An example of DB schema:

  - **Sailors**(<u>sid</u>: *integer*, sname: *string*, rating: *integer*, age:*real*)
  - **Boats**(<u>bid</u>:*integer* , bname: *string*, color: *string*)
  - **Reserves**(<u>sid: *integer*, bid: *integer*</u> , day: *date*)

Fpt University

2.2 Basics of the Relational Model

- Key attribute
- Non-key attribute
- Multi-valued attribute
- Derived- attribute
- Candidate key
- Primary key
- Foreign key

2.3 An Algebraic Query Language

Relational Algebra

- An algebra consists of operators and atomic operands
- Relational algebra is an example of an algebra, its atomic operands are
  - Variables that stand for relations
  - Constants, which are finite relations
- Relational algebra is a set of operations on relations
- Operations operate on one or more relations to create new relation

2.3 An Algebraic Query Language

Relational algebra fall into four classes

- Set operations – union, intersection, difference
- Selection and projection
- Cartesian product and joins
- Rename

2.3 An Algebraic Query Language

- **Set operations**

- Union

$$R \cup S = \{\, t \mid t \in R \vee t \in S \,\}$$

- Intersection

$$R \cap S = \{\, t \mid t \in R \wedge t \in S \,\}$$

- Difference

$$R \setminus S = \{\, t \mid t \in R \wedge t \notin S \,\}$$

- Intersection can be expressed in terms of set difference

$$R \cap S = R \setminus (R \setminus S)$$

R and S must be ‘type compatible’

- The same number of attributes
- The domain of corresponding attributes must be compatible

Set operations- Example

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Holywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Relation R

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Holywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	8/8/88

Relation S

Set operations- Example

R ∪ S

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Holywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88
Harrison Ford	789 Palm Dr., Beverly Hills	M	8/8/88

R ∩ S

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Holywood	F	9/9/99

R \ S

name	*address*	*gender*	*birthdate*
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Selection and projection

- **Selection**

  - R1 := σ_C(R2) with C illustrated conditions

  - **ex: $\sigma_{<C1>}(\sigma_{<C2>}(R)) = \sigma_{<C2>}(\sigma_{<C1>}(R)) = \sigma_{<C1>\ AND\ <C2>}$**

Movies

title	*year*	*length*	*genre*
Gone With the Wind	1939	231	Drama
Star Wars	1977	124	Scifi
Wayne's World	1992	95	Comedy

$\sigma_{length \geq 100}$(Movies)

title	*year*	*length*	*genre*
Gone With the Wind	1939	231	Drama
Star Wars	1977	124	Scifi

Selection and projection

- Projection S := $\pi_{A1,A2,\ldots,An}$ (R)
  - A1,A2,…,An are attributes of R
  - S relation schema S(A1,A2,…,An)

Movies

title	*year*	*length*	*genre*
Star Wars	1977	124	Scifi
Galaxy Quest	1999	104	Comedy
Wayne's World	1992	95	Comedy

$\pi_{title,year,length}$(Movies)

title	*year*	*length*
Star Wars	1977	124
Galaxy Quest	1999	104
Wayne's World	1992	95

π_{genre}(Movies)

genre
Scifi
Comedy

Cartesian product and joins

Cartesian product R3 := R1 X R2

Relation R

A	B
1	2
3	4

Relation S

B	C	D
2	5	6
4	7	8
9	10	11

Cartesian Product R X S

A	R.B	S.B	C	D
1	2	2	5	6
1	2	4	7	8
1	2	9	10	11
3	4	2	5	6
3	4	4	7	8
3	4	9	10	11

Cartesian product and joins

- theta joins R3 := R1 $\bowtie_{\text{<join condition>}}$ R2

A	B	C
1	2	3
6	7	8
9	7	8

Relation U

B	C	D
2	3	4
2	3	5
7	8	10

Relation V

A	U.B	U.C	V.B	V.C	D
1	2	3	2	3	4
1	2	3	2	3	5
1	2	3	7	8	10
6	7	8	7	8	10
9	7	8	7	8	10

Figure 2.17: Result of U $\bowtie_{A<D}$ V

A	U.B	U.C	V.B	V.C	D
1	2	3	7	8	10

Result of U $\bowtie_{A<D \text{ AND } U.B \neq V.B}$ V

Cartesian product and joins

Natural join R3 := R1 $\bowtie$ R2

Relation R

A	B
1	2
3	4

Relation S

B	C	D
2	5	6
4	7	8
9	10	11

Natural Join R $\bowtie$ S

A	B	C	D
1	2	5	6
3	4	7	8

Rename

- The ρ operation gives a new schema to a relation
- $\rho_{S(A1,\ldots,An)}(R)$ makes S be a relation with attributes A1,…,A*n* and the same tuples as R
- Simplified notation: S:=R (A1,A2,…,An)

Relation R

A	B
1	2
3	4

Relation S

B	C	D
2	5	6
4	7	8
9	10	11

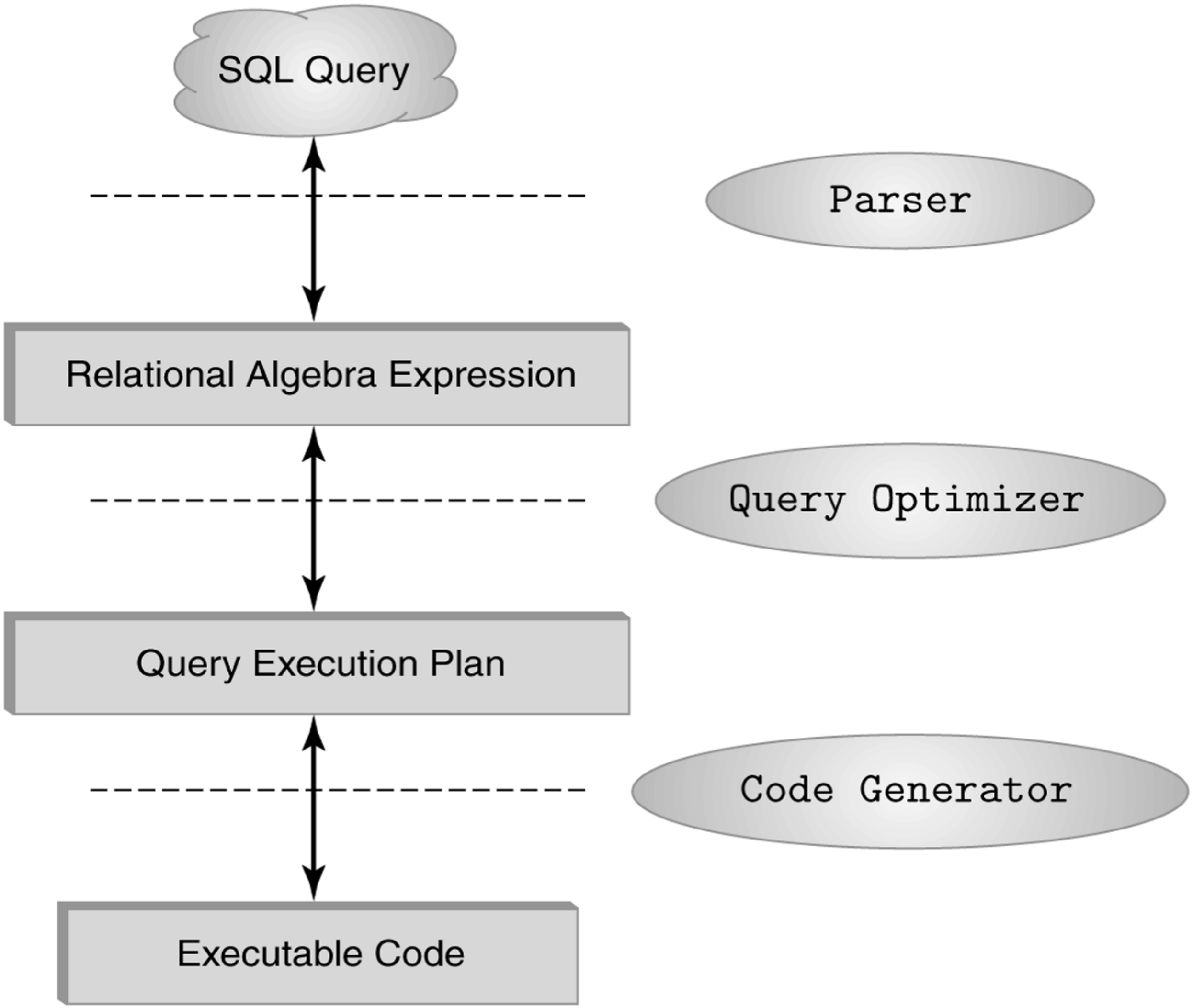
R X $\rho_{S(X,C,D)}$ (S)

A	B	X	C	D
1	2	2	5	6
1	2	4	7	8
1	2	9	10	11
3	4	2	5	6
3	4	4	7	8
3	4	9	10	11

Relational Expression

- How we need relational expression
- Relational algebra allows us to form expressions
- Relational expression is constructed by applying operations to the result of other operations
- Expressions can be presented as expression tree

The role of relational algebra in a DBMS



Relational Expression

Example: What are the titles and years of movies made by Fox that are at least 100 minutes long?

- (1) Select those Movies tuples that have length ≥ 100
- (2) Select those Movies tuples that have studioName=‘Fox’
- (3) Compute the intersection of (1) and (2)
- (4) Project the relation from (3) onto attributes title and year

Relational Expression

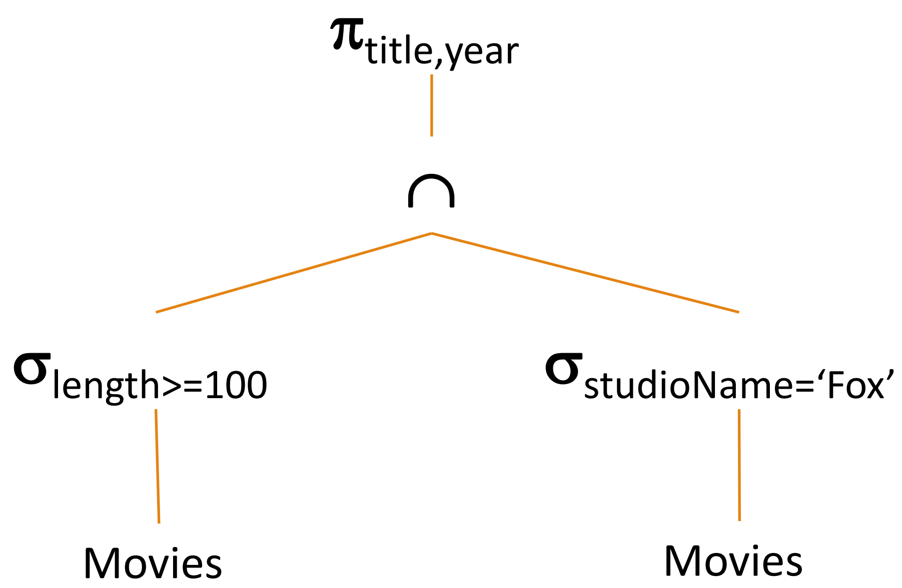


Figure 2.18: Expression tree for a relational algebra expression

$$\pi_{title,year}(\sigma_{length \geq 100}(Movies) \cap \sigma_{studioName='Fox'}(Movies))$$

$$\pi_{title,year}(\sigma_{length \geq 100 \text{ AND } studioName='Fox'}(Movies))$$

Exercise

```
Product(maker, model, type)
PC(model, speed, ram, hd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)
```

a) What PC models have a speed of at least 3.00?

b) Which manufacturers make laptops with a hard disk of at least 100GB?

c) Find the model number and price of all products (of any type) made by manufacturer *B*.

d) Find the model numbers of all color laser printers.

e) Find those manufacturers that sell Laptops, but not PC's.