UNIVERSITY OF ILLINOIS
AT URBANA-CHAMPAIGN

CS411 - Relational Algebra

illinois.edu

Review

- What is a data model?
- What two data models did we discuss?
- Why do DBMSs use the relational model?
- What are the components of a relation?
- What is the difference between a schema and an instance of a relation?

Review

- What is the domain of an attribute?
- What is a key?
- What is a relation called in SQL?
- What command creates a relation in SQL?

Algebra

- A mathematical system consisting of
  - *operands* - variables or values
  - *operators* - procedures that produce new values or variables from given values or variables

Examples

- Elementary algebra
  - operands - real numbers and variables
  - operators - addition, subtraction, division, …
- Boolean algebra
  - operands - truth values and variables
  - operators - conjunction, disjunction, negation
- More: algebraic number theory, algebraic geometry, linear algebra

Relational Algebra

- Operands: relations (and variables)
- Operations modify the relation, creating new relations
  - Changing tuples
  - Removing tuples
  - Changing attributes
  - Combining tuples of two relations

Relational Algebra

- Initially treat relations as *sets* of tuples
  - “a collection of well defined and *distinct* objects”
- Later generalize to *bags* or *multisets*
  - Relax the uniqueness constraint

Relational Algebra Operators

- We’re going to learn the relational algebra operators
  - How they create a new relation
  - Some have constraints that must be satisfied for the operation to be valid

Set Operations

- Union: $R \cup S$
  - Every tuple in R *or* in S
- Intersection: $R \cap S$
  - Every tuple in R *and* in S
- Difference: $R - S$
  - Every tuple in R but *not* in S

Set Operations

- Constraints:
  - Both R and S must have the same attributes with the same domains (aside from the name, the schema for R and S should be identical)
  - For clarity, we should keep a consistent ordering of the attributes while performing the set operation

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Hucklberry	Finn

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Hucklberry	Finn

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person1 \cup Person2$

First Name	Last Name
Holden	Caufield
Richard	Parker
Huckleberry	Finn
Bella	Swan
Marty	McFly

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Hucklberry	Finn

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person1 \cap Person2$

First Name	Last Name
Holden	Caufield
Richard	Parker

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Hucklberry	Finn

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person1 - Person2$

First Name	Last Name
Huckleberry	Finn

Fun Fact

- We don’t really need the intersection operator
- What tuples do relations R and S share?
  - Find the tuples in R not shared by S
  - Remove those from R

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

Fun Fact

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Hucklberry	Finn

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person1 - Person2$

First Name	Last Name
Huckleberry	Finn

$Person1 - (Person1 - Person2)$

First Name	Last Name
Holden	Caufield
Richard	Parker

$Person1 \cap Person2$

“Removal” operators

- Two operators to remove components from a relation
  - Projection
  - Selection

Projection

- $\pi_{A1,A2,\ldots,An}(R)$
- R is a relation and A1…An are attributes
- Creates a new relation with a subset of the attributes
    - All the tuples from R, but only the attributes A1, A2,…An

Example

Person

First Name	Last Name	Phone	Email
Holden	Caufield	(217)-555-3251	nophoney@hotmail.com
Richard	Parker	(217)-555-1212	pi_delicious@gmail.com
Luke	Skywalker	(217)-555-2917	wompratbullseye@gmail.com
Marty	McFly	(217)-555-1987	delorian88@gmail.com

$$\pi_{FirstName,LastName}(Person)$$

First Name	Last Name
Holden	Caufield
Richard	Parker
Luke	Skywalker
Marty	McFly

Selection

- $\sigma_C(R)$
- R is a relation and C is a boolean condition related to tuples
  - Example: “(x<y) and not (b=c)”
- Creates a new relation consisting of those tuples for which C is true

Example

Person

First Name	Last Name	Phone	Email
Holden	Caufield	(217)-555-3251	nophoney@hotmail.com
Richard	Parker	(217)-555-1212	pi_delicious@gmail.com
Luke	Skywalker	(217)-555-2917	wompratbullseye@gmail.com
Marty	McFly	(217)-555-1987	delorian88@gmail.com

$$\sigma_{FirstName="\text{Luke}" OR\ LastName="Parker"}(Person)$$

First Name	Last Name	Phone	Email
Richard	Parker	(217)-555-1212	pi_delicious@gmail.com
Luke	Skywalker	(217)-555-2917	wompratbullseye@gmail.com

Food for thought

- How are projection and selection similar?
- How are they different?

“Combining” operators

- Operators that combine tuples from relations
  - Cartesian product
  - natural join
  - theta join

Cartesian Product

- $R_1 \times R_2$
- Creates a new relation, pairing each tuple from R_1 with each tuple from R_2
- More theoretical than practical

Constraints

- The resulting schema has *all* of the attributes from both relations
  - What if some attributes have the same name?
  - Rename them by prepending the table name
  - For example relations R1(A,B,C) and R2(A,D)
    - R1×R2 has attributes R1.A,R2.A,B,C,D

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Lithium	Nevermind	4:17

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Lithium	Nevermind	4:17

$$Album \times Song$$

Album.AlbumTitle	BandName	DateReleased	SongTitle	Song.AlbumTitle	Length
Nevermind	Nirvana	09/24/1991	Breed	Nevermind	3:03
Nevermind	Nirvana	09/24/1991	Feel The Pain	Without a Sound	4:18
Nevermind	Nirvana	09/24/1991	Lithium	Nevermind	4:17
Without a Sound	Dinosaur Jr.	08/23/1994	Breed	Nevermind	3:03
Without a Sound	Dinosaur Jr.	08/23/1994	Feel The Pain	Without a Sound	4:18
Without a Sound	Dinosaur Jr.	08/23/1994	Lithium	Nevermind	4:17

Natural Join

- $R_1 \bowtie R_2$
- Identifies attributes common to each relation
- Creates a new relation, pairing each tuple from R_1 and R_2 where the common attributes are equal

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Lithium	Nevermind	4:17
Siva	Gish	4:21

$Album \bowtie Song$

AlbumTitle	BandName	DateReleased	SongTitle	Length
Nevermind	Nirvana	09/24/1991	Breed	3:03
Nevermind	Nirvana	09/24/1991	Lithium	4:17
Without a Sound	Dinosaur Jr.	08/23/1994	Feel The Pain	4:18

Example

- We call tuples that fail to pair *dangling tuples*

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Lithium	Nevermind	4:17
Siva	Gish	4:21

Theta-Join

- $R_1 \bowtie_{\mathrm{C}} R_2$
- Joins tuples from R_1 and R_2 such that boolean condition C is true
- Historically, “C” was designated with a “theta”
- Most commonly, C is of the type “A=B”
  - Called an *equi-join*

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Lithium	Nevermind	4:17

$$Album \bowtie_{BandName=DinosaurJr\ AND\ Length<4:00} Song$$

Album.AlbumTitle	BandName	DateReleased	SongTitle	Song.AlbumTitle	Length
Without a Sound	Dinosaur Jr.	08/23/1994	Breed	Nevermind	3:03

Food for thought

- If R_1 has *m* tuples and R_2 has *n* tuples
  - – How many tuples are in $R_1 \times R_2$?
  - – How many tuples are in $R_1 \bowtie R_2$?
  - – How many tuples in $R_1 \bowtie R_1$?

More food for thought

- Why have join operators when we can create them with cross product, projection, and selection?

$$R_1 \bowtie_C R_2 = \sigma_C(R_1 \times R_2)$$

$$R_1 \bowtie R_2 = \pi_L(\sigma_C(R_1 \times R_2))$$

Rename operator

- $\rho_{S(A_1,A_2,\ldots A_n)}(R)$
- Creates a new relation with the same tuples as R
  - Relation renamed S
  - Attributes renamed $A_1,A_2,\ldots,A_n$

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

$$\rho_{Simple(Title,Band,Released)}(Album)$$

Simple

Title	Band	Released
Nevermind	Nirvana	09/24/1991
Without a Sound	Dinosaur Jr.	08/23/1994

Writing Queries

- Given these relations:

  Album(AlbumTitle,BandName,DateReleased)

  Song(SongTitle,AlbumTitle,Length)

  Band(BandName,City,Genre,Formed,Ended,Label)

  Musician(FirstName,LastName,Band,Instrument, Birthday)

Writing Queries

- Write queries for the following:
  1. The length of the song “Cruel”
  2. The names of the songs not written by “Kiss”
  3. The instruments played by musicians in bands from ‘London’
  4. Find the name of all albums, songs, and bands
  5. Find every album with exactly one song

Writing Queries

- Write queries for the following:

6. The names of musicians in three or more bands
7. The band with the most recent album in the database

Example 1

The length of the song “Cruel”

Example 2

The names of the songs not written by “Kiss”

Example 3

The instruments played by musicians in bands from ‘London’

Example 4

Find the name of all albums, songs, and bands

Example 5

Find every album with exactly one song

Example 6

The names of musicians in three or more bands

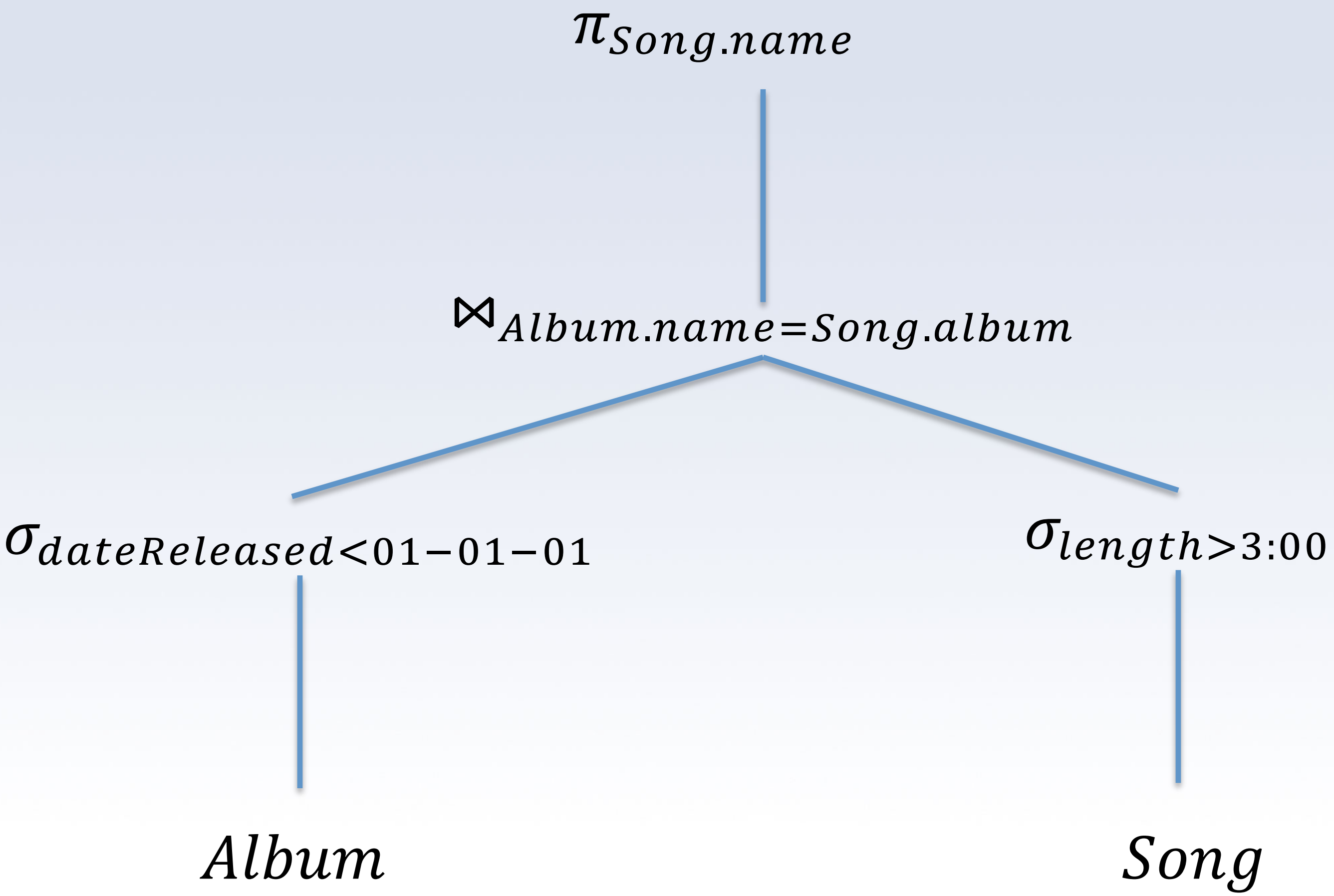
Example 7

The band with the most recent album in the database

Expression Trees

- Relational algebra expressions are represented internally as a tree
- Basically, queries are trees
  - Internal nodes are operators
  - Leaves are relations
- Perform query by a *post-order* traversal

Example



$\pi_{Song.name}$

$\bowtie_{Album.name=Song.album}$

$\sigma_{dateReleased<01-01-01}$

$\sigma_{length>3:00}$

$Album$

$Song$

Order of Precedence

- Precedence in relational algebra:
  1. Unary operators
  2. Products and joins
  3. Intersection
  4. Union and difference
- Obviously parentheses override these

Constraints

- In a data model, we need
  - structure: relations
  - operations: relational algebra
  - constraints: ???

Constraints

- We can use relational algebra and set operations:
  1. $R = \emptyset$
     - The result of this query must be empty
  2. $R \subseteq S$
     - Every tuple in R's result must be in S's result

Referential Integrity

- All values for attribute A in Relation R must appear as a value for attribute B in relation S
- Expressed as $\pi_A(R) \subseteq \pi_B(S)$
  - Equivalently $\pi_A(R) - \pi_B(S) = \emptyset$

Example

Values for the *AlbumName* attribute in relation *Song* must appear in the *Album* table

Album(AlbumTitle,BandName,DateReleased)

Song(SongTitle,AlbumTitle,Length)

$$\pi_{AlbumName}(Song) \subseteq \pi_{AlbumName}(Album)$$

Key Constraints

- All values for these attributes must be unique
  1. Rename two copies of the table
  2. Take the cross product of the relation with itself
  3. Select on the attributes for the key being equal
  4. Ensure that the result is the empty

Key Constraints

1. Rename two copies of the table
2. Take the cross product of the relation with itself
3. Select on the attributes in the key being equal
4. Ensure that the result is the empty

$$\sigma_{R1.key=R2.key}\left(\rho_{R1(A1\ldots An)}(R)\times\rho_{R2(A1\ldots An)}(R)\right)=\emptyset$$

Example

Song(SongTitle,AlbumTitle,Length)

$$\sigma_{S1.title=S2.title}\left(\rho_{S1(title,album,length)}(Song)\times\rho_{S2(title,album,name)}(Song)\right)$$
$$=\emptyset$$

Value Constraints

- All values of an attribute must not take certain illegal values
- Simple: just use the selection operator

$$\sigma_{Attribute1 \neq IllegalValue2\ AND\ Attribute2 \neq IllegalValue2}(R) = \emptyset$$

Example

Musician(FirstName,LastName,Band,Sex)

$$\sigma_{Sex \neq Female\ AND\ Sex \neq Male}(R) = \emptyset$$

Extending Relational Algebra

- This classical theoretical model of queries doesn't reflect all aspects of practical implementation
- We need to extend both the structure (operands) and the operators
  - structure: extend tuples from sets to *bags*
  - operators: add grouping, aggregation, and other new operators

Bags

- Also called “multisets”
- Generalize the concept of sets
- Members can appear more than once

Example

Person

First Name	Last Name	Phone	Email
Holden	Caufield	(217)-555-3251	nophoney@hotmail.com
Richard	Parker	(217)-555-1212	pi_delicious@gmail.com
Luke	Skywalker	(217)-555-2917	wompratbullseye@gmail.com
Marty	McFly	(217)-555-1987	delorian88@gmail.com
Richard	Parker	(217)-555-1212	pi_delicious@gmail.com
Luke	Skywalker	(217)-555-2917	wompratbullseye@gmail.com
Marty	McFly	(217)-555-1987	delorian88@gmail.com
Richard	Parker	(217)-555-1212	pi_delicious@gmail.com

Bags

- More efficient
  - Union or projection can require duplicate elimination
- Make new operations possible
  - Example: Average salary of people

$$AVERAGE(\pi_{Salary}(People))$$

  - This won't be correct if projection eliminates the duplicates

Example

$\pi_{Salary}(People)$

First Name	Last Name	Salary
Holden	Caufield	57,000
Richard	Parker	80,000
Luke	Skywalker	100,000
Marty	McFly	80,000

Average=79,250

Salary
57,000
80,000

verag 79,00

Set Operations

- Tuple t occurs m and times in R and and n times in S
- Union: $R \cup S$
  - Each tuple t appears n+m times
- Intersection: $R \cap S$
  - Each tuple t appears min(n,m) times
- Difference: R- S
  - Each tuple t appears max(0,n-m) times

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Holden	Caufield

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person1 \cup Person2$

First Name	Last Name
Holden	Caufield
Richard	Parker
Holden	Caufield
Bella	Swan
Marty	McFly
Richard	Parker
Holden	Caufield

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Holden	Caufield

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person1 \cap Person2$

First Name	Last Name
Holden	Caufield
Richard	Parker
Holden	Caufield

Examples

Person1

First Name	Last Name
Holden	Caufield
Richard	Parker
Holden	Caufield

Person2

Last Name	First Name
Swan	Bella
McFly	Marty
Parker	Richard
Caufield	Holden

$Person2 - Person1$

First Name	Last Name
Bella	Swan
Marty	McFly
Holden	Caufield

Other operators

- Selection, Projection, Product, and Joins all work the same, but duplicates are not removed

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Nevermind	Nirvana	09/24/1991

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Feel The Pain	Without a Sound	4:18

$$Album \times Song$$

Album.AlbumTitle	BandName	DateReleased	SongTitle	Song.AlbumTitle	Length
Nevermind	Nirvana	09/24/1991	Breed	Nevermind	3:03
Nevermind	Nirvana	09/24/1991	Feel The Pain	Without a Sound	4:18
Nevermind	Nirvana	09/24/1991	Feel The Pain	Without a Sound	4:18
Nevermind	Nirvana	09/24/1991	Breed	Nevermind	3:03
Nevermind	Nirvana	09/24/1991	Feel The Pain	Without a Sound	4:18
Nevermind	Nirvana	09/24/1991	Feel The Pain	Without a Sound	4:18

Example

Album

AlbumTitle	BandName	DateReleased
Nevermind	Nirvana	09/24/1991
Nevermind	Nirvana	09/24/1991

Song

SongTitle	AlbumTitle	Length
Breed	Nevermind	3:03
Feel The Pain	Without a Sound	4:18
Lithium	Nevermind	4:17
Siva	Gish	4:21

$Album \bowtie Song$

AlbumTitle	BandName	DateReleased	SongTitle	Length
Nevermind	Nirvana	09/24/1991	Breed	3:03
Nevermind	Nirvana	09/24/1991	Lithium	4:17
Nevermind	Nirvana	09/24/1991	Breed	3:03
Nevermind	Nirvana	09/24/1991	Lithium	4:17

Extended operations

- δ - duplicate elimination
- Aggregation
  - SUM, AVG, MIN, MAX, COUNT
- γ - grouping
- π - extended projection

Duplicate Elimination

- $\delta(R)$
- Converts a bag into a set

Example

Person

First Name	Last Name	Salary
Holden	Caufield	50,000
Richard	Parker	60,000
Luke	Skywalker	70,000
Marty	McFly	40,000
Richard	Parker	60,000
Luke	Skywalker	70,000
Marty	McFly	40,000
Richard	Parker	60,000

$\delta(Person)$

First Name	Last Name	Salary
Holden	Caufield	50,000
Richard	Parker	60,000
Luke	Skywalker	70,000
Marty	McFly	40,000

Aggregation

- Summarize values of one attribute
- Applied to an attribute of a relation
  - e.g. SUM(SALARY)
- Most of them are obvious
  - e.g. MAX finds the maximum value
- COUNT is a bit different
  - Counts the number of unique values

Example

Person

First Name	Last Name	Salary
Holden	Caufield	50,000
Richard	Parker	60,000
Luke	Skywalker	70,000
Marty	McFly	40,000
Richard	Parker	60,000
Luke	Skywalker	70,000
Marty	McFly	40,000
Richard	Parker	60,000

SUM(SALARY)=450,000

AVG(SALARY)=56,250

MAX(SALARY)=80,000

MIN(SALARY)=40,000

COUNT(FirstName)=4