DM505 Database Design and Programming
DM576 Database Systems

Panagiotis Tampakis

ptampakis@imada.sdu.dk

Relational Algebra

Why do we need a special Query Language?

- Won't conventional languages like C or Java suffice to ask and answer any computable question about relations?
- Relational algebra is useful because it is *less* powerful than C or Java
  - ease of programming and
  - the ability of the compiler to produce highly optimized code

What is an “Algebra”

- Mathematical system consisting of:
  - *Operands* – variables or values from which new values can be constructed
  - *Operators* – symbols denoting procedures that construct new values from given values
- Example:
  - Integers ..., -1, 0, 1, ... as operands
  - Arithmetic operations +/- as operators

What is Relational Algebra?

- An algebra whose operands are relations or variables that represent relations
- Operators are designed to do the most common things that we need to do with relations in a database
  - The result is an algebra that can be used as a *query language* for relations

Core Relational Algebra

- Set Operations
  - Union, intersection, and difference
  - Usual set operations, but *both operands must have the same relation schema*
- Filter Operations
  - Selection: picking certain rows
  - Projection: picking certain columns
- Products and joins: compositions of relations
- Renaming of relations and attributes

Set operations

- Union
  - $R_3 = R_1 \cup R_2$

R_1(A, B)

A	B
1	2
3	4

R_2(A, B)

A	B
5	6
1	2
9	10

R_3(A, B)

A	B
1	2
3	4
5	6
9	10

Set operations

- Intersection
  - $R_3 = R_1 \cap R_2$

$R_1($

A,	B)
1	2
3	4

$R_2($

A,	B)
5	6
1	2
9	10

$R_3($

A,	B)
1	2

Set operations

- Difference
  - $R_3 = R_1 - R_2$

$R_1($

A,	B)
1	2
3	4

$R_2($

A,	B)
5	6
1	2
9	10

$R_3($

A,	B)
3	4

Selection

- $R_1 := \sigma_C(R_2)$
  - *C* is a condition (as in "if" statements) that refers to attributes of R_2
  - R_1 is all those tuples of R_2 that satisfy *C*

Example: Selection

Relation Sells:

bar	beer	price
Cafe Chino	Od. Cla.	20
Cafe Chino	Erd. Wei.	35
Cafe Bio	Od. Cla.	20
Bryggeriet	Pilsener	31

ChinoMenu := $\sigma_{bar=\text{“Cafe Chino”}}$(Sells):

bar	beer	price
Cafe Chino	Od. Cla.	20
Cafe Chino	Erd. Wei.	35

Projection

- $R_1 := \pi_L(R_2)$
  - L is a list of attributes from the schema of R_2
  - R_1 is constructed by looking at each tuple of R_2, extracting the attributes on list L, in the order specified, and creating from those components a tuple for R_1
  - Eliminate duplicate tuples, if any

Example: Projection

Relation Sells:

bar	beer	price
Cafe Chino	Od. Cla.	20
Cafe Chino	Erd. Wei.	35
Cafe Bio	Od. Cla.	20
Bryggeriet	Pilsener	31

Prices := $\pi_{beer,price}$(Sells):

beer	price
Od. Cla.	20
Erd. Wei.	35
Pilsener	31

Extended Projection

- Using the same π_L operator, we allow the list L to contain arbitrary expressions involving attributes:
  1. Arithmetic on attributes, e.g., $A+B$->C
  2. Duplicate occurrences of the same attribute

Example: Extended Projection

R = (

A	B
1	2
3	4

)

$\pi_{A+B\text{->}C,A,A}$ (R) =

C	A_1	A_2
3	1	1
7	3	3

Exercise

class	*type*	*country*	*numGuns*	*bore*	*displacement*
Bismarck	bb	Germany	8	15	42000
Iowa	bb	USA	9	16	46000
Kongo	bc	Japan	8	14	32000
North Carolina	bb	USA	9	16	37000
Renown	bc	Gt. Britain	6	15	32000
Revenge	bb	Gt. Britain	8	15	29000
Tennessee	bb	USA	12	14	32000
Yamato	bb	Japan	9	18	65000

(a) Sample data for relation Classes

name	*date*
Denmark Strait	5/24-27/41
Guadalcanal	11/15/42
North Cape	12/26/43
Surigao Strait	10/25/44

(b) Sample data for relation Battles

ship	*battle*	*result*
Arizona	Pearl Harbor	sunk
Bismarck	Denmark Strait	sunk
California	Surigao Strait	ok
Duke of York	North Cape	ok
Fuso	Surigao Strait	sunk
Hood	Denmark Strait	sunk
King George V	Denmark Strait	ok
Kirishima	Guadalcanal	sunk
Prince of Wales	Denmark Strait	damaged
Rodney	Denmark Strait	ok
Scharnhorst	North Cape	sunk
South Dakota	Guadalcanal	damaged
Tennessee	Surigao Strait	ok
Washington	Guadalcanal	ok
West Virginia	Surigao Strait	ok
Yamashiro	Surigao Strait	sunk

(c) Sample data for relation Outcomes

name	*class*	*launched*
California	Tennessee	1921
Haruna	Kongo	1915
Hiei	Kongo	1914
Iowa	Iowa	1943
Kirishima	Kongo	1915
Kongo	Kongo	1913
Missouri	Iowa	1944
Musashi	Yamato	1942
New Jersey	Iowa	1943
North Carolina	North Carolina	1941
Ramillies	Revenge	1917
Renown	Renown	1916
Repulse	Renown	1916
Resolution	Revenge	1916
Revenge	Revenge	1916
Royal Oak	Revenge	1916
Royal Sovereign	Revenge	1916
Tennessee	Tennessee	1920
Washington	North Carolina	1941
Wisconsin	Iowa	1944
Yamato	Yamato	1941

Figure 2.23: Sample data for relation Ships

a) Give the class names and countries of the classes that carried guns of at least 16-inch bore.

b) Find the ships launched prior to 1921.

c) Find the ships sunk in the battle of the Denmark Strait.

Cartesian Product

- $R_3 := R_1 \text{ X } R_2$
  - Pair each tuple t_1 of R_1 with each tuple t_2 of R_2
  - Concatenation t_1t_2 is a tuple of R_3
  - Schema of R_3 is the attributes of R_1 and then R_2, in order
  - But beware attribute *A* of the same name in R_1 and R_2: use $R_1.A$ and $R_2.A$

Example: $R_3 := R_1 \text{ X } R_2$

R_1(

A,	B
1	2
3	4

)

R_2(

B,	C
5	6
7	8
9	10

)

R_3(

A,	R_1.B,	R_2.B,	C
1	2	5	6
1	2	7	8
1	2	9	10
3	4	5	6
3	4	7	8
3	4	9	10

)

Theta-Join

- $R_3 := R_1 \bowtie_C R_2$
  - Take the product R_1 X R_2
  - Then apply σ_C to the result
- As for σ, C can be any boolean-valued condition
  - Historic versions of this operator allowed only A θ B, where θ is =, <, etc.; hence the name "theta-join"

Example: Theta Join

Sells(

bar,	beer,	price
C.Ch.	Od.C.	20
C.Ch.	Er.W.	35
C.Bi.	Od.C.	20
Bryg.	Pils.	31

)

Bars(

name,	addr
C.Ch.	Brandts
C.Bi.	Brandts
Bryg.	Flakhaven

)

BarInfo := Sells $\bowtie_{\text{Sells.bar = Bars.name}}$ Bars

BarInfo(

bar,	beer,	price,	name,	addr
C.Ch.	Od.C.	20	C.Ch.	Brandts
C.Ch.	Er.W.	35	C.Ch.	Brandts
C.Bi.	Od.C.	20	C.Bi.	Brandts
Bryg.	Pils.	31	Bryg.	Flakhaven

)

Natural Join

- A useful join variant (*natural* join) connects two relations by:
  - Equating attributes of the same name, and
  - Projecting out one copy of each pair of equated attributes
- Denoted $R_3 := R_1 \bowtie R_2$

Example: Natural Join

Sells(bar, beer, price)

bar,	beer,	price
C.Ch.	Od.Cl.	20
C.Ch.	Er.We.	35
C.Bi.	Od.Cl.	20
Bryg.	Pils.	31

Bars(bar, addr)

bar,	addr
C.Ch.	Brandts
C.Bi.	Brandts
Bryg.	Flakhaven

BarInfo := Sells $\bowtie$ Bars

Note: Bars.name has become Bars.bar to make the natural join "work"

BarInfo(bar, beer, price, addr)

bar,	beer,	price,	addr
C.Ch.	Od.Cl.	20	Brandts
C.Ch.	Er.We.	35	Brandts
C.Bi.	Od.Cl.	20	Brandts
Bryg.	Pils.	31	Flakhaven

Exercise

class	*type*	*country*	*numGuns*	*bore*	*displacement*
Bismarck	bb	Germany	8	15	42000
Iowa	bb	USA	9	16	46000
Kongo	bc	Japan	8	14	32000
North Carolina	bb	USA	9	16	37000
Renown	bc	Gt. Britain	6	15	32000
Revenge	bb	Gt. Britain	8	15	29000
Tennessee	bb	USA	12	14	32000
Yamato	bb	Japan	9	18	65000

(a) Sample data for relation Classes

name	*class*	*launched*
California	Tennessee	1921
Haruna	Kongo	1915
Hiei	Kongo	1914
Iowa	Iowa	1943
Kirishima	Kongo	1915
Kongo	Kongo	1913
Missouri	Iowa	1944
Musashi	Yamato	1942
New Jersey	Iowa	1943
North Carolina	North Carolina	1941
Ramillies	Revenge	1917
Renown	Renown	1916
Repulse	Renown	1916
Resolution	Revenge	1916
Revenge	Revenge	1916
Royal Oak	Revenge	1916
Royal Sovereign	Revenge	1916
Tennessee	Tennessee	1920
Washington	North Carolina	1941
Wisconsin	Iowa	1944
Yamato	Yamato	1941

Figure 2.23: Sample data for relation Ships

name	*date*
Denmark Strait	5/24-27/41
Guadalcanal	11/15/42
North Cape	12/26/43
Surigao Strait	10/25/44

(b) Sample data for relation Battles

ship	*battle*	*result*
Arizona	Pearl Harbor	sunk
Bismarck	Denmark Strait	sunk
California	Surigao Strait	ok
Duke of York	North Cape	ok
Fuso	Surigao Strait	sunk
Hood	Denmark Strait	sunk
King George V	Denmark Strait	ok
Kirishima	Guadalcanal	sunk
Prince of Wales	Denmark Strait	damaged
Rodney	Denmark Strait	ok
Scharnhorst	North Cape	sunk
South Dakota	Guadalcanal	damaged
Tennessee	Surigao Strait	ok
Washington	Guadalcanal	ok
West Virginia	Surigao Strait	ok
Yamashiro	Surigao Strait	sunk

(c) Sample data for relation Outcomes

d) The treaty of Washington in 1921 prohibited capital ships heavier than 35,000 tons. List the ships that violated the treaty of Washington.

e) List the name, displacement, and number of guns of the ships engaged in the battle of Guadalcanal.

f) List all the capital ships mentioned in the database. (Remember that all these ships may not appear in the Ships relation.)

Renaming

- The ρ operator gives a new schema to a relation
- $R_1 := \rho_{R_1(A_1,\ldots,A_n)}(R_2)$ makes R_1 be a relation with attributes $A_1,\ldots,A_n$ and the same tuples as R_2
- Simplified notation: $R_1(A_1,\ldots,A_n) := R_2$

Example: Renaming

Bars(

name,	addr
C.Ch.	Reventlo.
C.Bi.	Brandts
Bryg.	Flakhaven

)

R(bar, addr) := Bars

R(

bar,	addr
C.Ch.	Reventlo.
C.Bi.	Brandts
Bryg.	Flakhaven

)

Building Complex Expressions

- Combine operators with parentheses and precedence rules
- Three notations, just as in arithmetic:
  1. Sequences of assignment statements
  2. Expressions with several operators
  3. Expression trees

Sequences of Assignments

- Create temporary relation names
- Renaming can be implied by giving relations a list of attributes
- Example: $R_3 := R_1 \bowtie_C R_2$ can be written:

  $R_4 := R_1 \text{ X } R_2$

  $R_3 := \sigma_C(R_4)$

Expressions in a Single Assignment

- Example: the theta-join $R_3 := R_1 \bowtie_C R_2$ can be written: $R_3 := \sigma_C (R_1 \text{ X } R_2)$
- Precedence of relational operators:
  1. [σ, π, ρ] (highest)
  2. [X, $\bowtie$]
  3. $\cap$
  4. [$\cup$, —]

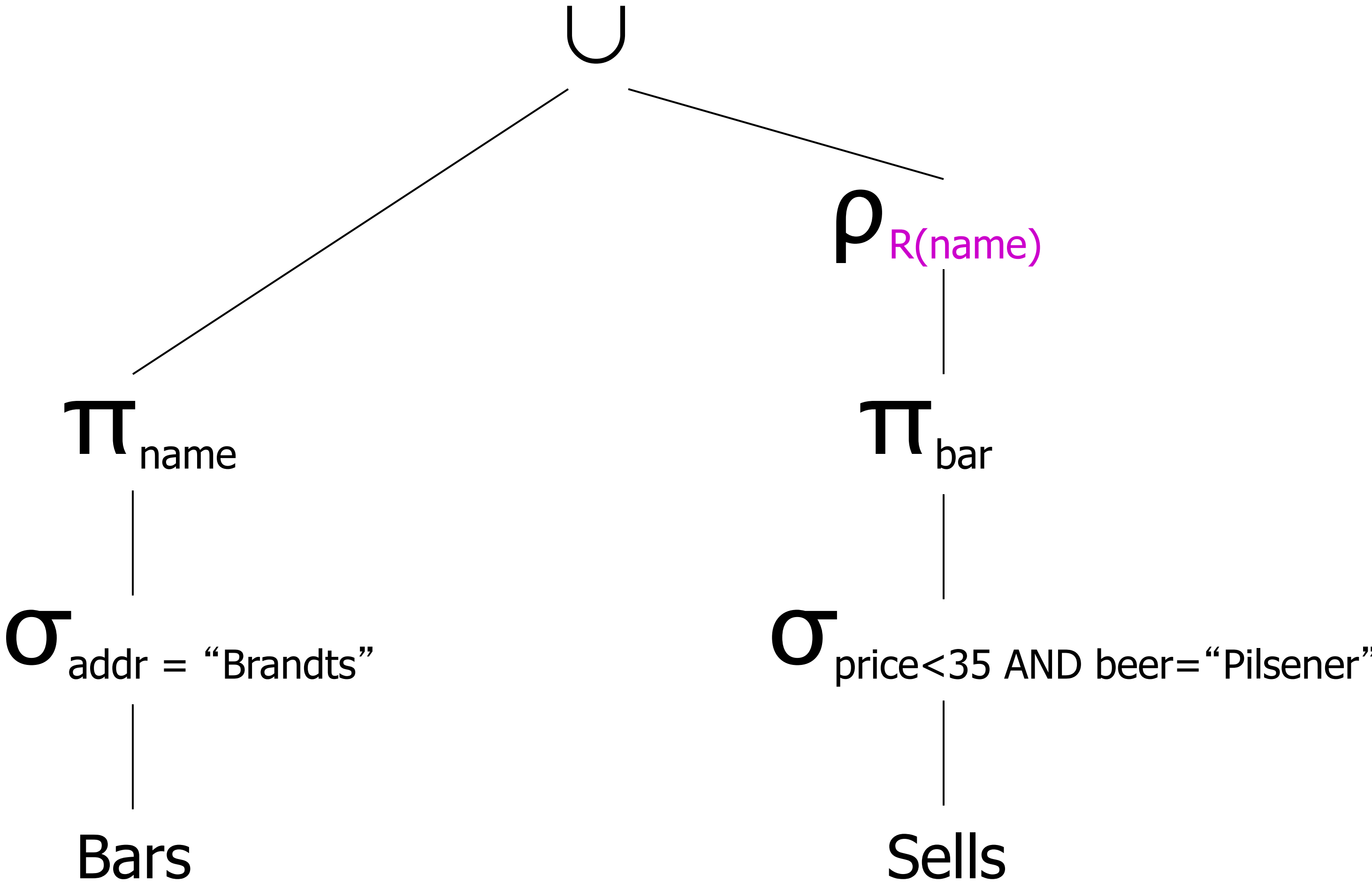
Expression Trees

- Leaves are operands – either variables standing for relations or particular, constant relations
- Interior nodes are operators, applied to their child or children

Example: Tree for a Query

- Using the relations Bars(name, addr) and Sells(bar, beer, price), find the names of all the bars that are either at Brandts or sell Pilsener for less than 35:

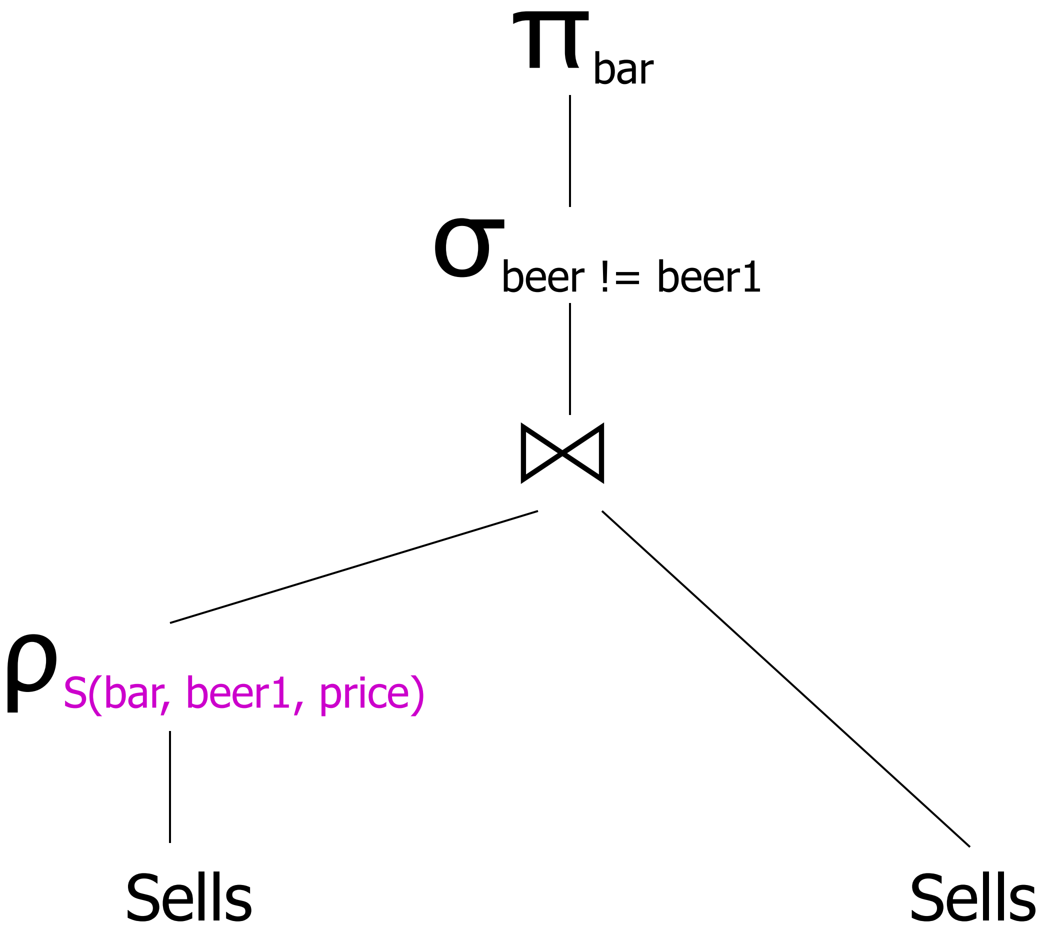
As a Tree:


∪
ρ R(name)
π name
π bar
σ addr = “Brandts”
σ price<35 AND beer=“Pilsener”
Bars
Sells

Example: Self-Join

- Using Sells(bar, beer, price), find the bars that sell two different beers at the same price
- Strategy: by renaming, define a copy of Sells, called S(bar, beer1, price). The natural join of Sells and S consists of quadruples (bar, beer, beer1, price) such that the bar sells both beers at this price

The Tree



Schemas for Results

- Union, intersection, and difference: the schemas of the two operands must be the same → same schema for the result
- Selection: schema of the result is the same as the schema of the operand
- Projection: list of attributes

Exercise

class	type	country	numGuns	bore	displacement
Bismarck	bb	Germany	8	15	42000
Iowa	bb	USA	9	16	46000
Kongo	bc	Japan	8	14	32000
North Carolina	bb	USA	9	16	37000
Renown	bc	Gt. Britain	6	15	32000
Revenge	bb	Gt. Britain	8	15	29000
Tennessee	bb	USA	12	14	32000
Yamato	bb	Japan	9	18	65000

(a) Sample data for relation Classes

name	class	launched
California	Tennessee	1921
Haruna	Kongo	1915
Hiei	Kongo	1914
Iowa	Iowa	1943
Kirishima	Kongo	1915
Kongo	Kongo	1913
Missouri	Iowa	1944
Musashi	Yamato	1942
New Jersey	Iowa	1943
North Carolina	North Carolina	1941
Ramillies	Revenge	1917
Renown	Renown	1916
Repulse	Renown	1916
Resolution	Revenge	1916
Revenge	Revenge	1916
Royal Oak	Revenge	1916
Royal Sovereign	Revenge	1916
Tennessee	Tennessee	1920
Washington	North Carolina	1941
Wisconsin	Iowa	1944
Yamato	Yamato	1941

Figure 2.23: Sample data for relation Ships

name	date
Denmark Strait	5/24-27/41
Guadalcanal	11/15/42
North Cape	12/26/43
Surigao Strait	10/25/44

(b) Sample data for relation Battles

ship	battle	result
Arizona	Pearl Harbor	sunk
Bismarck	Denmark Strait	sunk
California	Surigao Strait	ok
Duke of York	North Cape	ok
Fuso	Surigao Strait	sunk
Hood	Denmark Strait	sunk
King George V	Denmark Strait	ok
Kirishima	Guadalcanal	sunk
Prince of Wales	Denmark Strait	damaged
Rodney	Denmark Strait	ok
Scharnhorst	North Cape	sunk
South Dakota	Guadalcanal	damaged
Tennessee	Surigao Strait	ok
Washington	Guadalcanal	ok
West Virginia	Surigao Strait	ok
Yamashiro	Surigao Strait	sunk

(c) Sample data for relation Outcomes

Build Expression Trees for the following queries:

d) The treaty of Washington in 1921 prohibited capital ships heavier than 35,000 tons. List the ships that violated the treaty of Washington.

e) List the name, displacement, and number of guns of the ships engaged in the battle of Guadalcanal.

f) List all the capital ships mentioned in the database. (Remember that all these ships may not appear in the Ships relation.)

Schemas for Results

- Product: schema is the attributes of both relations
  - Use $R_1.A$ and $R_2.A$, etc., to distinguish two attributes named *A*
- Theta-join: same as product
- Natural join: union of the attributes of the two relations
- Renaming: defined by the operator

Relational Algebra on Bags

- A *bag* (or *multiset*) is like a set, but an element may appear more than once
- Example: {1,2,1,3} is a bag
- Example: {1,2,3} is also a bag that happens to be a set

Why Bags?

- SQL, the most important query language for relational databases, is actually a bag language
- Some operations, like projection, are more efficient on bags than sets

Operations on Bags

- Selection applies to each tuple, so its effect on bags is like its effect on sets.
- Projection also applies to each tuple, but as a bag operator, we do not eliminate duplicates.
- Products and joins are done on each pair of tuples, so duplicates in bags have no effect on how we operate.

Example: Bag Selection

R(

A,	B
1	2
5	6
1	2

)

$\sigma_{A+B<5}(R) =$

A	B
1	2
1	2

Example: Bag Projection

R(

A,	B)
1	2
5	6
1	2

$\pi_A(R) =$

A
1
5
1

Example: Bag Product

R(

A,	B
1	2
5	6
1	2

)

S(

B,	C
3	4
7	8

)

R x S =

A	R.B	S.B	C
1	2	3	4
1	2	7	8
5	6	3	4
5	6	7	8
1	2	3	4
1	2	7	8

Example: Bag Theta-Join

R(

A,	B
1	2
5	6
1	2

)

S(

B,	C
3	4
7	8

)

$R \bowtie_{R.B<S.B} S =$

A	R.B	S.B	C
1	2	3	4
1	2	7	8
5	6	7	8
1	2	3	4
1	2	7	8

Bag Union

- An element appears in the union of two bags the sum of the number of times it appears in each bag
- Example: $\{1,2,1\} \cup \{1,1,2,3,1\} = \{1,1,1,1,1,2,2,3\}$

Bag Intersection

- An element appears in the intersection of two bags the minimum of the number of times it appears in either.
- Example:
  $\{1,2,1,1\} \cap \{1,2,1,3\} = \{1,1,2\}$.

Bag Difference

- An element appears in the difference $A - B$ of bags as many times as it appears in A, minus the number of times it appears in B.
  - But never less than 0 times.
- Example: $\{1,2,1,1\} - \{1,2,3\} = \{1,1\}$.

Exercise

Exercise 5.1.2: Let PC be the relation of Fig. 2.21(a), and suppose we compute the projection $\pi_{hd}(\text{PC})$. What is the value of this expression as a set? As a bag? What is the average value of tuples in this projection, when treated as a set? As a bag?

model	*speed*	*ram*	*hd*	*price*
1001	2.66	1024	250	2114
1002	2.10	512	250	995
1003	1.42	512	80	478
1004	2.80	1024	250	649
1005	3.20	512	250	630
1006	3.20	1024	320	1049
1007	2.20	1024	200	510
1008	2.20	2048	250	770
1009	2.00	1024	250	650
1010	2.80	2048	300	770
1011	1.86	2048	160	959
1012	2.80	1024	160	649
1013	3.06	512	80	529

(a) Sample data for relation PC

Beware: Bag Laws != Set Laws

- Some, but *not all* algebraic laws that hold for sets also hold for bags
- Example: the commutative law for union ($R \cup S = S \cup R$) *does* hold for bags
  - Since addition is commutative, adding the number of times x appears in R and S does not depend on the order of R and S

Example: A Law That Fails

- Set union is *idempotent*, meaning that $S \cup S = S$
- However, for bags, if x appears n times in S, then it appears $2n$ times in $S \cup S$
- Thus $S \cup S$!= S in general
  - e.g., {1} ∪ {1} = {1,1} != {1}

Summary 2

More things you should know:

- Relational Algebra
- Selection, (Extended) Projection, Product, Join, Natural Join, Renaming
- Complex Operations as Sequences, Expressions, or Trees
- Difference between Sets and Bags