



DA 512H: Database Management Systems

Relational Calculus

Debangra Raj Neog
Mehta Family School of Data
Science and Artificial
Intelligence (MFSDS&AI)
IIT Guwahati

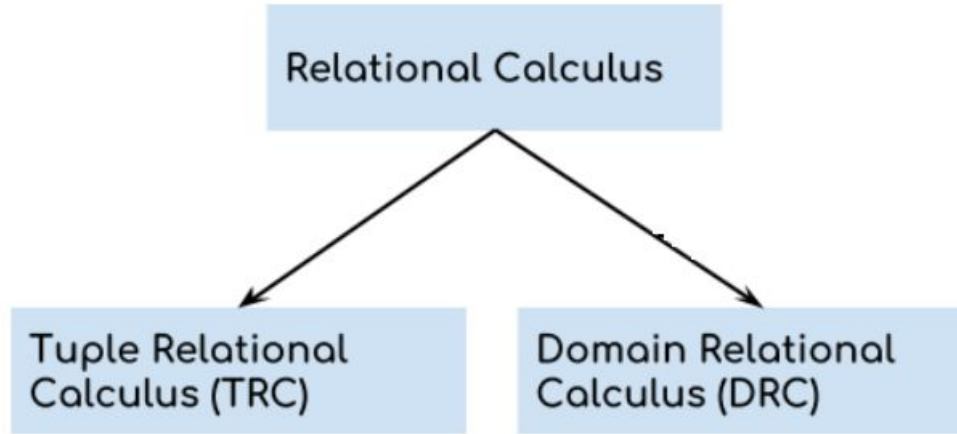
Slides courtesy:
Prof. Ashok Singh Sairam, IITG

Relational Query Languages

- Query languages allow the manipulation and retrieval of data from a database
- Two mathematical Query Languages form the basis for “real” languages (e.g. SQL),
- Relational Algebra
 - Procedural query language (step-by-step procedure)
 - used to represent execution plans
- **Relational Calculus**
 - Non-procedural (declarative) query language
 - Describe **what** you want, rather than **how** to compute it
 - Foundation for SQL

Relational Calculus

Relational Calculus types



Tuple Relational Calculus

- Interested in finding tuples for which a predicate (condition) is true. Based on use of tuple variables.
- Tuple variable is a variable that 'ranges over' a named relation: i.e., variable whose only permitted values are tuples of the relation.
- Specify range of a tuple variable S as the Sailor relation as:

Sailor(S) or $S \in \text{Sailor}$

Sailors

<i>sid</i>		<i>rating</i>	<i>max</i>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

Tuple Relational Calculus Query

- A TRC query is of the form

$$\{T|p(T)\}$$

which means select those tuples T that satisfy a given condition(s) $p(T)$

T : resulting tuples

$p(T)$: Predicate (condition) used to fetch T .

- That is the set of all tuples T where formula $p(T)$ evaluates to TRUE

TRC: Example

- To find details of all sailors whose age is less than 35
 $\{S \mid \text{Sailor}(S) \wedge S.\text{age} < 35\}$

<i>sid</i>	<i>sname</i>	<i>rating</i>	<i>age</i>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

S: Sailors

Quantifiers

- TRC can use two *quantifiers* to tell how many instances the predicate applies to:
 - Existential quantifier \exists ('there exists')
 - Universal quantifier \forall ('for all')
- Tuple variables qualified by \forall or \exists are called *bound* variables, otherwise called *free* variables.

TRC: Formula

$$\{T | p(T)\}$$

↑
Formula

- A (well-formed) formula is made out of atoms:
 - $R(S_i)$, where S_i is a tuple variable and R is a relation
 - $S_i.a_1 \text{ op } S_j.a_2$
 - $S_i.a_1 \text{ op constant}$
- op: Set of operator $\{<, \leq, =, \geq, >\}$

TRC: Formula

$$\{T | p(T)\}$$

↑
Formula

- A (well-formed) formula is made out of atoms:
 - $R(S_i)$, where S_i is a tuple variable and R is a relation
 - $S_i.a_1 \text{ op } S_j.a_2$
 - $S_i.a_1 \text{ op constant}$op: Set of operator $\{<, \leq, =, \geq, >\}$
- Can recursively build up formulae from atoms:
 - An atom is a formula
 - If p and q are formulae, so are their conjunction, $p \wedge q$; disjunction, $p \vee q$; and negation, $\sim p$
 - $\exists R(p(R))$, where $p(R)$ denotes a formula and R is a tuple variable
 - $\forall R(p(R))$

Example Relations

<u>sid</u>	<u>sname</u>	<u>rating</u>	<u>age</u>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

S3: Sailors

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

R2: Reserves

<u>bid</u>	<u>bname</u>	<u>color</u>
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

H1: Boats

TRC Example: Selection and Projection

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/98
22	102	10/10/98

- Ex1: Find all sailors with rating above 7

$$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7\}$$

<u>sid</u>	<u>sname</u>	<u>rating</u>	<u>age</u>
22	Dustin	7	45.0
29	Brutus	1	33.0

<u>bid</u>	<u>bname</u>	<u>color</u>
101	Interlake	blue
102	Interlake	red

- Ex2: Find names and ages of sailors with rating above 7.

$$\{S \mid \exists S1 \in \text{Sailors} (S1.\text{rating} > 7 \wedge S.\text{sname} = S1.\text{sname} \wedge S.\text{age} = S1.\text{age})\}$$

- Note, here S is a tuple variable of 2 fields (i.e. $\{S\}$ is a projection of *sailors*), since only 2 fields are ever mentioned and S is never used to range over any relations in the query.

Example Relations

Ex3: Find sailors rated > 7 who've reserved boat #103

<i>sid</i>	<i>sname</i>	<i>rating</i>	<i>age</i>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

S3: Sailors

<i>sid</i>	<i>bid</i>	<i>day</i>
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

R2: Reserves

<i>bid</i>	<i>bname</i>	<i>color</i>
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

H1: Boats

TRC Example: Joins

- **Ex3: Find sailors rated > 7 who've reserved boat #103**

$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7 \wedge \exists R (R \in \text{Reserves} \wedge R.\text{sid} = S.\text{sid} \wedge R.\text{bid} = 103)\}$

- Note the use of \exists to find a tuple in Reserves that 'joins with' the Sailors tuple under consideration

Example Relations

Ex4: Find sailors rated > 7 who've reserved a red boat

<u>sid</u>	<u>sname</u>	<u>rating</u>	<u>age</u>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

S3: Sailors

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

R2: Reserves

<u>bid</u>	<u>bname</u>	<u>color</u>
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

H1: Boats

TRC Example: Joins contd.

- Find sailors rated > 7 who've reserved a red boat

$$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7 \wedge \\ \exists R \in \text{Reserves} (R.\text{sid} = S.\text{sid} \\ \wedge \exists B \in \text{Boats} (B.\text{bid} = R.\text{bid} \\ \wedge B.\text{color} = \text{'red'}))\}$$

Example Relations

Ex5: Find sailors who've reserved **all** boats

<i>sid</i>	<i>sname</i>	<i>rating</i>	<i>age</i>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

S3: Sailors

<i>sid</i>	<i>bid</i>	<i>day</i>
<u>22</u>	<u>101</u>	<u>10/10/98</u>
<u>22</u>	<u>102</u>	<u>10/10/98</u>
<u>22</u>	<u>103</u>	<u>10/8/98</u>
<u>22</u>	<u>104</u>	<u>10/7/98</u>
<u>31</u>	<u>102</u>	<u>11/10/98</u>
<u>31</u>	<u>103</u>	<u>11/6/98</u>
<u>31</u>	<u>104</u>	<u>11/12/98</u>
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

R2: Reserves

<i>bid</i>	<i>bname</i>	<i>color</i>
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

H1: Boats

TRC Example: Division

- Ex5: Find sailors who've reserved **all** boats

$$\{S \mid S \in \text{Sailors} \wedge \\ \forall B \in \text{Boats} (\exists R \in \text{Reserves} \\ (S.\text{sid} = R.\text{sid} \\ \wedge B.\text{bid} = R.\text{bid})) \}$$

- Find all sailors S such that for each tuple B in Boats there is a tuple in Reserves showing that sailor S has reserved it.

<u>sid</u>	<u>sname</u>	rating	age
22	Dustin	7	45.0

Sailor

<u>sid</u>	<u>Sid</u>	day
22	101	10/10/98

Reserve

<u>bid</u>	<u>bname</u>	color
101	Interlake	blue

Boat

s

TRC Example: Division continued

- Ex6: Find sailors who've reserved all **Red** boats

<u>sid</u>	<u>sname</u>	rating	age
22	Dustin	7	45.0

Sailors

<u>sid</u>	<u>bid</u>	day
22	101	10/10/98

Reserves

<u>bid</u>	<u>bname</u>	color
101	Interlake	blue

Boats

TRC Example: Division continued

- Ex6: Find sailors who've reserved all **Red** boats

$\{ S \mid S \in \text{Sailors} \wedge$
 $\forall B \in \text{Boats} (B.\text{color} = \text{'red'} \Rightarrow$
 $\exists R \in \text{Reserves} (S.\text{sid} = R.\text{sid} \wedge B.\text{bid} = R.\text{bid})) \}$

<u>sid</u>	<u>sname</u>	rating	age
22	Dustin	7	45.0

Sailors

<u>sid</u>	<u>bid</u>	day
22	101	10/10/98

Reserves

<u>bid</u>	<u>bname</u>	color
101	Interlake	blue

Boats

TRC Example: Division continued

- Find sailors who've reserved all **Red** boats

$S \mid S \in \text{Sailors} \wedge$

$\forall B \in \text{Boats} (B.\text{color} = \text{'red'} \Rightarrow$

$\exists R \in \text{Reserves} (S.\text{sid} = R.\text{sid} \wedge B.\text{bid} = R.\text{bid}))\}$

$p \Rightarrow q$ is equivalent to $\neg p \vee q$.

		q	
p		T	F
	T	T	F
	F	T	T

<u>sid</u>	<u>sname</u>	rating	age
22	Dustin	7	45.0

Sailors

<u>sid</u>	<u>bid</u>	day
22	101	10/10/98

Reserves

<u>bid</u>	<u>bname</u>	color
101	Interlake	blue

Boats

TRC Example: Division continued

- Find sailors who've reserved all **Red** boats

$S \mid S \in \text{Sailors} \wedge$
 $\forall B \in \text{Boats} (B.\text{color} = \text{'red'} \Rightarrow$
 $\exists R \in \text{Reserves} (S.\text{sid} = R.\text{sid} \wedge B.\text{bid} = R.\text{bid})) \}$

$p \Rightarrow q$ is equivalent to $\neg p \vee q$. Alternatively...

$\{S \mid S \in \text{Sailors} \wedge$
 $\forall B \in \text{Boats} (B.\text{color} \neq \text{'red'} \vee$
 $\exists R \in \text{Reserves} (S.\text{sid} = R.\text{sid}$
 $\wedge B.\text{bid} = R.\text{bid})) \}$

TRC Query (Revisited)

- The use of **quantifiers** $\exists X$ and $\forall X$ in a formula is said to **bind** X in the formula.
 - A variable that is **not bound** is **free**.
- Let us revisit the definition of a **query**:
 - $\{T \mid p(T)\}$
- There is an important restriction
 - the variable **T** that appears to the left of \mid must be the **only** free variable in the formula $p(T)$.
 - in other words, all other tuple variables must be bound using a quantifier.

Unsafe Queries

- \exists syntactically correct calculus queries that have an infinite number of answers! Unsafe queries.
 - e.g., $\{S \mid \neg (S \in \textit{Sailors})\}$
 - Solution???? Don't do that!

Unsafe Queries

- \exists syntactically correct calculus queries that have an infinite number of answers! Unsafe queries.
 - e.g., $\{S \mid \neg(S \in Sailors)\}$
 - Solution???? Don't do that!
- A TRC is safe if all values in its results are members of its domain

Summary

- The relational model has rigorously defined query languages — simple and powerful.
- Relational algebra is more operational
 - useful as internal representation for query evaluation plans.
- Relational calculus is non-operational
 - users define queries in terms of what they want, not in terms of how to compute it. (Declarative)

Domain Relational Calculus

- Domain variable: represent values in the domain of an attribute
 - for example the domain variable can be integer if it represent an attribute whose domain is the set of integers

Domain Relational Calculus

- **Query** has the form:

$$\left\{ \langle x_1, x_2, \dots, x_n \rangle \mid p(\langle x_1, x_2, \dots, x_n \rangle) \right\}$$

x_i is either a domain variable or a constant

- **Answer** includes all tuples $\langle x_1, x_2, \dots, x_n \rangle$ that make the formula $p(\langle x_1, x_2, \dots, x_n \rangle)$ be true.
- **Formula** is recursively defined, starting with simple **atomic formulas** (getting rows from relations or making comparisons of values), and building bigger and better formulas using the **logical connectives**.

DRC Formulas

- Atomic formula:
 - $\langle X_1, X_2, \dots, X_n \rangle \in \text{Rname}$, or $X \text{ op } Y$, or $X \text{ op constant}$
 - op is $<, >, =, \leq, \geq$
- Formula
 - Atomic formula
 - $\neg p, p \vee q, p \wedge q$, where p and q are formulas
 - $\exists X(p(X))$, where variable X is free in $p(X)$, or
 - $\forall X(p(X))$, where variable X is free in $p(X)$
- The use of quantifiers $\exists X$ and $\forall X$ are said to bound the variable X

Free and Bound variables

- The use of quantifiers $\exists X$ and $\forall X$ are said to bound the variable
 - Variables that are not bound are free

- Let us revisit the query

$$\{\langle x_1, x_2, \dots, x_n \rangle \mid p(\langle x_1, x_2, \dots, x_n \rangle)\}$$

- There is an important restriction, the variables x_1, x_2, \dots, x_n that appear to the left of ' \mid ' must be the only free variable in the formula $p(..)$

Domain Relational Calculus

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

- Example: To find details of all sailors whose age is less than 35

$\{S \mid \text{Sailor}(S) \wedge S.\text{age} < 35\}$ (TRC)

Domain Relational Calculus

sid	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

- Example: To find details of all sailors whose age is less than 35

$\{S \mid \text{Sailor}(S) \wedge S.\text{age} < 35\}$ (TRC)

$\{ \langle I, N, T, A \rangle \mid (I, N, T, A) \in \text{Sailors} \wedge A > 35 \}$ (DRC)

- Note: $(I, N, T, A) \in \text{Sailors}$ the **domain variables** I , N , T and A are **bound** to fields of the same Sailors tuple

DRC: Examples

- Ex1: Find all sailors with a rating above 7

$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7\}$
(TRC)

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

DRC: Examples

- Ex1: Find all sailors with a rating above 7

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

$$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7\}$$

(TRC)

$$\{\langle I, N, T, A \rangle \mid \langle I, N, T, A \rangle \in \text{Sailors} \wedge T > 7\} \quad (\text{DRC})$$

- The condition ensures that the domain variables I , N , T and A are bound to fields of the same Sailors tuple.
- The term to the left of ' \mid ' (which should be read as *such that*) says that every tuple that satisfies $T > 7$ is in the answer.

DRC: Examples

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

<u>sid</u>	<u>bid</u>	<u>day</u>
<u>22</u>	101	10/10/98
<u>22</u>	102	10/10/98
<u>22</u>	103	10/8/98

- Ex2: Find sailors rating > 7 who've reserved boat #103

$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7 \wedge \exists R(R \in \text{Reserves} \wedge R.\text{sid} = S.\text{sid} \wedge R.\text{bid} = 103)\}$ (TRC)

DRC: Examples

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98

- Ex2: Find sailors rating > 7 who've reserved boat #103

$$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7 \wedge \exists R(R \in \text{Reserves} \wedge R.\text{sid} = S.\text{sid} \wedge R.\text{bid} = 103)\} \quad (\text{TRC})$$

$$\left\{ \langle I, N, T, A \rangle \mid \langle I, N, T, A \rangle \in \text{Sailors} \wedge T > 7 \wedge \exists Ir, Br, D \left(\langle Ir, Br, D \rangle \in \text{Reserves} \wedge Ir = I \wedge Br = 103 \right) \right\} \quad (\text{DRC})$$

- We have used $\exists Ir, Br, D (\dots)$ as a shorthand for $\exists Ir (\exists Br (\exists D(\dots)))$
- Note the use of \exists to find a tuple in Reserves that 'joins with' the Sailors tuple under consideration

DRC: Examples

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Sailors

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/96
58	103	11/12/96

Reserves

bid	bname	color
101	interlake	red
103	marine	green

Boats

- Ex3: Find sailors rated > 7 who've reserved a red boat

$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7 \wedge$

$\exists R(R \in \text{Reserves} \wedge R.\text{sid} = S.\text{sid}$

$\wedge \exists B(B \in \text{Boats} \wedge B.\text{bid} = R.\text{bid} \wedge B.\text{color} = \text{'red'}))\}$

(TRC)

DRC: Examples

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Sailors

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/96
58	103	11/12/96

Reserves

bid	bname	color
101	interlake	red
103	marine	green

Boats

- Ex3: Find sailors rated > 7 who've reserved a red boat

$$\{S \mid S \in \text{Sailors} \wedge S.\text{rating} > 7 \wedge \\ \exists R(R \in \text{Reserves} \wedge R.\text{sid} = S.\text{sid} \\ \wedge \exists B(B \in \text{Boats} \wedge B.\text{bid} = R.\text{bid} \wedge B.\text{color} = \text{'red'}))\}$$

(TRC)

$$\langle I, N, T, A \rangle \mid \langle I, N, T, A \rangle \in \text{Sailors} \wedge T > 7 \wedge$$

$$\exists Ir, Br, D \langle Ir, Br, D \rangle \in \text{Reserves} \wedge Ir = I \wedge$$

$$\exists B, BN, C \langle B, BN, C \rangle \in \text{Boats} \wedge B = Br \wedge C = \text{'red'} \rangle \rangle \quad \text{(DRC)}$$

DRC: Examples

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Sailors

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/96
58	103	11/12/96

Reserves

bid	bname	color
101	interlake	red
103	marine	green

Boats

- Ex4: Find sailors who've reserved **all** boats

$\{S \mid S \in \text{Sailors} \wedge$
 $\quad \forall B \in \text{Boats} (\exists R \in \text{Reserves} (S.\text{sid} = R.\text{sid} \wedge B.\text{bid} = R.\text{bid})) \}$
(TRC)

DRC: Examples

sid	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Sailors

sid	bid	day
22	101	10/10/96
58	103	11/12/96

Reserves

bid	bname	color
101	interlake	red
103	marine	green

Boats

- Ex4: Find sailors who've reserved **all** boats

$$\{S \mid S \in \text{Sailors} \wedge \forall B \in \text{Boats} (\exists R \in \text{Reserves} (S.\text{sid} = R.\text{sid} \wedge B.\text{bid} = R.\text{bid}))\} \quad (\text{TRC})$$

$$\begin{aligned} & \{ \langle I, N, T, A \rangle \mid \langle I, N, T, A \rangle \in \text{Sailors} \wedge \\ & \quad \forall \langle B, BN, C \rangle \in \text{Boats} \\ & \quad \left(\exists \langle Ir, Br, D \rangle \in \text{Reserves} (I = Ir \wedge Br = B) \right) \} \end{aligned} \quad (\text{DRC})$$