Week 8 Workshop – Query Processing and Optimisation

Housekeeping

- Assignment 2 (Database Theory) for both COMP2400/6240 students:
  - The submission deadline is 23:59, Oct 12, 2021.
  -

Housekeeping

1. Assignment 2 (Database Theory) for both COMP2400/6240 students:
   - The submission deadline is 23:59, Oct 12, 2021.
   -
2. All t
   sa

Housekeeping

1. Assignment 2 (Database Theory) for both COMP2400/6240 students:
   - The submission deadline is 23:59, Oct 12, 2021.
   -

2. All t
   sa
3. Lab 8 is optional (no associated with any assessme
   - We will open a separate sign-up page on Wat
   - All the optional labs will be scheduled from O
   - Three options are available
     (1) Database Programming with Java
     (2) Database Programming with Python
     (3) Database Exercises on IMDB

Query Processing – Example


```
SELECT name FROM Person WHERE age<21;
```

High-level language (SQL)

uage
ebra)

name
Rickon
Bran

Query result

From SQL to RA Expressions

```
Students(matNr, firstName, lastName, email)
Exams(matNr, crsNr, result, semester)
Courses(crsNr, title, unit)
```



```
EXAMS.crsNr=COURSES.crsNr AND result  1.3;
```

From SQL to RA Expressions


```
Students(matNr, firstName, lastName, email)
Exams(matNr, crsNr, result, semester)
Courses(crsNr, title, unit)
```



```
EXAMS.crsNr=COURSES.crsNr AND result  1.3;
```

- **RA Expressions:**
  1. $\pi_{lastName,result,title}(\sigma_{result \leq 1.3}(($Student ms) $\bowtie_{\sigma_{\texttt{Exams}.crsNr=\texttt{Courses}.crsNr}}$ Courses))
  2. $\pi_{lastName,result,title}(\sigma_{result \leq 1.3}(\sigma_{\texttt{EXAMS}.crsNr=\texttt{Courses}.crsNr}(\sigma_{\texttt{Students}.matNr=\texttt{Exams}.matNr}(\texttt{Students} \times \texttt{Exams} \times \texttt{Courses}))))$
  3. $\pi_{lastName,result,title}((\texttt{Students} \bowtie_{\texttt{Students}.matNr=\texttt{Exams}.matNr} (\sigma_{result \leq 1.3}(\texttt{Exams}))) \bowtie_{\texttt{Exams}.crsNr=\texttt{Courses}.crsNr} \texttt{Courses})$
  4. ...

From RA Expressions to Query Trees

- Each RA expression can be represented as a **query tree**:
  - **leaf nodes** represent the input relations;
  - **internal nodes** represent the intermediate result;
  -
- **E**

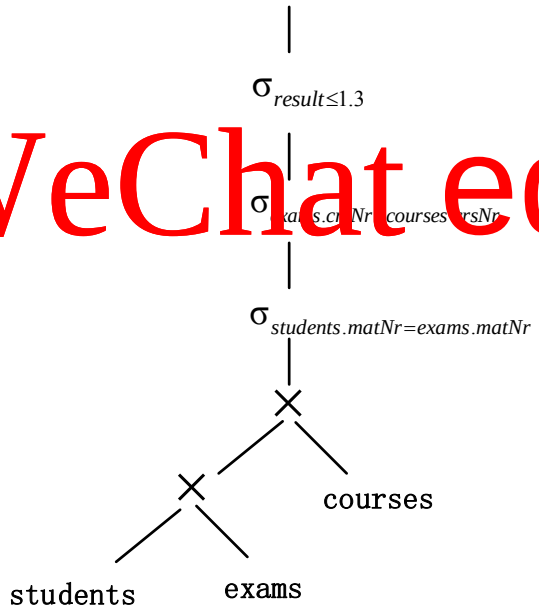
π_{la} $\bowtie$ rs)

Query Tree Example

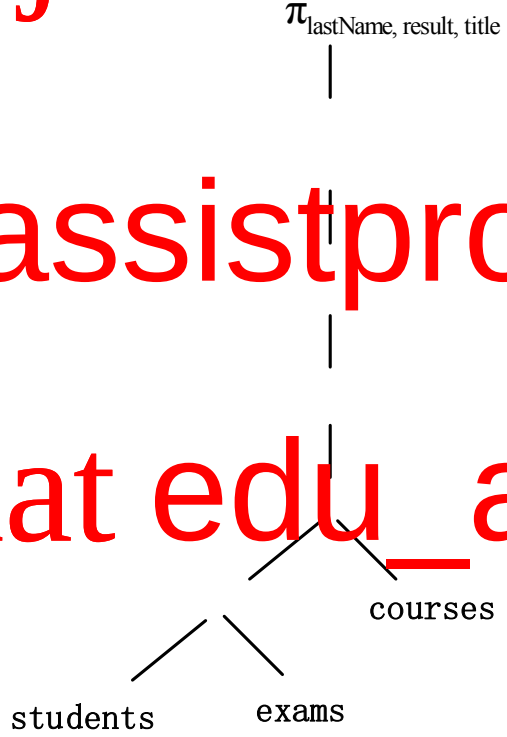
- For each query tree, computation proceeds **bottom-up**:
  - child nodes must be executed before their parent nodes;
  - but there can exist multiple methods of executing sibling nodes.
- **E**

π_{la}

(σ



Equivalent Query Trees (Query Optimisation)



Execution Plan

(Slide 8-27 will not be assessed in our course)

- A **query execution plan** consists of an (extended) query tree with additional annotation at each node indicating:
  - (
  - (

Execution Plan

- **Materialized**: The intermediate result of an operator may be saved in a temporary table for processing by the next operator.
- **Pipelined**: the intermediate result of an operator is directly sent to another operator without creating a temporary table (also called **on-the-fly**).

Note: Pipelined evaluation may have significant saving on I/O cost, while materialized evaluation can avoid repeated computations.

Execution Plan

- **Question:** Which execution plan is "**optimal**" in terms of processing efficiency?

Execution Plan

- **Question:** Which execution plan is "**optimal**" in terms of processing efficiency?

- Th
  (F
- R
  consistently find a plan that is good.

Execution Plan

- **Question:** Which execution plan is "**optimal**" in terms of processing efficiency?

- Th
  (**F**
- R
  consistently find a plan that is good.

- The **performance** of different execution plan considerably (e.g., seconds vs. hours vs. days):
  - different but equivalent **RA expressions**;
  - different algorithms for **each RA operator**.

Execution Plan

- Basic ideas of algorithms used for RA operators

Execution Plan

- Basic ideas of algorithms used for RA operators
  - **Selection:** If there is no index, we have to scan the table. Otherwise, we scan the indexes to retrieve matching tuples and apply remaining

Execution Plan

- Basic ideas of algorithms used for RA operators
  - **Selection:** If there is no index, we have to scan the table. Otherwise, we scan the indexes to retrieve matching tuples and apply remaining

Execution Plan

- Basic ideas of algorithms used for RA operators
  - **Selection:** If there is no index, we have to scan the table. Otherwise, we scan the indexes to retrieve matching tuples and apply remaining

- **Join:** We may use *nested loops join* *s*, *block nested loops join*, etc.

Execution Plan

- Basic ideas of algorithms used for RA operators
  - **Selection:** If there is no index, we have to scan the table. Otherwise, we scan the indexes to retrieve matching tuples and apply remaining

  - **Join:** We may use *nested loops join* *s*, *block nested loops join*, etc.
  - **Group by** and **order by** are typically i

Execution Plan

- Basic ideas of algorithms used for RA operators
  - **Selection:** If there is no index, we have to scan the table. Otherwise, we scan the indexes to retrieve matching tuples and apply remaining

  - **Join:** We may use *nested loops join* *s*, *block nested loops join*, etc.
  - **Group by** and **order by** are typically i
  - **Aggregation operators** use temporary counters in main memory when retrieving tuples.

Execution Plan

- Basic ideas of algorithms used for RA operators
  - **Selection:** If there is no index, we have to scan the table. Otherwise, we scan the indexes to retrieve matching tuples and apply remaining

  - **Join:** We may use *nested loops join* *s*, *block nested loops join*, etc.
  - **Group by** and **order by** are typically i
  - **Aggregation operators** use temporary counters in main memory when retrieving tuples.
  - **Set operators** can use the same approach as projection to eliminate duplicates.

Estimating Query Costs - Example

- Which movies got a non-US award for one of its actors playing an *agent*'?

Estimating Query Costs - Example

- Which movies got a non-US award for one of its actors playing an *'agent'*?

$$\pi_{title,production\_year}(\sigma_{role\_description='agent'}(\text{ROLE} \bowtie \text{ACTOR\_AWARD} \bowtie (\text{AWARD} \sigma_{award\_country='USA}(\text{AWARD}))))$$

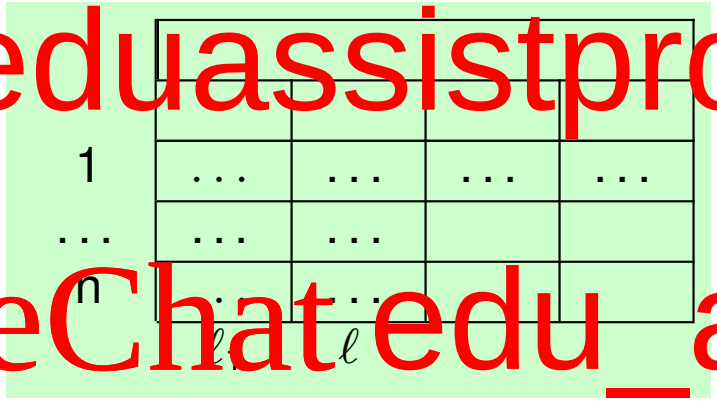
Estimating Query Costs - Example

- Which movies got a non-US award for one of its actors playing an *'agent'*?

$$\pi_{title,production\_year}(\sigma_{role\_description='agent'}(\text{ROLE} \bowtie \text{ACTOR\_AWARD} \bowtie (\text{AWARD} \sigma_{award\_country='USA}(\text{AWARD}))))$$

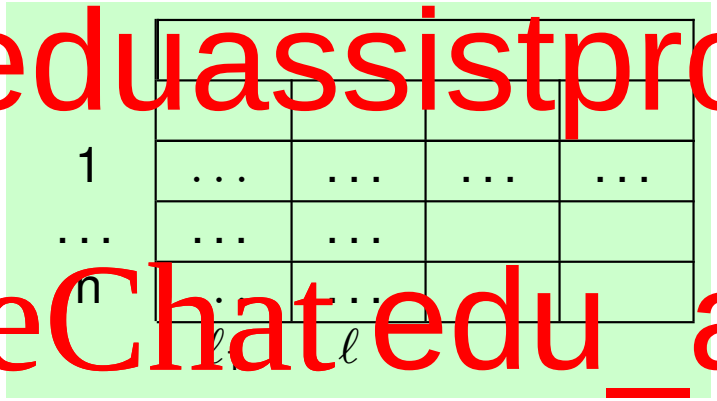
Size of Relations

- How to determine the **size of a relation** r over $R(A_1, \ldots, A_k)$?
  - Let n denote the average number of tuples in r, and ℓ_j the the average



Size of Relations

- How to determine the **size of a relation** r over $R(A_1, \ldots, A_k)$?
  - Let n denote the average number of tuples in r, and ℓ_j the the average



  - Then, $n \cdot \sum_{j=1}^{k} \ell_j$ is the size of the relation r.
- We use this formula to assign sizes to leaf nodes in the query tree.

Estimating Query Costs - Example (Relation Sizes)

- AWARD(Award_name:varchar(30),Institution:varchar(50),Award_country: varchar(20))

Estimating Query Costs - Example (Relation Sizes)

- AWARD(Award_name:varchar(30),Institution:varchar(50),Award_country: varchar(20))

Estimating Query Costs - Example (Relation Sizes)

- AWARD(Award_name:varchar(30),Institution:varchar(50),Award_country: varchar(20))

Estimating Query Costs - Example (Relation Sizes)

- AWARD(Award_name:varchar(30),Institution:varchar(50),Award_country: varchar(20))

- Award_name: $8 \cdot 20 = 160$ bits (t
- Institution: $8 \cdot 30 = 240$ bits (the
- Award_country: $8 \cdot 10 = 80$ bits (

Estimating Query Costs - Example (Relation Sizes)

- AWARD(Award_name:varchar(30),Institution:varchar(50),Award_country: varchar(20))

  - Award_name: $8 \cdot 20 = 160$ bits (t
  - Institution: $8 \cdot 30 = 240$ bits (the
  - Award_country: $8 \cdot 10 = 80$ bits (

- The average size of a tuple is $160 + 80 + 240 = 480$ bits.

Estimating Query Costs - Example (Relation Sizes)

- AWARD(Award_name:varchar(30),Institution:varchar(50),Award_country: varchar(20))

- Award_name: $8 \cdot 20 = 160$ bits (t
- Institution: $8 \cdot 30 = 240$ bits (the
- Award_country: $8 \cdot 10 = 80$ bits (

- The average size of a tuple is $160 + 80 + 240 = 480$ bits.
- The average size of a relation is estimated to be $15 \cdot 480 = 7,200$ bits.

Estimating Query Costs - Example (Relation Sizes)

- ROLE(Id:`char`(4), Title:`varchar`(40), Production_year`number`(4),
  Role_description:`varchar`(100),Credits:`varchar`(40))

Estimating Query Costs - Example (Relation Sizes)

- ROLE(Id:`char`(4), Title:`varchar`(40), Production_year`number`(4), Role_description:`varchar`(100),Credits:`varchar`(40))

Estimating Query Costs - Example (Relation Sizes)

- ROLE(Id:`char`(4), Title:`varchar`(40), Production_year`number`(4),
  Role_description:`varchar`(100),Credits:`varchar`(40))

Estimating Query Costs - Example (Relation Sizes)

- ROLE(Id:`char`(4), Title:`varchar`(40), Production_year:`number`(4),
  Role_description:`varchar`(100),Credits:`varchar`(40))

  - Title: $8 \cdot 25 = 200$ bits (the mean le
  - Production_year: 13 bits (as the dom
  - Role_description: $8 \cdot 50 = 400$
  - Credits: $8 \cdot 20 = 160$ bits (the mean length is 20).

Estimating Query Costs - Example (Relation Sizes)

- ROLE(Id:`char`(4), Title:`varchar`(40), Production_year:`number`(4), Role_description:`varchar`(100),Credits:`varchar`(40))

  - Title: $8 \cdot 25 = 200$ bits (the mean le
  - Production_year: 13 bits (as the dom
  - Role_description: $8 \cdot 50 = 400$
  - Credits: $8 \cdot 20 = 160$ bits (the mean length is 20).

- The average size of a tuple is $64 + 200 + 13 + 400 + 160 = 837$ bits

Estimating Query Costs - Example (Relation Sizes)

- ROLE(Id:`char`(4), Title:`varchar`(40), Production_year:`number`(4), Role_description:`varchar`(100),Credits:`varchar`(40))

  - Title: $8 \cdot 25 = 200$ bits (the mean le
  - Production_year: 13 bits (as the dom
  - Role_description: $8 \cdot 50 = 400$
  - Credits: $8 \cdot 20 = 160$ bits (the mean length is 20).

- The average size of a tuple is $64 + 200 + 13 + 400 + 160 = 837$ bits
- The average size of a relation is to be $500 \cdot 837 = 418,500$ bits

Estimating Query Costs - Example (Relation Sizes)

- ACTOR AWARD(Title:varchar(40),Production_year:number(4),
  Role_description:varchar(100),Award_name:varchar(30),
  Year_of_award:number(4),Category:varchar(100),Result:varchar(20))

  - Production_year: 13 bits (as before);
  - Role_description: 400 bits (as befor
  - Award_name: 160 bits (as before);
  - Year_of_award: 13 bits (as the domai
  - Category: $8 \cdot 40 = 320$ bits (the m
  - Result: $8 \cdot 7 = 56$ bits (the mean length is 7).

- The average size of a tuple is $200 + 13 + 400 + 160 + 13 + 320 + 56 = 1,162$ bits.
- The average size of a relation is $40 \cdot 1162 = 46,480$ bits.

Estimating Query Costs - Example (Query Tree)

Size of Selection Node

- Selection σ_φ is linear in the number n of tuples of the involved relation:

Size of Selection Node

- Selection σ_φ is linear in the number n of tuples of the involved relation:
  - 
  - 
  - 

- Let s be the size of its single relevant node.
- The **size of a selection node** σ_φ is

$$a_\varphi \cdot s\,,$$

where a_φ is the average percentage of tuples satisfying φ.

Estimating Query Costs - Example (Selection)

- For selection $\sigma_{award\_country='USA'}$ assume $a_\varphi = 0.4$ i.e., 40% of the movie awards from the USA). Hence, we have: $a_\varphi \cdot s = 0.4 \cdot 7{,}200 = 2{,}880$.

Size of Difference Node

- Let s_1 and s_2 be the sizes of the two relevant nodes.
- A rel

Size of Difference Node

- Let s_1 and s_2 be the sizes of the two relevant nodes.
- A rel
- The **size of a difference node** is

$$s = (1 -$$

where $(1 - p)$ is the probability that tuples from s_1 does not occur in s_2.

Estimating Query Costs - Example (Difference)

- Since 40% of the movie awards from the USA, the probability of an award to be a US-award is $p = 0.4$. We have: $s_1 \cdot (1 - p) = 7{,}200 \cdot (1\text{-}0.4) = 4{,}320$.

Size of Natural Join Node

- Let s_1 and s_2 be the sizes of the two relevant nodes, and r_1 and r_2 be the

si $\frac{s_1}{\quad}$ $\frac{s_2}{\quad}$ ber of

tu

Size of Natural Join Node

- Let s_1 and s_2 be the sizes of the two relevant nodes, and r_1 and r_2 be the si $\frac{s_1}{\quad}$ $\frac{s_2}{\quad}$ ber of tu

- Th

$$\frac{s_1}{\quad} \cdot p \quad \frac{s_2}{\quad} \cdot (r_1$$

where r is the size of a tuple over the **co**

matching probability (for any tuple of the first relevant node to match with any tuples in the second relevant relation). Note that $r_1 + r_2 - r$ is the size of a tuple after the natural join operation.

Estimating Query Costs - Example (Natural Join)

- For join with ACTOR_AWARD assume $p = 0.08$, i.e., 8% of the actor awards are non-US awards. By $\frac{s_1}{r_1} \cdot p \cdot \frac{s_2}{r_2} \cdot (r_1 + r_2 - r)$, we have:

$$\frac{4}{1} \qquad \frac{\qquad}{\qquad}$$

Estimating Query Costs - Example (Natural Join)

- Assume $p = 0.002$. By $\frac{s_1}{r_1} \cdot p \cdot \frac{s_2}{r_2} \cdot (r_1 + r_2 - r)$, we have:

$$\frac{418,500}{} \qquad \frac{42,682}{} \qquad 9,133.$$

Estimating Query Costs - Example (Selection)

- For selection $\sigma_{\text{role\_description='agent'}}$, assume $a_{\sigma} = 0.05$ (i.e., non-US awards for "agent" roles are 5%). Hence, we have: $a_{\sigma} \cdot s = 0.05 \times 49133 = 2{,}457$.

Size of Projection Node

- Projection $\pi_{A_1,\ldots,A_n}$:
  - Project each tuple to the attributes in $\{A_1, \ldots, A_n\}$
  -

Size of Projection Node

- Projection $\pi_{A_1,\ldots,A_n}$:
  - Project each tuple to the attributes in $\{A_1, \ldots, A_n\}$
  -

- Le lage nu
- The **size of a projection node** $\pi_{A_1,\ldots,A_n}$ i

$$(1 - p_i) \cdot s \quad -$$

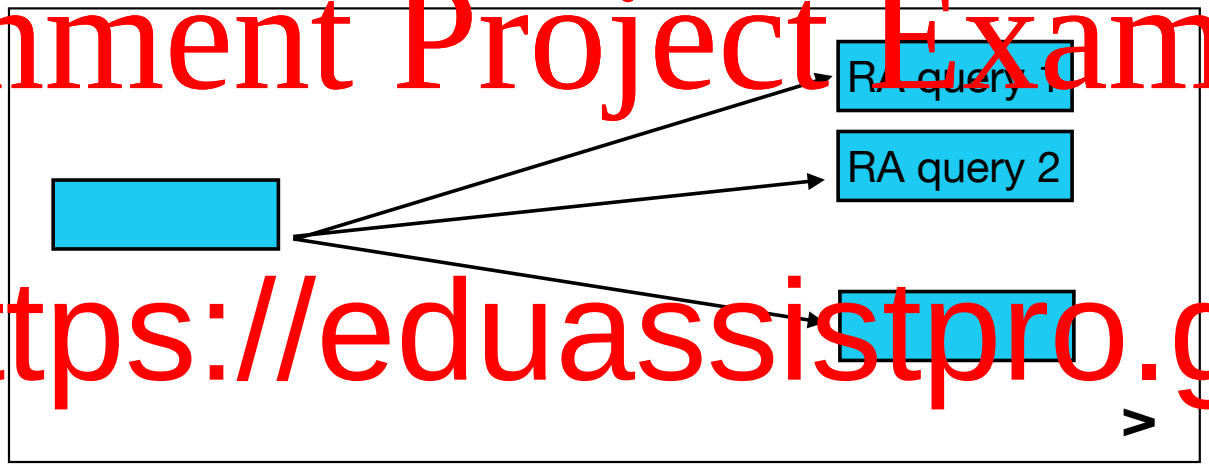
where r_i is the average size of a tuple over $\{A_1, \ldots, A_n\}$, and p_i is the probability that two tuples coincide on $A_1, \ldots, A_n$ (i.e., the same values on all attributes $A_1, \ldots, A_n$).

Estimating Query Costs - Example (Projection)

- For projection $\pi_{\text{title, production\_year}}$ assume that there are 1% of duplicates, i.e., $p_i = 0.01$. By $(1 - p_i) \cdot s \cdot \frac{r_i}{r}$, we have $(1 - 0.01) \cdot 2,457 \frac{213}{1706} = 304$

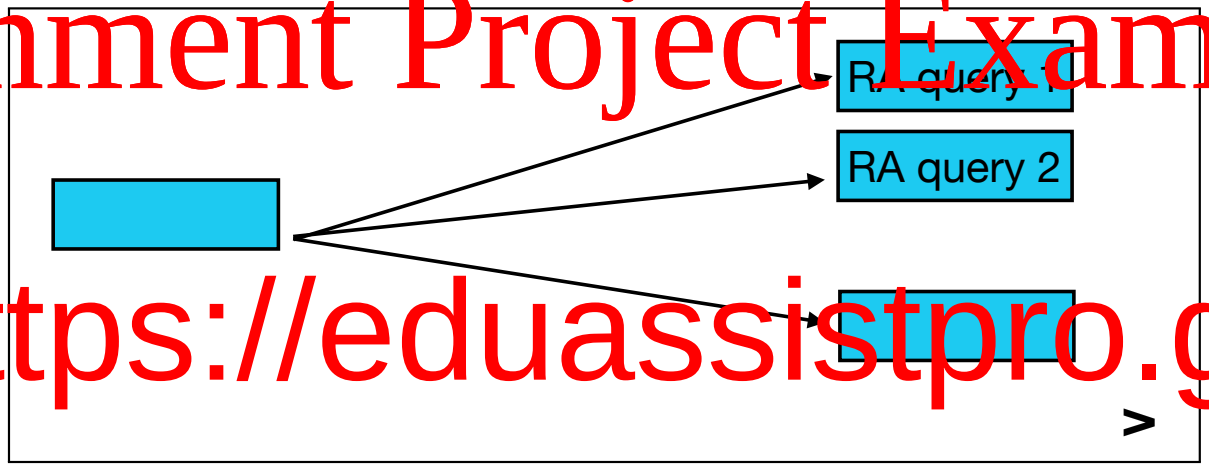
Query Optimisation

Relational Algebra ⇒ Query Optimisation



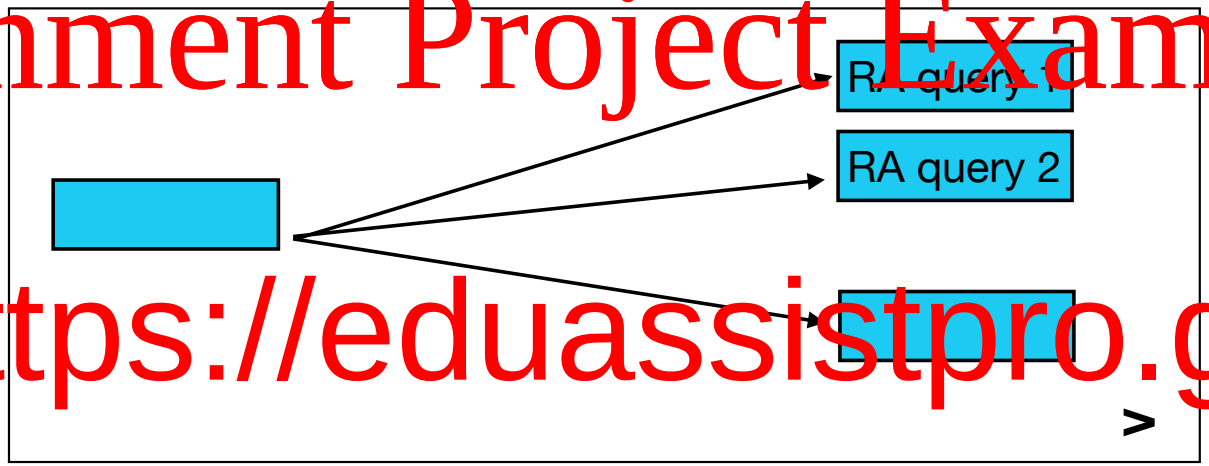
- Which RA query should be chosen for a given SQL q

Relational Algebra ⇒ Query Optimisation



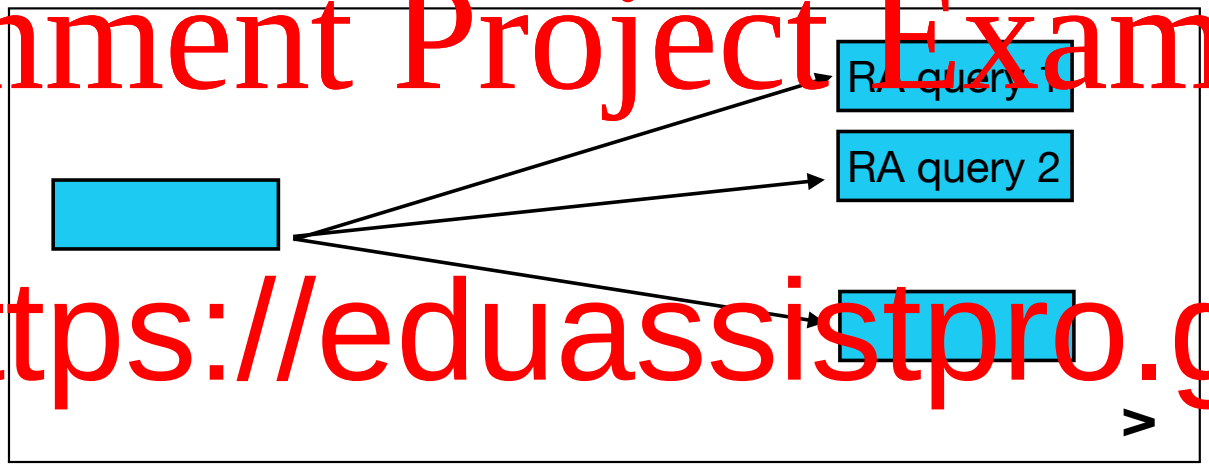
- Which RA query should be chosen for a given SQL q
  - **Who choose?**

Relational Algebra ⇒ Query Optimisation



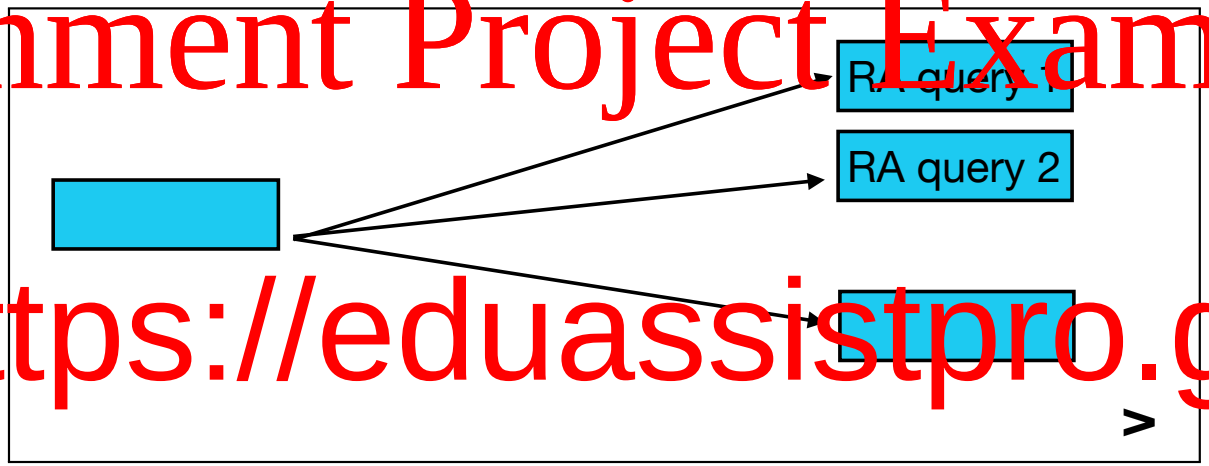
- Which RA query should be chosen for a given SQL q
  - **Who choose?** Query optimiser!

Relational Algebra ⇒ Query Optimisation



- Which RA query should be chosen for a given SQL q
  - **Who choose?** Query optimiser!
  - **How to choose?**

Relational Algebra ⇒ Query Optimisation



- Which RA query should be chosen for a given SQL q
  - **Who choose?** Query optimiser!
  - **How to choose?**
    - Semantic query optimisation
    - Rule-based optimisation
    - Cost-based optimisation

Query Optimisation

- In practice, query optimisers incorporate elements of the following three optimisation approaches:

Query Optimisation

- In practice, query optimisers incorporate elements of the following three optimisation approaches:

Query Optimisation

- In practice, query optimisers incorporate elements of the following three optimisation approaches:

- **Rule-based query optimisation**
  - Use heuristic rules to transform a relational a equivalent one with a possibly lower cost.

Query Optimisation

- In practice, query optimisers incorporate elements of the following three optimisation approaches:

- **Rule-based query optimisation**

  Use heuristic rules to transform a relational a equivalent one with a possibly lower cost.

- **Cost-based query optimisation**

  Use a cost model to estimate the costs of plans, and then select the most cost-effective plan. This will not be assessed in our course.

Semantic Query Optimisation

- **Example:**

  PERSON(id, first_name, last_name, year_born)

- List the ids of the writers who have written movies pro

Semantic Query Optimisation

- **Example:**

  PERSON(id, first_name, last_name, year_born)

- List the ids of the writers who have written movies pro

$\pi_{id}\sigma_{production\_year=2000}(\text{WRITER} \bowtie \text{PERSON}$

$\pi_{id}\sigma_{production\_year=2000}(\text{WRITER} \bowtie \text{PERSON})$

$\pi_{id}\sigma_{production\_year=2000}(\text{WRITER} \bowtie \text{MOVIE})$

Semantic Query Optimisation

- **Example:**

  PERSON(id, first_name, last_name, year_born)

- List the ids of the writers who have written movies pro

  $\pi_{id}\sigma_{production\_year=2000}$(WRITER $\bowtie$ PERSON

  $\pi_{id}\sigma_{production\_year=2000}$(WRITER $\bowtie$ PERSON)

  $\pi_{id}\sigma_{production\_year=2000}$(WRITER $\bowtie$ MOVIE)

- $\pi_{id}\sigma_{production_year=2000}$WRITER $\longleftarrow$ the optimised RA

Rule-based Query Optimisation

- A rule-based optimisation transforms the RA expression by using a set of heuristic rules that typically improve the execution performance.

Rule-based Query Optimisation

- A rule-based optimisation transforms the RA expression by using a set of heuristic rules that typically improve the execution performance.

- K
  w

Rule-based Query Optimisation

- A rule-based optimisation transforms the RA expression by using a set of heuristic rules that typically improve the execution performance.
- K
  w

  - Apply as early as possible to reduce the num

Rule-based Query Optimisation

- A rule-based optimisation transforms the RA expression by using a set of heuristic rules that typically improve the execution performance.
- K
  w
  - Apply as early as possible to reduce the num
  - **Push-down projection:**

    Apply as early as possible to reduce the number of attributes.

Rule-based Query Optimisation

- A rule-based optimisation transforms the RA expression by using a set of heuristic rules that typically improve the execution performance.
- K
  w
  - Apply as early as possible to reduce the num
  - **Push-down projection:**

    Apply as early as possible to reduce the number of attributes.

Rule-based Query Optimisation

- A rule-based optimisation transforms the RA expression by using a set of heuristic rules that typically improve the execution performance.
- **K**
  w
  - Apply as early as possible to reduce the num
  - **Push-down projection:**

    Apply as early as possible to reduce the number of attributes.

- But we must ensure that the resulting query tree gives the same result as the original query tree, i.e., **the equivalence of RA expressions**.

Rule-based Optimisation

- Can they be executed in one go? ↪ Merging RA operators.

Rule-based Optimisation

- Can they be executed in one go? ↪ Merging RA operators.

- $\sigma_{\varphi}(\ _1 \times\ \ _2) \equiv\ \ _1 \bowtie_{\varphi}\ \ _2$
- $\sigma_{\varphi_1}(R_1 \bowtie_{\varphi_2} R_2) \equiv R_2 \bowtie_{\varphi_1 \wedge \varphi_2} R_1$;

Rule-based Optimisation

- Can they be executed in one go? ↪ Merging RA operators

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
  - $\sigma_\varphi(\sigma_\psi(R)) \equiv \sigma_{\varphi \wedge \psi}(R)$;

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
- $\sigma_\varphi(\sigma_\psi(R)) \equiv \sigma_{\varphi \wedge \psi}(R)$;

$\sigma_{CourseNo}$ *STUDY*)

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
- $\sigma_\varphi(\sigma_\psi(R)) \equiv \sigma_{\varphi \wedge \psi}(R)$;

$\sigma_{CourseNo}$ *STUDY*)

UID		
111		
222	COMP2400	115
333	STAT2001	120
111	BUSN2011	110
111	ECON2102	120
333	BUSN2011	130

111	COMP2400	120
111	BUSN2011	
111	ECON2102	

STUDY		
UID	CourseNo	Hours
	0	120

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
- $\sigma_\varphi(\sigma_\psi(R)) \equiv \sigma_{\varphi \wedge \psi}(R)$;

$\sigma_{CourseNo}$ *STUDY*)

UID		
111		
222	COMP2400	115
333	STAT2001	120
111	BUSN2011	110
111	ECON2102	120
333	BUSN2011	130

111	COMP2400	120
111	BUSN2011	
111	ECON2102	

STUDY		
UID	CourseNo	Hours
	0	120

STUDY		
UID	CourseNo	Hours
111	COMP2400	120
222	COMP2400	115
333	STAT2001	120
111	BUSN2011	110
111	ECON2102	120
333	BUSN2011	130

$\Rightarrow$

(without any intermediate relation)

STUDY		
UID	CourseNo	Hours
111	COMP2400	120

Rule-based Optimisation

- Can they be executed in one go? ↪ Merging RA operators

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
- $\pi_X(\pi_Y(R)) \equiv \pi_X(R)$ if $X \subseteq Y$;

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
  - $\pi_X(\pi_Y(R)) \equiv \pi_X(R)$ if $X \subseteq Y$;

$\pi_{UID}(\pi_{UI}$

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
  - $\pi_X(\pi_Y(R)) \equiv \pi_X(R)$ if $X \subseteq Y$;

$\pi_{UID}(\pi_{UI}$

UID		
111		
222	COMP2400	115
333	STAT2001	120
111	BUSN2011	110
111	ECON2102	120
333	BUSN2011	130

333	STAT2001
111	BUSN2
111	ECON2
333	BUSN2

ID
111

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators
- $\pi_X(\pi_Y(R)) \equiv \pi_X(R)$ if $X \subseteq Y$;

$\pi_{UID}(\pi_{UI}$

UID		
111		
222	COMP2400	115
333	STAT2001	120
111	BUSN2011	110
111	ECON2102	120
333	BUSN2011	130

333	STAT2001
111	BUSN2
111	ECON2
333	BUSN2

ID
111

STUDY		
UID	CourseNo	Hours
111	COMP2400	120
222	COMP2400	115
333	STAT2001	120
111	BUSN2011	110
111	ECON2102	120
333	BUSN2011	130

$\Rightarrow$

(without any intermediate relation)

UID
111
222
333

Rule-based Optimisation

- Can they be executed in one go? $\hookrightarrow$ Merging RA operators.

Rule-based Optimisation

- Can they be executed in one go? $\leadsto$ Merging RA operators.
  - $\sigma_\varphi(R_1 \quad R_2) \quad R_1 \bowtie_\varphi R_2$

Rule-based Optimisation

- Can they be executed in one go? $\leadsto$ Merging RA operators.
  - $\sigma_\varphi(R_1 \quad R_2) \quad R_1 \bowtie_\varphi R_2$

Rule-based Optimisation

- Can they be executed in one go? ↪→ Merging RA operators.
  - $\sigma_\varphi(R_1 \quad R_2) \quad R_1 \bowtie_\varphi R_2$

No		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

Rule-based Optimisation

- Can they be executed in one go? ↪→ Merging RA operators.
  - $\sigma_{\varphi}(R_1 \quad R_2) \quad R_1 \bowtie_{\varphi} R_2$

No		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

No	Cr[illegible]	Unit	Student			Us
COMP2400	Relational Databases	6	111			e
COMP2400	Relational Databases	6	222			e
COMP2400	Relational Databases	6	111	COMP2400	2016 S2	active
BUSN2011	Management Accounting	6	111	BUSN2011	2016 S1	active
BUSN2011	Management Accounting	6	222	COMP2400	2016 S1	active
BUSN2011	Management Accounting	6	111	COMP2400	2016 S2	active

Rule-based Optimisation

- Can they be executed in one go? ↪→ Merging RA operators.
  - $\sigma_{\varphi}(R_1 \quad R_2) \quad R_1 \bowtie_{\varphi} R_2$

No		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

No	Cr[illegible]	Unit	Student			us
COMP2400	Relational Databases	6	111			e
COMP2400	Relational Databases	6	222			e
COMP2400	Relational Databases	6	111	COMP2400	2016 S2	active
BUSN2011	Management Accounting	6	111	BUSN2011	2016 S1	active
BUSN2011	Management Accounting	6	222	COMP2400	2016 S1	active
BUSN2011	Management Accounting	6	111	COMP2400	2016 S2	active

Rule-based Optimisation

- Can they be executed in one go? ↪→ Merging RA operators.
  - $\sigma_{\varphi}(R_1 \quad R_2) \quad R_1 \bowtie_{\varphi} R_2$

No		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

Rule-based Optimisation

- Can they be executed in one go? ↪→ Merging RA operators.
  - $\sigma_{\varphi}(R_1 \quad R_2) \quad R_1 \bowtie_{\varphi} R_2$

No		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

Inner Join on Course.No=Enrol.CourseNo (no intermediate)

No	Cname	Unit	StudentID	CourseNo	Semester	Status
COMP2400	Relational Databases	6	222	COMP2400	2016 S1	active
COMP2400	Relational Databases	6	111	COMP2400	2016 S2	active
BUSN2011	Management Accounting	6	111	BUSN2011	2016 S1	active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
  - R_1;
  - tributes
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}($
  attributes not in X, where X_i contain , and ones both in R_1 and R_2;
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, where X_i contains attributes both in R_i and X, and ones both in R_1 and R_2;

Rule-based Optimisation

- Can $\bowtie$ be executed last? $\rightsquigarrow$ Push select/project before join
  - $\sigma_\varphi(R_1 \bowtie R_2) \quad \sigma_\varphi(R_1) \bowtie R_2$, if φ contains only attributes in R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join
  - $\sigma_\varphi(R_1 \bowtie R_2) \quad \sigma_\varphi(R_1) \bowtie R_2$, if φ contains only attributes in R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

			Status
			active
222	COMP2400	2016 S1	active
1			ctive

Rule-based Optimisation

- Can $\bowtie$ be executed last? $\rightsquigarrow$ Push select/project before $\bowtie$
  - $\sigma_\varphi(R_1 \bowtie R_2) \quad \sigma_\varphi(R_1) \bowtie R_2$, if φ contains only attributes in R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

			active
222	COMP2400	2016 S1	active
1			ctive

CourseNo	Cname	Uni			
COMP2400	Relational Databases	6			
COMP2400	Relational Databases	6			
BUSN2011	Management Accounting	6	111	2016 S1	active

Rule-based Optimisation

- Can $\bowtie$ be executed last? $\hookrightarrow$ Push select/project before join
  - $\sigma_\varphi(R_1 \bowtie R_2) \quad \sigma_\varphi(R_1) \bowtie R_2$, if φ contains only attributes in R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

			active
222	COMP2400	2016 S1	active
1			ctive

CourseNo	Cname	Unit			
COMP2400	Relational Databases	6			
COMP2400	Relational Databases	6			
BUSN2011	Management Accounting	6	111	2016 S1	active

CourseNoNo	Cname	Unit	StudentID	Semester	Status
BUSN2011	Management Accounting	6	111	2016 S1	active

Rule-based Optimisation

- Can $\bowtie$ on be executed last? $\hookrightarrow$ Push select/project before $\bowtie$ on R_1;

Rule-based Optimisation

- Can $\bowtie$ on be executed last? $\hookrightarrow$ Push select/project before $\bowtie$ on

- R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join

- R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

COURSE

CourseNo	Cname	Unit
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can $\bowtie$ on be executed last? $\hookrightarrow$ Push select/project before $\bowtie$ on

- R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

COURSE		
CourseNo	Cname	Unit
BUSN2011	Management Accounting	6

StudentID	CourseNo	Semester	Status
			active
			active
			active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join

- R_1;

Course		
COMP2		
BUSN2011	Management Accounting	6

StudentID	CourseNo	Semester	Status
			active
			active
			active

COURSE		
CourseNo	Cname	Unit
BUSN2011	Management Accounting	6

CourseNo	Cname	Unit	StudentID	Semester	Status
BUSN2011	Management Accounting	6	111	2016 S1	active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_{\pi_X}(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_{\pi_X}(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

			ter	Status
			1	active
222	COMP2400		2016 S1	active
				active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

CourseNo	Cname	Unit			
COMP2400	Relational Databases	6			
COMP2400	Relational Databases	6			
BUSN2011	Management Accounting	6	111	2016 S1	active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

CourseNo	Cname	Uni			
COMP2400	Relational Databases	6			
COMP2400	Relational Databases	6			
BUSN2011	Management Accounting	6	111	2016 S1	active

CourseNo	Cname	StudentID
COMP2400	Relational Databases	222
COMP2400	Relational Databases	111
BUSN2011	Management Accounting	111

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.
- $\sigma_{\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)}$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.
- $\sigma_{\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)}$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Rule-based Optimisation

- Can join be executed last? $\rightsquigarrow$ Push select/project before join.
- $\sigma_{\pi_X}(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

$\pi_{CourseNo,Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

$\pi_{CourseNo,Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

$\pi_{CourseNo,Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

π_{Co}

Stu	
111	COMP2400

Rule-based Optimisation

- Can join be executed last? ⇝ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)$, if the join condition involves only attributes in X, how could we derive X_1 and X_2?

Course		
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

$\pi_{CourseNo,Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

π_{Co}

Stu	
111	COMP2400

CourseNo	Cname	StudentID
COMP2400	Relational Databases	222
COMP2400	Relational Databases	111
BUSN2011	Management Accounting	111

Rule-based Optimisation

- Can join be executed last? ↪→ Push select/project before join.

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_{\pi_X}(R_1 \bowtie R_2) \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_{\pi_X}(R_1 \bowtie R_2) \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_{\pi_X}(R_1 \bowtie R_2) \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_{\pi_X(R_1 \bowtie R_2)} \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

CourseNo	Cname	Uni			
COMP2400	Relational Databases	6			
COMP2400	Relational Databases	6			
BUSN2011	Management Accounting	6	111	2016 S1	active

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join.
- $\sigma_X(R_1 \bowtie R_2) \equiv \sigma_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Course		
COMP2		
BUSN2011	Management Accounting	6

		ter	Status
		1	active
222	COMP2400	2016 S1	active
			active

CourseNo	Cname	Uni			
COMP2400	Relational Databases	6			
COMP2400	Relational Databases	6			
BUSN2011	Management Accounting	6	111	2016 S1	active

Cname	StudentID
Relational Databases	222
Relational Databases	111
Management Accounting	111

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join

- es

$_{tip}(Enrol)$?

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join

- es

$_{tip}(Enrol)$?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join

- es

$_{tip}(Enrol)$?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

π_{Cname} COURSE

Cname
Relational
Management

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join

- es

$_{tID}(Enrol)$?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

StudentID	CourseNo	Semester	Status
			active
			active
			active

π_{Cname} COURSE

Cname
Relational
Management

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join

- es

$\pi_{tID}(Enrol)$?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

StudentID	CourseNo	Semester	Status
			active
			active
			active

π_{Cname} COURSE

Cname
Relational
Management

π_{St}

111
222

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join

- es

$_{tip}(Enrol)$?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

StudentID	CourseNo	Semester	Status
			active
			active
			active

π_{Cname} COURSE
Cname
Relational
Management

π_{St}
111
222

Is $\pi_{Cname}(Course) \bowtie \pi_{StudentID}(Enrol)$ our desired result?

Rule-based Optimisation

- Can join be executed last? ↪ Push select/project before join

- es

$\pi_{tid}(Enrol)$?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

StudentID	CourseNo	Semester	Status
			active
			active
			active

π_{Cname} COURSE
Cname
Relational
Management

π_{St}
111
222

Is $\pi_{Cname}(Course) \bowtie \pi_{StudentID}(Enrol)$ our desired result?

No. $\pi_{Cname,StudentID}(Course \bowtie Enrol) \neq \pi_{Cname}(Course) \bowtie \pi_{StudentID}(Enrol)$

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

nrol))

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

$\pi_{CourseNo,Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

nrol))

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
111	BUSN2011	2016 S1	active
			active
			active

$\pi_{CourseNo, Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
111	BUSN2011	2016 S1	active
			active
			active

$\pi_{CourseNo,Cname}$ COURSE	
CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

π_{Co}	
Stu	
222	COMP2400
111	COMP2400

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

nrol))

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

		ter	Status
111	BUSN2011	2016 S1	active
			active
			active

$\pi_{CourseNo, Cname}$ COURSE

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

π_{Co}

Stu	
222	COMP2400
111	COMP2400

CourseNo	Cname	StudentID
COMP2400	Relational Databases	222
COMP2400	Relational Databases	111
BUSN2011	Management Accounting	111

Rule-based Optimisation

- Can join be executed last? $\hookrightarrow$ Push select/project before join.
- $\pi_X(R_1 \bowtie R_2) = \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2))$, if the join condition involves attributes outside X, how could we derive X_1 and X_2?

nrol))

CourseNo	Cname	Unit
COMP2400	Relational Databases	6
BUSN2011	Management Accounting	6

111	BUSN2011	2016 S1	active
			active
			active

$\pi_{CourseNo, Cname}$(COURSE)

CourseNo	Cname
COMP2400	Relational Databases
BUSN2011	Management Accounting

π_{Co}

Stu	
222	COMP2400
111	COMP2400

CourseNo	Cname	StudentID
COMP2400	Relational Databases	222
COMP2400	Relational Databases	111
BUSN2011	Management Accounting	111

Cname	StudentID
Relational Databases	222
Relational Databases	111
Management Accounting	111

Heuristic Rules and Query Trees

(1) $\sigma_{\varphi}(\sigma_{\psi}(R)) \equiv \sigma_{\varphi \wedge \psi}(R)$

Heuristic Rules and Query Trees

(1) $\sigma_{\varphi}(\sigma_{\psi}(R)) \equiv \sigma_{\varphi \wedge \psi}(R)$;

(2) $\pi_X(\pi_Y(R)) \equiv \pi_X(R)$ if $X \subseteq Y$;

Heuristic Rules

(3) $\sigma_{\varphi}(R_1 \times R_2) \equiv R_1 \bowtie_{\varphi} R_2$

Heuristic Rules

(3) $\sigma_{\varphi}(R_1 \times R_2) \equiv R_1 \bowtie_{\varphi} R_2$

(4) $\sigma_{\varphi_1}(R_1 \bowtie_{\varphi_2} R_2) \equiv R_2 \bowtie_{\varphi_1 \wedge \varphi_2} R_1$

Push-down Selection – Example

- Given the relation schemas:

  PERSON(id, first_name, last_name, year_born)

  MOVIE(title, production_year, country, run_time, major_genre)

  R

- Q

π_{tit}

$first\_name = \text{'}Tom\text{'} \wedge last\_name = \text{'}Cruise\text{'}$(MOVIE $\times$ (PE

Push-down Selection – Example

- Given the relation schemas:
  PERSON(id, first_name, last_name, year_born)
  MOVIE(title, production_year, country, run_time, major_genre)
  R
- Q
  π_{tit}
  $first\_name='Tom' \wedge last\_name='Cruise'$(MOVIE $\times$ (PE
- **Question:** Can we apply the following rule to optimi
  $\sigma_{\varphi_1 \wedge \varphi_2}(R_1 \times R_2) \equiv \sigma_{\varphi_1}(R_1) \times \sigma_{\varphi_2}(R_2)$ if
  and φ_2 contains only attributes in R_2

Push-down Selection – Example

- Given the relation schemas:
  PERSON(id, first_name, last_name, year_born)
  MOVIE(title, production_year, country, run_time, major_genre)
  ROLE(id, mid, role)
- Query: https://eduassistpro.github.i
  π_{tit}
  $_{first\_name=\text{'Tom'} \wedge last\_name=\text{'Cruise'}}$(MOVIE $\times$ (PE
- **Question:** Can we apply the following rule to optimi
  $\sigma_{\varphi_1 \wedge \varphi_2}(R_1 \times R_2) \equiv \sigma_{\varphi_1}(R_1) \times \sigma_{\varphi_2}(R_2)$ if φ_1 contains only attributes in R_1 and φ_2 contains only attributes in R_2
- We would have

$$\pi_{title,production\_year}(\sigma_{title=mtitle \wedge production\_year=mprod\_year}(\sigma_{major\_genre=\text{'war'}}(\text{MOVIE}) \times \sigma_{first\_name=\text{'Tom'} \wedge last\_name=\text{'Cruise'}}(\text{PERSON} \bowtie \text{ROLE})))$$

Push-down Selection – Example

Push-down Selection – Example

- Given the relation schemas:
  PERSON(id, first_name, last_name, year_born)
  MOVIE
  R
- Q

$\pi_{title,production\_year}(\sigma_{title=mtitle \wedge production\_year=mpr}$ (σ (MOVIE)

$\times \sigma_{first\_name='Tom' \wedge last\_name='Cruise'}$(PERSON ▷

Push-down Selection – Example

- Given the relation schemas:
  PERSON(id, first_name, last_name, year_born)
  MOVIE
  R
- Q

$\pi_{title,production\_year}(\sigma_{title=mtitle \wedge production\_year=mpr}$ (MOVIE)

$\times \sigma_{first\_name='Tom' \wedge last\_name='Cruise'}$ (PERSON ▷

- Can we apply $\sigma_\varphi(R_1 \times R_2) = R_1 \bowtie_\varphi R_2$?

Push-down Selection – Example

- Given the relation schemas:
  PERSON(id, first_name, last_name, year_born)
  MOVIE
  R
- Q

$\pi_{title,production\_year}(\sigma_{title=mtitle \wedge production\_year=mpr}$ $(\sigma$ (MOVIE)

$\times \sigma_{first\_name='Tom' \wedge last\_name='Cruise}$ (PERSON $\triangleright$

- Can we apply $\sigma_\varphi(R_1 \times R_2) = R_1 \bowtie_\varphi R_2$?
- We would have

$\pi_{title,production\_year}(\sigma_{major\_genre='war'}(\text{MOVIE}) \bowtie_{title=mtitle \wedge production\_year=mprod\_year}$ (

$\sigma_{first\_name='Tom' \wedge last\_name='Cruise'}(\text{PERSON} \bowtie \text{ROLE})))$

Push-down Selection – Example

Push-down Projection – Example

- Given the relation schemas:

  PERSON(id, first_name, last_name, year_born)

  M

  R

- Q

$$\pi_{title,production\_year}(\sigma_{major\_genre='war'}(\text{MOVIE}) \triangleright \quad r\ ($$

$$\sigma_{first\_name='Tom' \wedge last\_name='Cruise'}(\text{PERSON} \bowtie$$

Push-down Projection – Example

- Given the relation schemas:

  PERSON(id, first_name, last_name, year_born)

  M

  R

- **Q**

$$\pi_{title,production\_year}(\sigma_{major\_genre='war'}(\text{MOVIE}) \triangleright \quad r\ ($$

$$\sigma_{first\_name='Tom' \wedge last\_name='Cruise'}(\text{PERSON}) \bowtie$$

- **Question**: Can we apply the following rule to optimi

$$\pi_X(R_1 \bowtie R_2) \equiv \pi_X(\pi_{X_1}(R_1) \bowtie \pi_{X_2}(R_2)),$$

where X_i contains attributes both in R_i and X, and ones both in R_1 and R_2

Push-down Projection – Example

- Given the relation schemas:

  PERSON(id, first_name, last_name, year_born)

  MOVIE(title, production_year, country, run_time, major_genre)

  R

- Q

$\pi_{title,production\_year}(\sigma_{major\_genre='war'}(\quad \text{OVIE}) \bowtie_{title=mtitle \quad production\_year=mprod\_year}($

$\sigma_{first\_name='Tom' \wedge last\_name='Cruise'}(\text{PERSON} \bowtie$

Push-down Projection – Example

- Given the relation schemas:

  PERSON(id, first_name, last_name, year_born)

  MOVIE(title, production_year, country, run_time, major_genre)

  ROLE(id, mtitle, mprod_year, role)

- Query:

$$\pi_{title,production\_year}(\sigma_{major\_genre='war'}(\text{MOVIE}) \bowtie_{title=mtitle \wedge production\_year=mprod\_year} (\sigma_{first\_name='Tom' \wedge last\_name='Cruise'}(\text{PERSON} \bowtie \text{ROLE})))$$

- We would have:

$$\pi_{title,production\_year}(\pi_{title,production\_year}(\sigma_{major\_genre='war'} \bowtie_{title=mtitle \wedge production\_year=mprod\_year} (\pi_{mtitle,mprod\_year}(\sigma_{first\_name='Tom' \wedge last\_name='Cruise'}(\text{PERSON} \bowtie \text{ROLE})))))$$

We further apply some rules to optimise the query ...

Push-down Projection – Example

Cost-based Optimisation (not assessed)

- Consider CHARTS={Rank, Artist, Song} with 100 tuples and 3 attributes

Rank	Artist	Song
1	Chingy	Right Thurr
6	Brooke Fraser	
7	Black Eyed Peas	
...	...	
...	...	

- Compare two strategies of evaluating “Who is top of the pops?”:
  - $\sigma_{\text{Rank}=1}(\pi_{\text{Rank, Artist}}(\text{CHARTS}))$
  - $\pi_{\text{Rank, Artist}}(\sigma_{\text{Rank}=1}(\text{CHARTS}))$

Cost-based Optimisation (not assessed)

- Consider CHARTS={Rank, Artist, Song} with 100 tuples and 3 attributes

Rank	Artist	Song
1	Chingy	Right Thurr
6	Brooke Fraser	
7	Black Eyed Peas	
...	...	
...	...	

- Compare two strategies of evaluating “Who is top of the pops?”:
  - $\sigma_{\text{Rank}=1}(\pi_{\text{Rank, Artist}}(\text{CHARTS}))$
  - $\pi_{\text{Rank, Artist}}(\sigma_{\text{Rank}=1}(\text{CHARTS}))$

  Selection before Projection is preferred.

Cost-based Optimisation (not assessed)

- Consider CHARTS={Rank, Artist, ...} with 100 tuples and 50 attributes:

Rank	Artist	Song	...	...	...
				.	...
				.	...
				.	...
				.	...
5	Justin Timberlake	Senorita	...	...	...
6	Brooke Fraser				
7	Black Eyed Peas	Wher			
...	...				

- Compare two strategies of evaluating?
  - $\sigma_{\text{Rank} > 10}(\pi_{\text{Rank, Artist}}(\text{CHARTS}))$
  - $\pi_{\text{Rank, Artist}}(\sigma_{\text{Rank} > 10}(\text{CHARTS}))$

Cost-based Optimisation (not assessed)

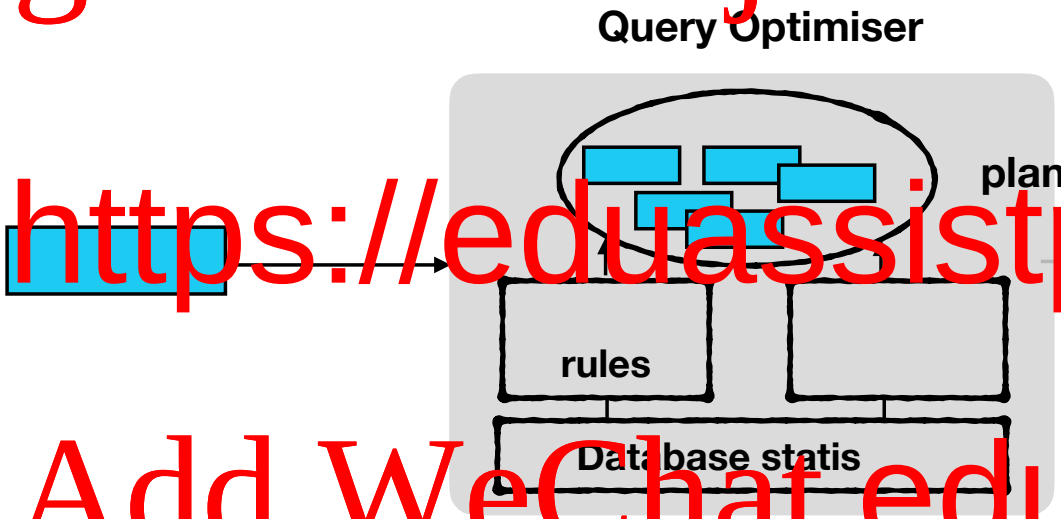
- Consider CHARTS={Rank, Artist, ...} with 100 tuples and 50 attributes:

Rank	Artist	Song	...	...	...
				.	...
				.	...
				.	...
				.	...
5	Justin Timberlake	Senorita	...	...	...
6	Brooke Fraser				
7	Black Eyed Peas	Wher			
...	...				

- Compare two strategies of evaluating?
  - $\sigma_{\text{Rank} > 10}(\pi_{\text{Rank, Artist}}(\text{CHARTS}))$
  - $\pi_{\text{Rank, Artist}}(\sigma_{\text{Rank} > 10}(\text{CHARTS}))$

  Projection before Selection is preferred.

Query Optimisation



- Trade-off:

  Time for executing a RA query vs **Time for finding a better RA query**

(credit cookie) memorising vs understanding