Rename Operation

- Enables the renaming of relations and/or attributes
- Sometimes necessary to use the same relation or the same attribute several times in a query
- The rename operation only changes names but neither the attribute domains nor the tuples of a relation
- Variations
  - $\rho_S(R)$: relation R is renamed into relation S
  - $\rho_{S(B_1, \ldots, B_n)}(R)$: relation $R(A_1, \ldots, A_n)$ is renamed into relation $S(B_1, \ldots, B_n)$
  - $\rho_{B \leftarrow A}(R)$: attribute A of relation R is renamed into B

Example Queries (I)

- Query 1: Which students have enrolled for more than 10 semesters?
- Relational Algebra (RA) expression: $\sigma_{sem > 10}$(students)

$\sigma_{sem > 10}$(students)		
reg-id	name	sem
24002	Xenokrates	18
25403	Jonas	12

- Query 2: Which ranks do the professors have?
- RA expression: π_{rank}(professors)

π_{rank}(professors)
rank
C4
C3

No duplicates due to set property of relations!

Example Queries (II)

- Query 3: List the personnel ids and names of all professors and assistants
- RA expression: $\pi_{\text{pers-id, name}}(\text{professors}) \cup \pi_{\text{pers-id, name}}(\text{assistants})$ [= R]

R	
pers-id	name
2125	Sokrates
2126	Russel
2127	Kopernikus
2133	Popper
2134	Augustinus
2136	Curie
2137	Kant
3002	Platon
3003	Aristoteles
3004	Wittgenstein
3005	Rhetikus
3006	Newton
3007	Spinoza

- Query 4: Determine all students who have so far not passed any exam
- RA expression: $\pi_{\text{reg-id}}(\text{students}) - \pi_{\text{reg-id}}(\text{tests})$ [= R]

R
reg-id
24002
26120
26830
29120
29555

Example Queries (III)

- Query 5: Connect all *professor* tuples with all *attends* tuples
- This query is not meaningful but demonstrates the Cartesian product operation
- RA expression: professors $\times$ attends

professors $\times$ attends					
pers-id	name	rank	room	reg-id	id
2125	Sokrates	C4	226	26120	5001
...	...	...	...	...	...
2137	Kant	C4	007	29555	5001

- Cartesian product operation is an expensive operation
- The query processor of a relational database system aims to avoid the execution of Cartesian products by sophisticated join techniques

Example Queries (IV)

- Query 6: What are the predecessors of the predecessors of lecture 5216?

- RA expression (with dot notation):

$$\pi_{T1.predecessor}(\sigma_{T1.successor=T2.predecessor \wedge T2.successor=5216}(\rho_{T1}(is\_precondition\_of) \times \rho_{T2}(is\_precondition\_of)))$$

- Alternative RA expression (according to previously defined syntax):

$$\pi_{predecessorT1}(\sigma_{successorT1 = predecessorT2 \wedge successorT2 = 5216}(\rho_{T1(predecessorT1,\ successorT1)}(is\_precondition\_of) \times \rho_{T2(predecessorT2,\ successorT2)}(is\_precondition\_of)))$$

- Result: single tuple with a single attribute with value 5001

- Transitive closure cannot be computed by the Relational Algebra in the sense of “Determine *all* predecessors of lecture 5216”

Example Queries (V)

T1		T2	
predecessor	successor	predecessor	successor
5001	5041	5001	5041
5001	5041	5001	5043
5001	5041	5001	5049
5001	5041	5041	5216
5001	5041	5043	5052
5001	5041	5041	5052
5001	5041	5052	5259
5001	5043	5001	5041
5001	5043	5001	5043
5001	5043	5001	5049
5001	5043	5041	5216
5001	5043	5043	5052
5001	5043	5041	5052
5001	5043	5052	5259
5001	5049	5001	5041
5001	5049	5001	5043
5001	5049	5001	5049
5001	5049	5041	5216
5001	5049	5043	5052
5001	5049	5041	5052
5001	5049	5052	5259
5041	5216	5001	5041
5041	5216	5001	5043
5041	5216	5001	5049
5041	5216	5041	5216
5041	5216	5043	5052
5041	5216	5041	5052
5041	5216	5052	5259

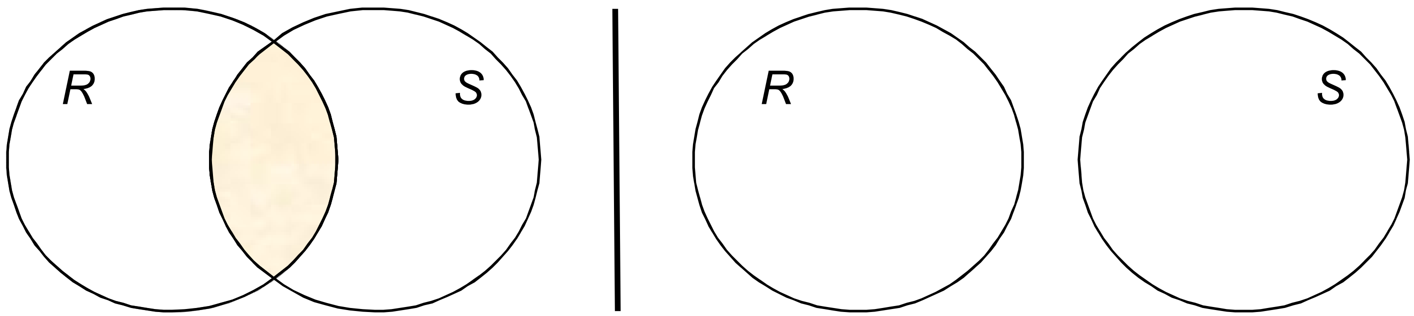
T1		T2	
predecessor	successor	predecessor	successor
5043	5052	5001	5041
5043	5052	5001	5043
5043	5052	5001	5049
5043	5052	5041	5216
5043	5052	5043	5052
5043	5052	5041	5052
5043	5052	5052	5259
5041	5052	5001	5041
5041	5052	5001	5043
5041	5052	5001	5049
5041	5052	5041	5216
5041	5052	5043	5052
5041	5052	5041	5052
5041	5052	5052	5259
5052	5259	5001	5041
5052	5259	5001	5043
5052	5259	5001	5049
5052	5259	5041	5216
5052	5259	5043	5052
5052	5259	5041	5052
5052	5259	5052	5259

Overview of Derived Relational Algebra Operations

- The term *derived* means that these operations can be formally defined as algebraic expressions on the basis of the 5(+1) basic Relational Algebra operations
- The algebraic expressions are *compositions* (or: combinations) of basic Relational Algebra operations
- The Relational Algebra comprises the following derived operations:
  - Intersection ($\cap$)
  - Symmetrical difference (Δ)
  - Quotient (division) ($\div$)
  - Natural join ($\bowtie$)
  - Theta join ($\bowtie_F$)
  - Left outer join (⟕), right outer join (⟖), (full) outer join (⟗)
  - Semijoin ($\ltimes$, $\rtimes$)
  - Antijoin ($\triangleright$)

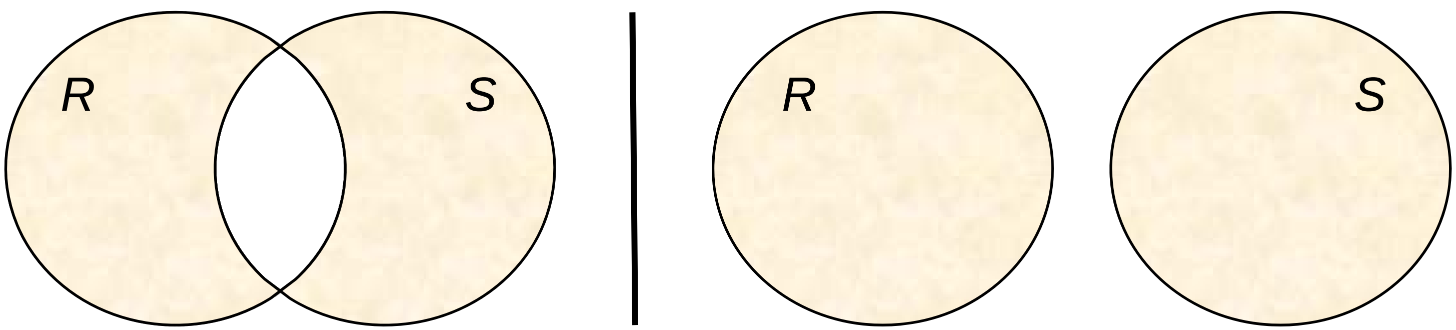
Intersection Operation

- At the schema level
  - Precondition is that *R* and *S* are schema compliant
  - Result schema is equal to *R* (or *S*), that is, there is no change
- At the data level
  - $R \cap S = R - (R - S) = \{t \mid t \in R \wedge t \in S\}$
  - No duplicates in result due to set property
  - Number of tuples of $R \cap S$ is $|R \cap S| = |\{t \mid t \in R \wedge t \in S\}|$
  - Number of tuples of $R \cap S$ is bounded by $0 \leq |R \cap S| \leq \min(|R|, |S|)$



Symmetrical Difference Operation

- At the schema level
  - Precondition is that *R* and *S* are schema compliant
  - Result schema is equal to *R* (or *S*), that is, there is no change
- At the data level
  - $R \Delta S = (R - S) \cup (S - R)$
  - No duplicates in result due to set property
  - Union of those tuples of *R* and *S* that are only contained in one of both relations
  - Number of tuples of $R \Delta S$ is $|R \Delta S| = |R| + |S| - 2 \cdot |R \cap S|$



Quotient (Division) Operation (I)

Common attributes

Remaining attributes

R	A	B	C	D
	a	b	c	d
	a	b	e	f
	b	c	e	f
	e	d	c	d
	e	d	e	f
	b	c	c	e

S	C	D
	c	d
	e	f

$R \div S$	A	B
	a	b
	e	d

- Used in queries that contain an *all* (universal) quantification
- At the schema level
  - $\mathcal{R}(A_1, \ldots, A_r)$ schema of a relation R, $\mathcal{S}(B_1, \ldots, B_s)$ schema of a relation S
  - $\mathcal{S} \subseteq \mathcal{R}$
  - Result relation has schema $\mathcal{R} - \mathcal{S}$