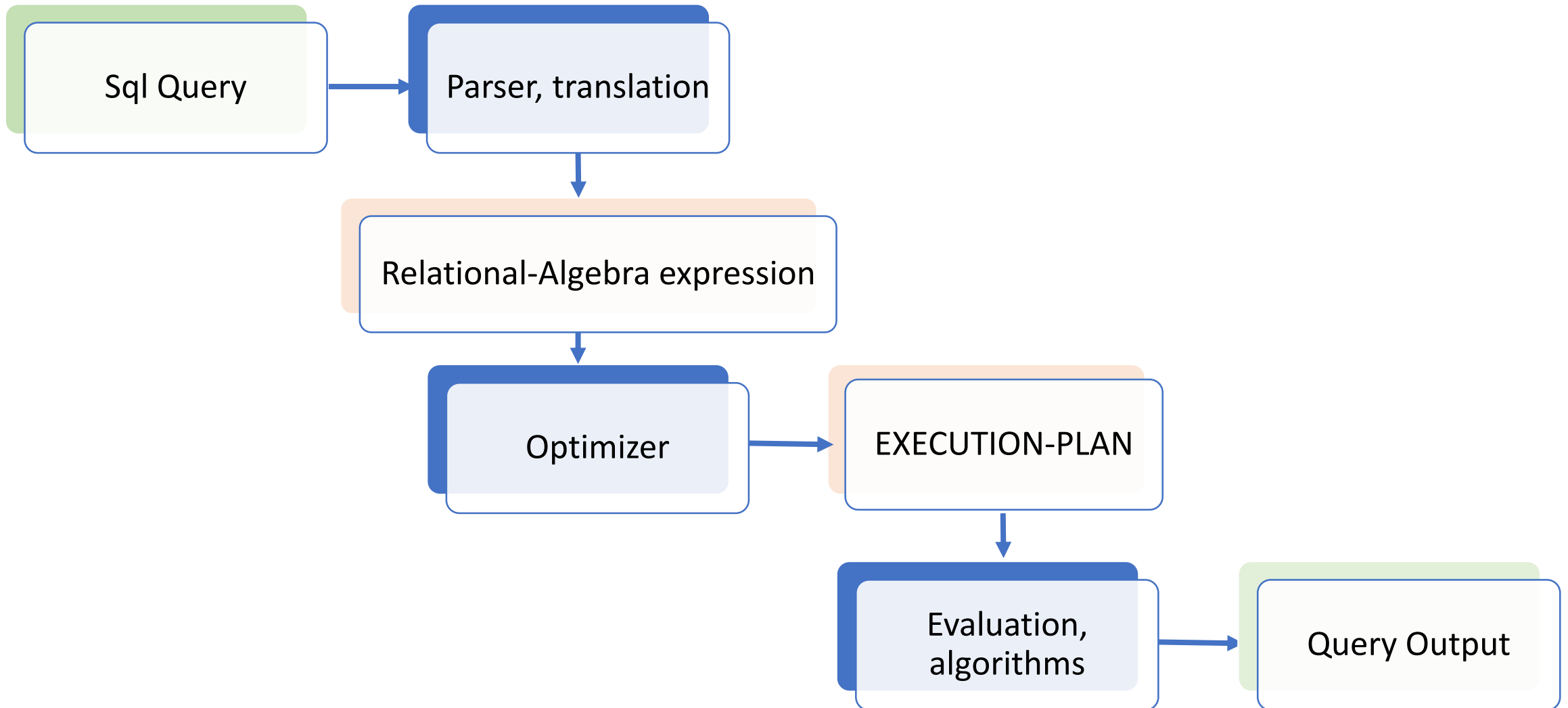
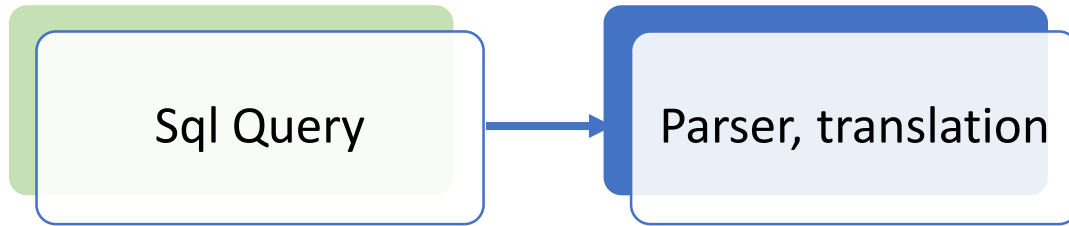


Query Optimization

COURSE 6: Databases

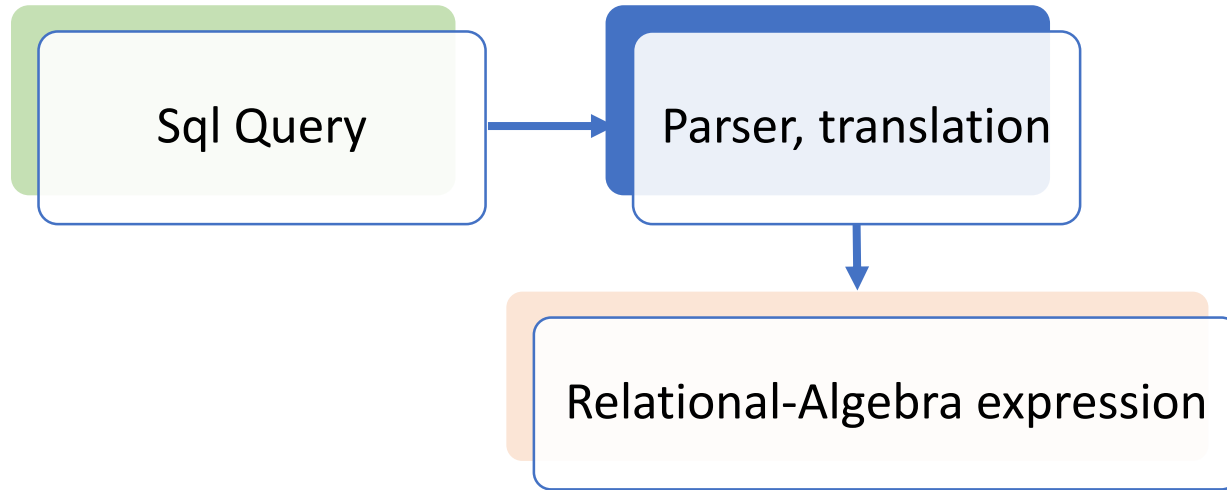
Query execution





```
select p1.prod_name, p2.prod_name, p1.prod_min_price  
from products p1 join products p2  
on p1.prod_min_price = p2.prod_min_price
```

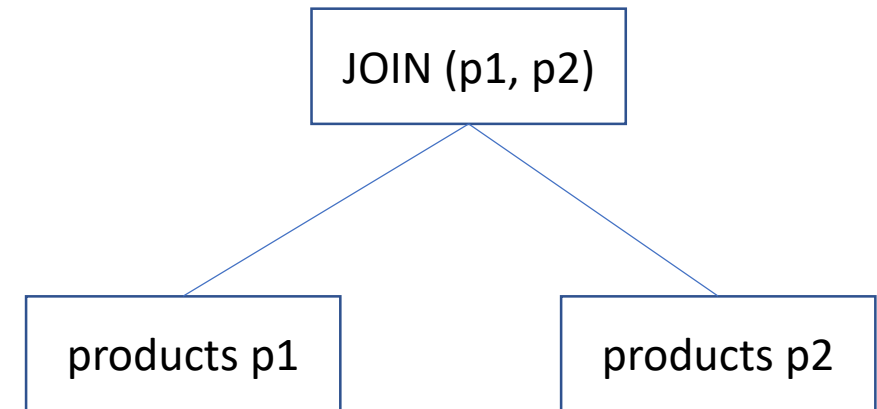
check syntax, table names, column names

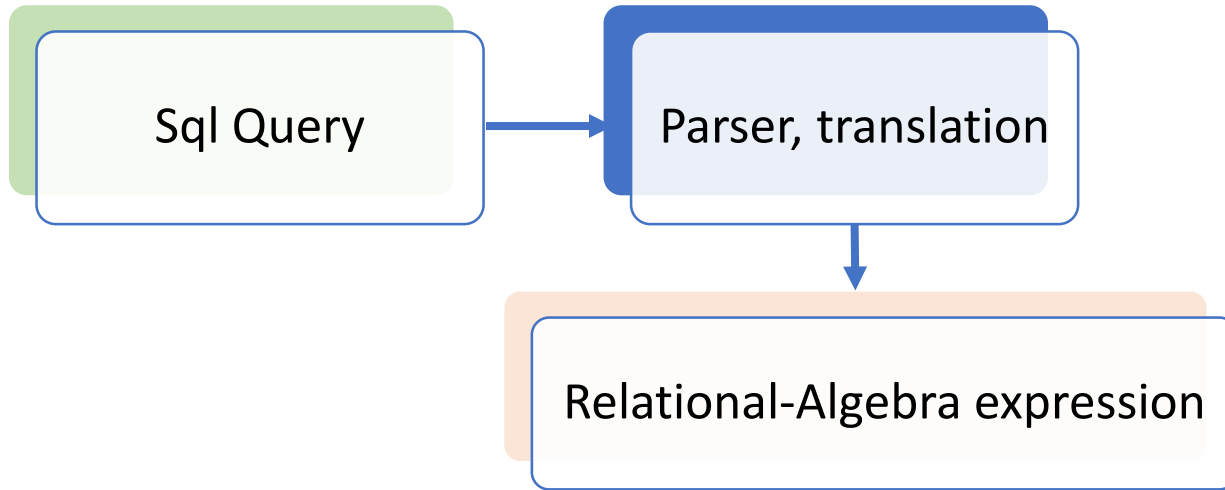


```
select p1.prod_name, p2.prod_name, p1.prod_min_price  
from products p1 join products p2  
on p1.prod_min_price = p2.prod_min_price
```

relations + operators

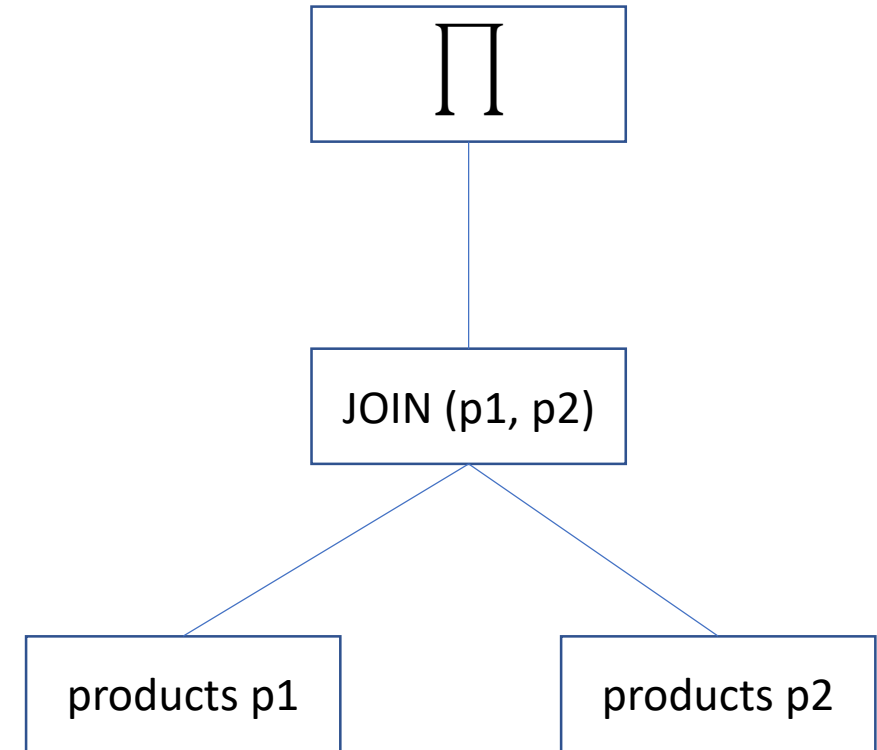
$JOIN(p1, p2)$

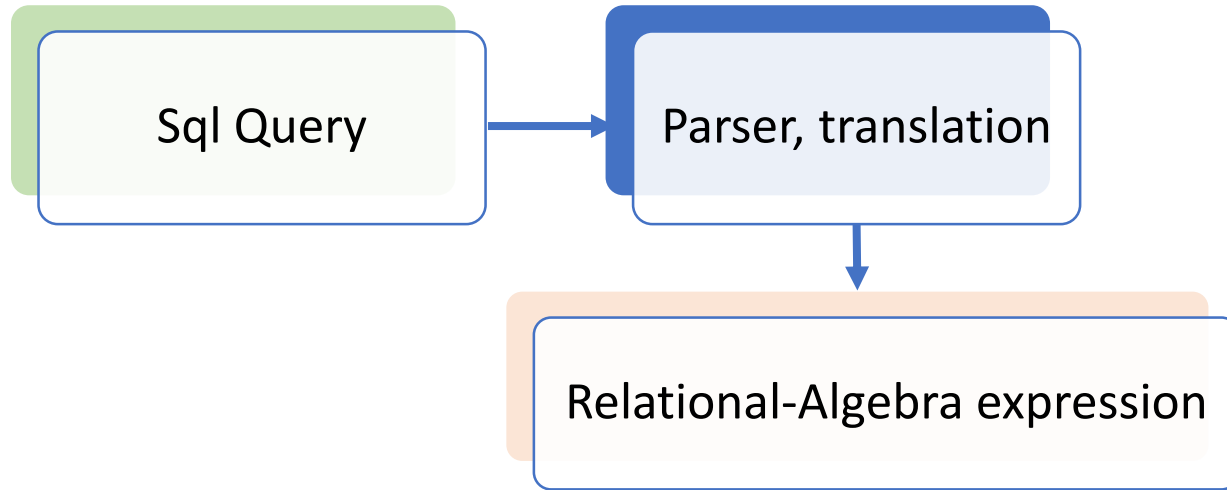




```

select p1.prod_name, p2.prod_name, p1.prod_min_price
from products p1 join products p2
on p1.prod_min_price = p2.prod_min_price
  
```

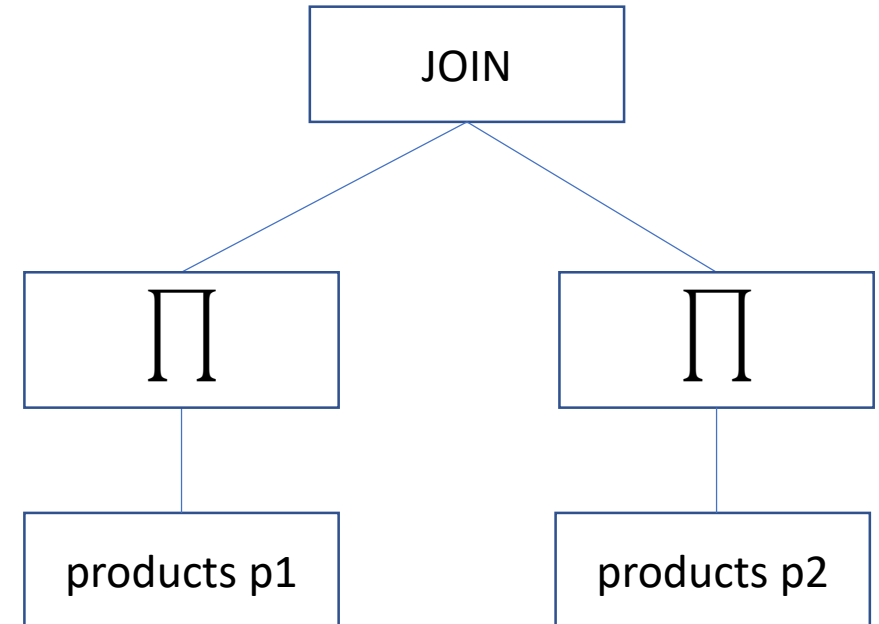
$$\Pi_{p1.name, p1.minprice, p2.name, p2.price} JOIN(p1, p2)$$


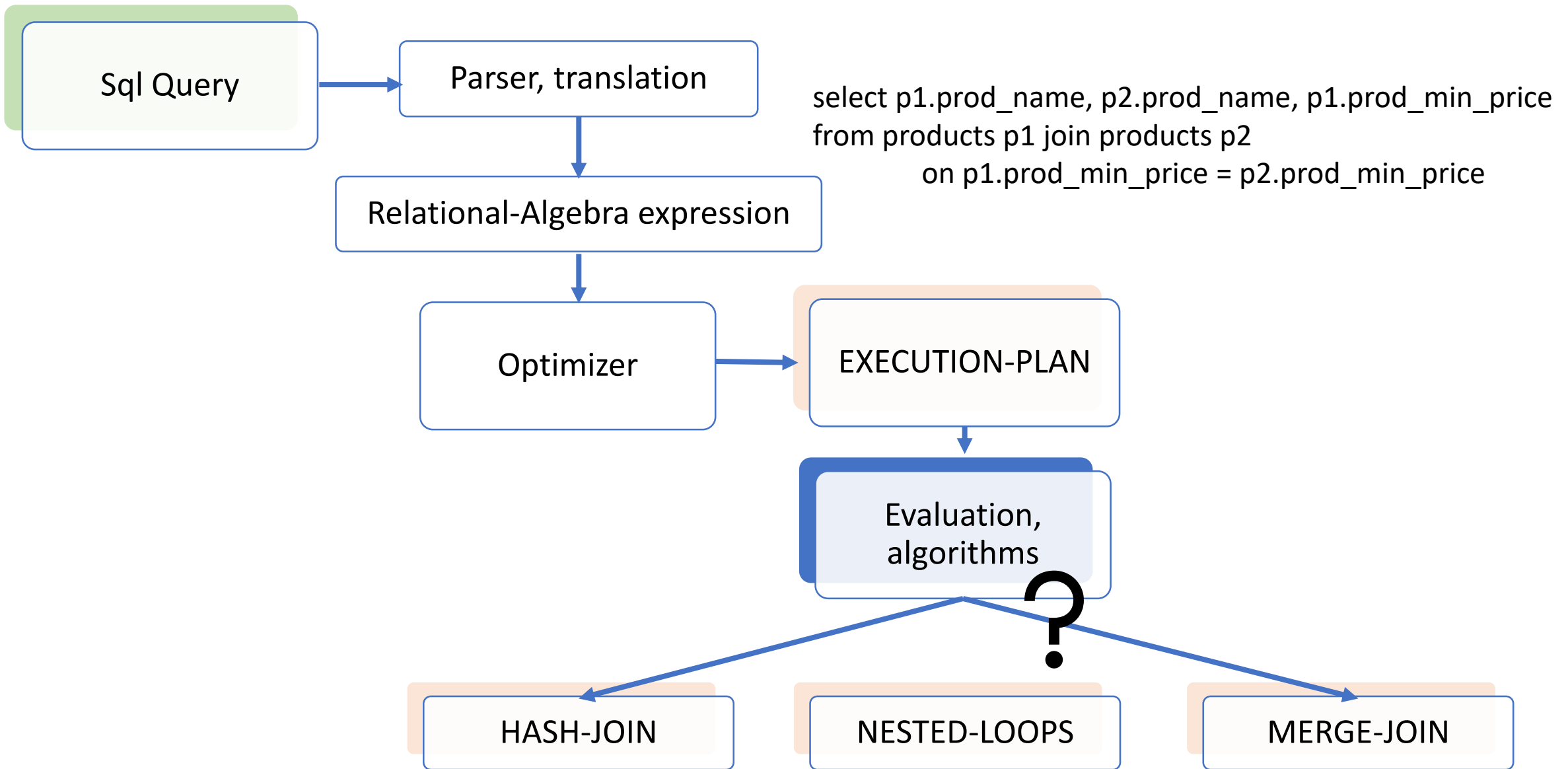


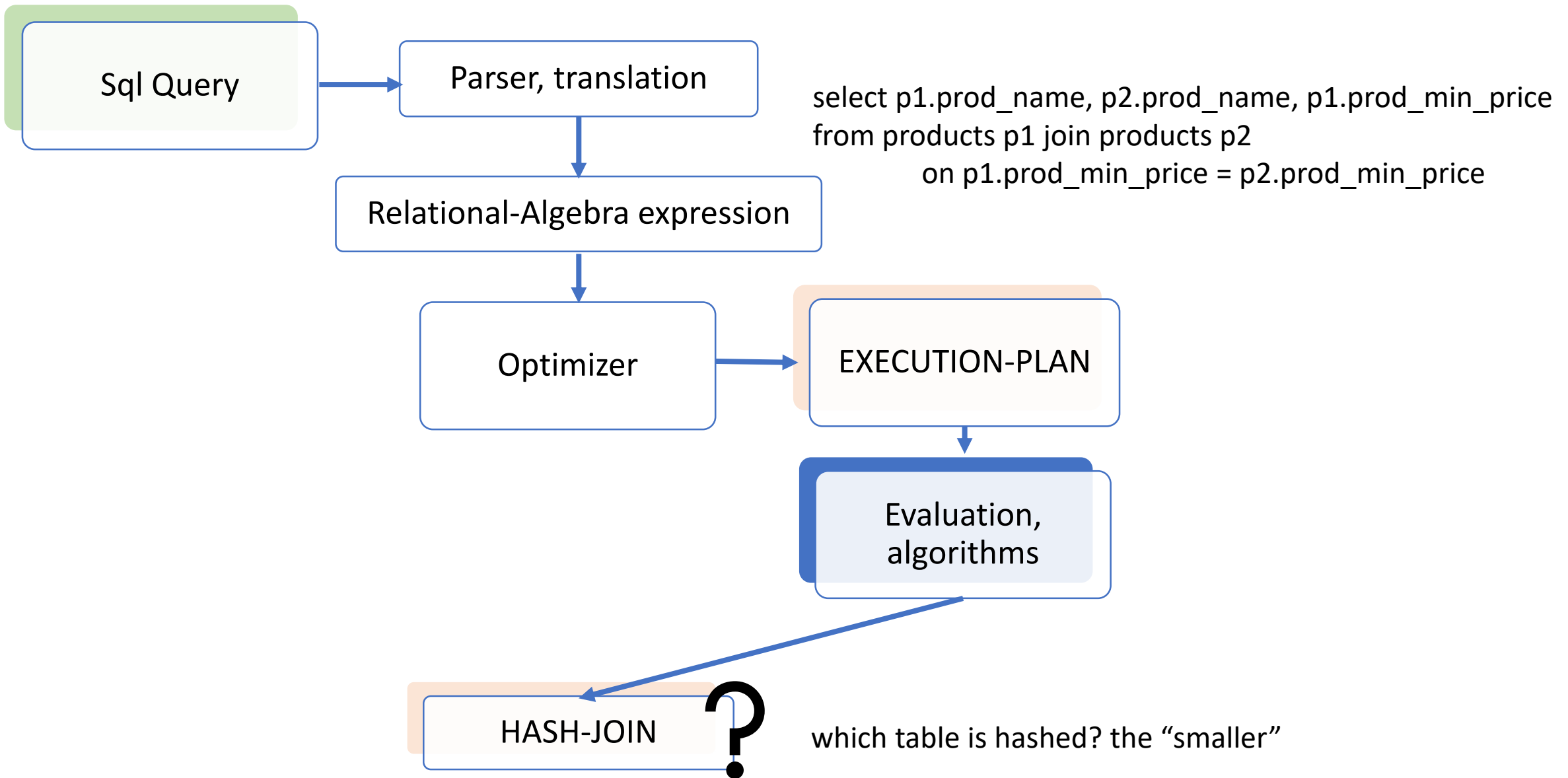
```
select p1.prod_name, p2.prod_name, p1.prod_min_price  
from products p1 join products p2  
on p1.prod_min_price = p2.prod_min_price
```

relations + operators

$JOIN(\prod_{name, minprice}^{p1}, \prod_{name, minprice}^{p2})$







Relational algebra properties

Relational algebra properties

- PROP1: join and cross product commute

$$\text{JOIN}(R1, R2) = \text{JOIN}(R2, R1)$$

$$R1 \times R2 = R2 \times R1$$

- PROP2: associativity

$$\text{JOIN}(\text{JOIN}(R1, R2), R3) = \text{JOIN}(R1, \text{JOIN}(R2, R3))$$

$$(R1 \times R2) \times R3 = R1 \times (R2 \times R3)$$

Relational algebra properties

- PROP3: projection composition

$$\Pi_{A_1, \dots, A_m} (\Pi_{B_1, \dots, B_n} (R)) = \Pi_{A_1, \dots, A_m} (R),$$

$$\{A_1, A_2, \dots, A_m\} \subseteq \{B_1, B_2, \dots, B_n\}.$$

- PROP4: selection composition

$$\sigma_{cond1} (\sigma_{cond2} (R)) = \sigma_{cond1 \wedge cond2} (R) = \sigma_{cond2} (\sigma_{cond1} (R)).$$

Relational algebra properties

- PROP5: selection and projection commute

$$\Pi_{A1, \dots, Am} (\sigma_{cond} (R)) = \sigma_{cond} (\Pi_{A1, \dots, Am} (R)).$$

$$\Pi_{A1, \dots, Am} (\sigma_{cond} (R)) = \Pi_{A1, \dots, Am} (\sigma_{cond} (\Pi_{A1, \dots, Am, B1, \dots, Bn} (R)))$$

- PROP6: selection and cross join commute

$$\sigma_{cond} (R1 \times R2) = \sigma_{cond} (R1) \times R2$$

$$\sigma_{cond} (R1 \times R2) = \sigma_{cond1} (R1) \times \sigma_{cond2} (R2)$$

$$\sigma_{cond} (R1 \times R2) = \sigma_{cond2} (\sigma_{cond1} (R1) \times R2)$$

Relational algebra properties

- PROP7: selection and union commute

$$\sigma_{cond}(R1 \cup R2) = \sigma_{cond}(R1) \cup \sigma_{cond}(R2)$$

- PROP8: selection and difference commute

$$\sigma_{cond}(R1 - R2) = \sigma_{cond}(R1) - \sigma_{cond}(R2)$$

Relational algebra properties

- PROP9: projection and cross product commute

$$\Pi_{A1, \dots, Am} (R1 \times R2) = \Pi_{B1, \dots, Bn} (R1) \times \Pi_{C1, \dots, Ck} (R2)$$

- PROP10: projection and union commute

$$\Pi_{A1, \dots, Am} (R1 \cup R2) = \Pi_{A1, \dots, Am} (R1) \cup \Pi_{A1, \dots, Am} (R2)$$

Relational algebra properties

- PROP11: join and projection commute

$$\Pi_{A1, \dots, Am} (\text{JOIN}(R1, R2, D)) = \Pi_{A1, \dots, Am} (\text{JOIN}(\Pi_{D, B1, \dots, Bn}(R1), \Pi_{D, C1, \dots, Ck}(R2), D)).$$

- PROP12: selection and join composition

$$\sigma_{\text{cond}} (\text{JOIN} (R1, R2, D)) = \sigma_{\text{cond}} (\text{JOIN} (\Pi_{D,A} (R1), \Pi_{D,A} (R2), D)).$$

General optimization rules

General optimization rules

- Execute selections first
 - Reduce relation size (number of rows)
- Avoid cross-joins, use joins
- First join to be executed is the one obtaining the smaller relation
- Execute projections first

Mesure Query Cost

rule-based execution
plans

obsolete

cost-based execution
plans

IO-cost

CPU-cost

disk accesses

number of blocks
transferred

number of tuples

CPU time

cost for processing
a tuple

cost for processing
an index entry

cost for processing
a tuple

cost for processing
a function

