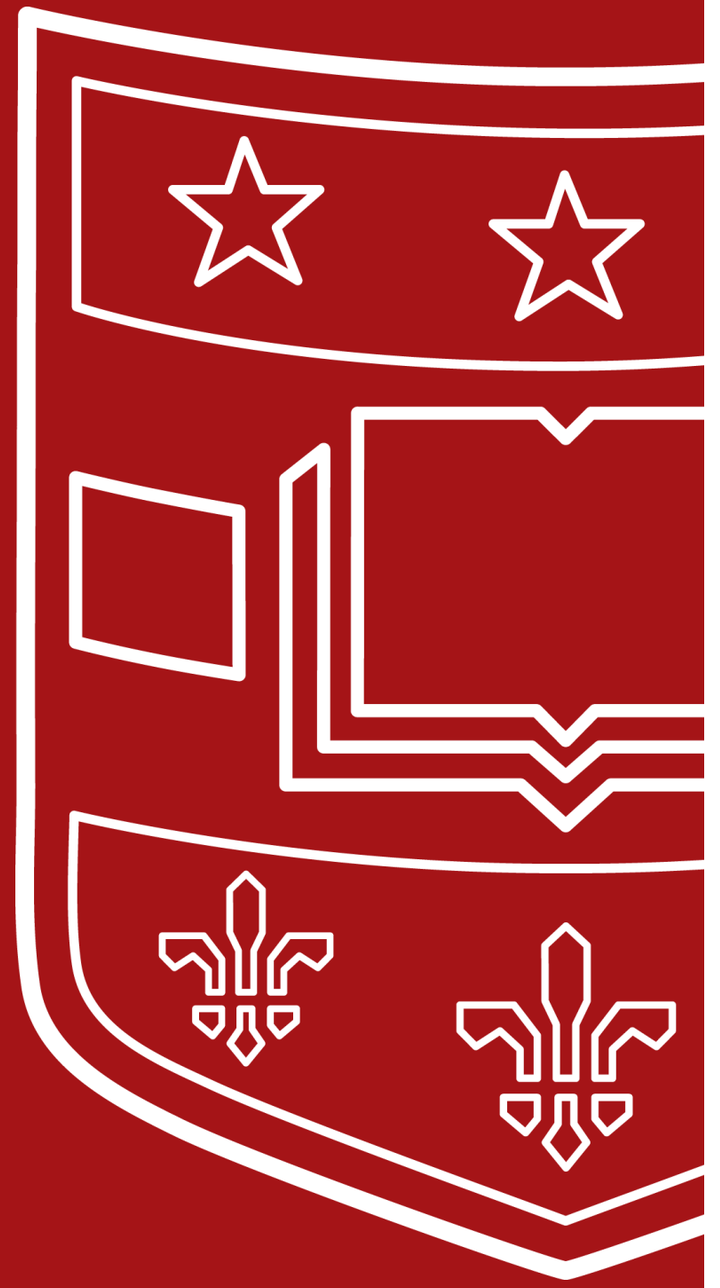
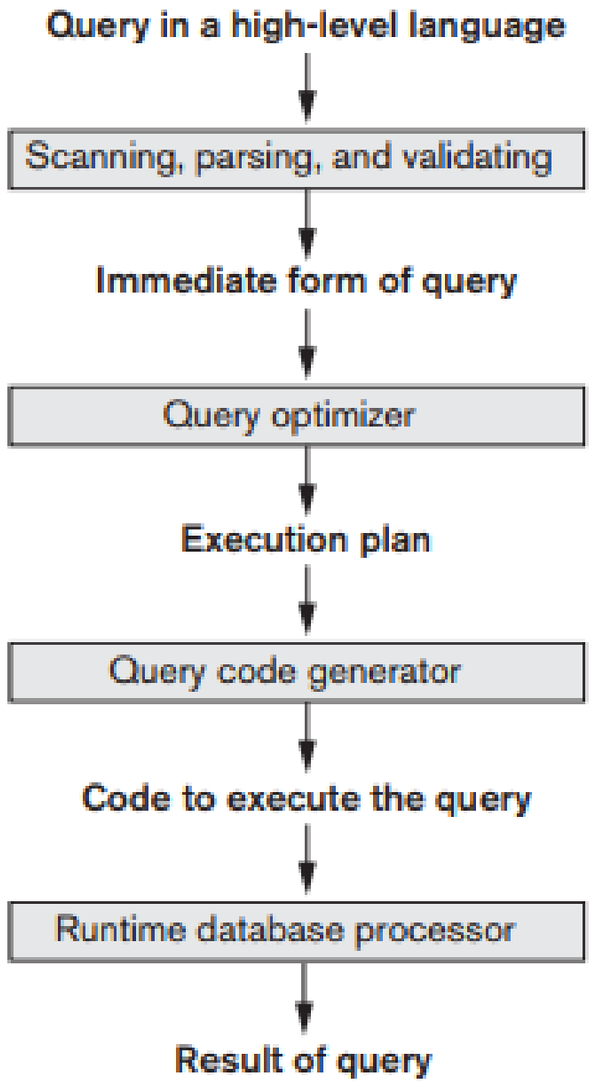
Database Management Systems



- Query Optimization

Washington University in St. Louis

Query Execution



Query Optimization

- Optimization is hard
  - Many possible combinations
- Maybe we aren’t finding the “optimal” solution
- What operations should we focus on?

SELECT Optimization

- Consider a simple SELECT query:

SELECT
FROM
WHERE

- How many possible ways are there to execute this?
  - What if WHERE is more complex?

Selectivity

- Ratio of tuples that satisfy the condition to the total number of tuples

- Can we compute this value exactly?

- How does this help us with optimization?

JOIN Optimization

- There’s more than one way to perform a JOIN
  - Nested Loop
  - Single Loop
  - Sort-merge
  - Partition-hash

Nested Loop Join

- How does table size affect this type of join?

- Example: assume one table has 10 pages and another has 2000 pages. How many page reads do we need for each nested loop configuration?

JOIN Selection Factor

- How many records from each table do we expect to match the join condition?

- How does this help us?
  - What type(s) of join does this affect?

Heuristics

- We can optimize queries by manipulating the query tree directly
  - Must satisfy order of operations

- Idea: one tree can be rewritten in numerous ways
  - Let’s find the fastest version

Heuristics

- Select
  - Can cascade conjunctive select conditions
  - Operations are commutative

- Project
  - Cascading projects are somewhat irrelevant
  - Can be commuted with a select
    - When?

Heuristics

- JOINs
  - Can be commuted with select if select only affects one side of the join
  - Can be commuted with project if all projected columns are part of the tables being joined

- Set operations
  - Union and intersection are commutative
  - Union, intersection, and join are associative
  - All set operations are commutative with select
  - Project is commutative with union

- Cartesian product + select = Join

Optimization Using Heuristics

- Break up conjunctive selects
  - Allows us to move them around more easily

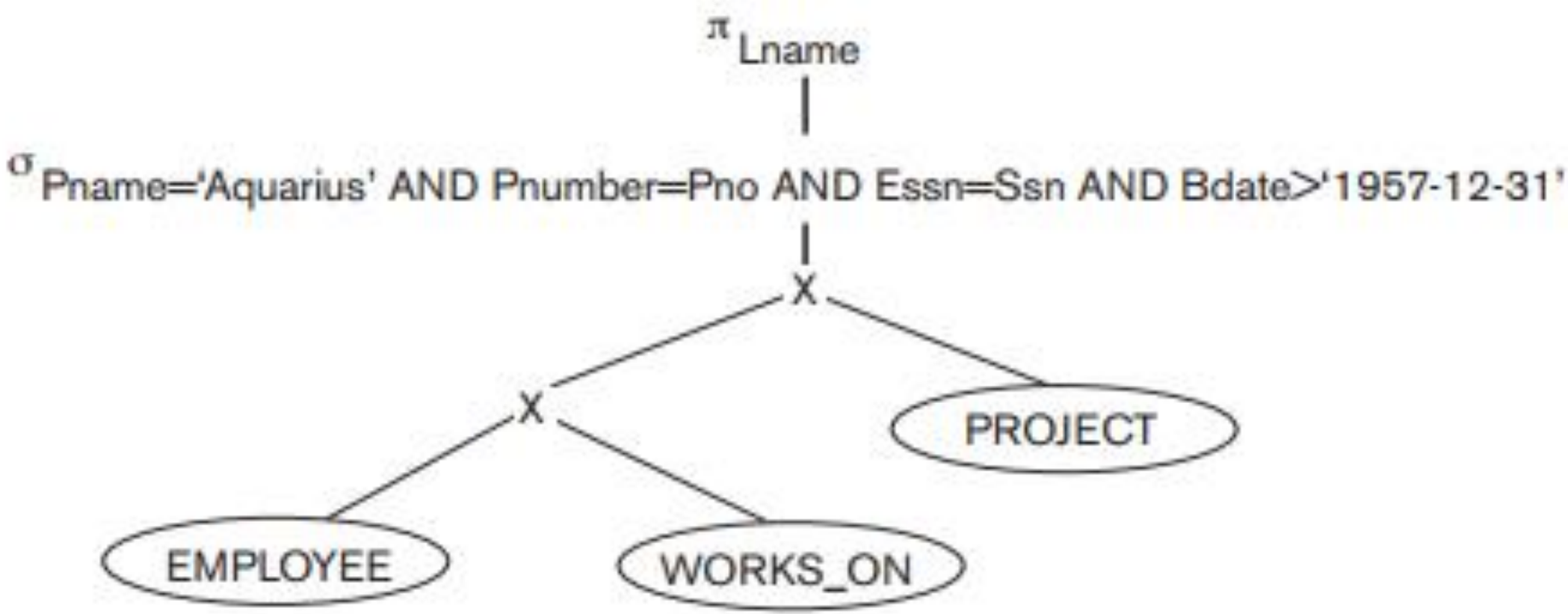
- Use select commutativity to move selects as far down the tree as possible

- When performing operations on multiple tables, move relations with select restrictions as far down the tree as possible

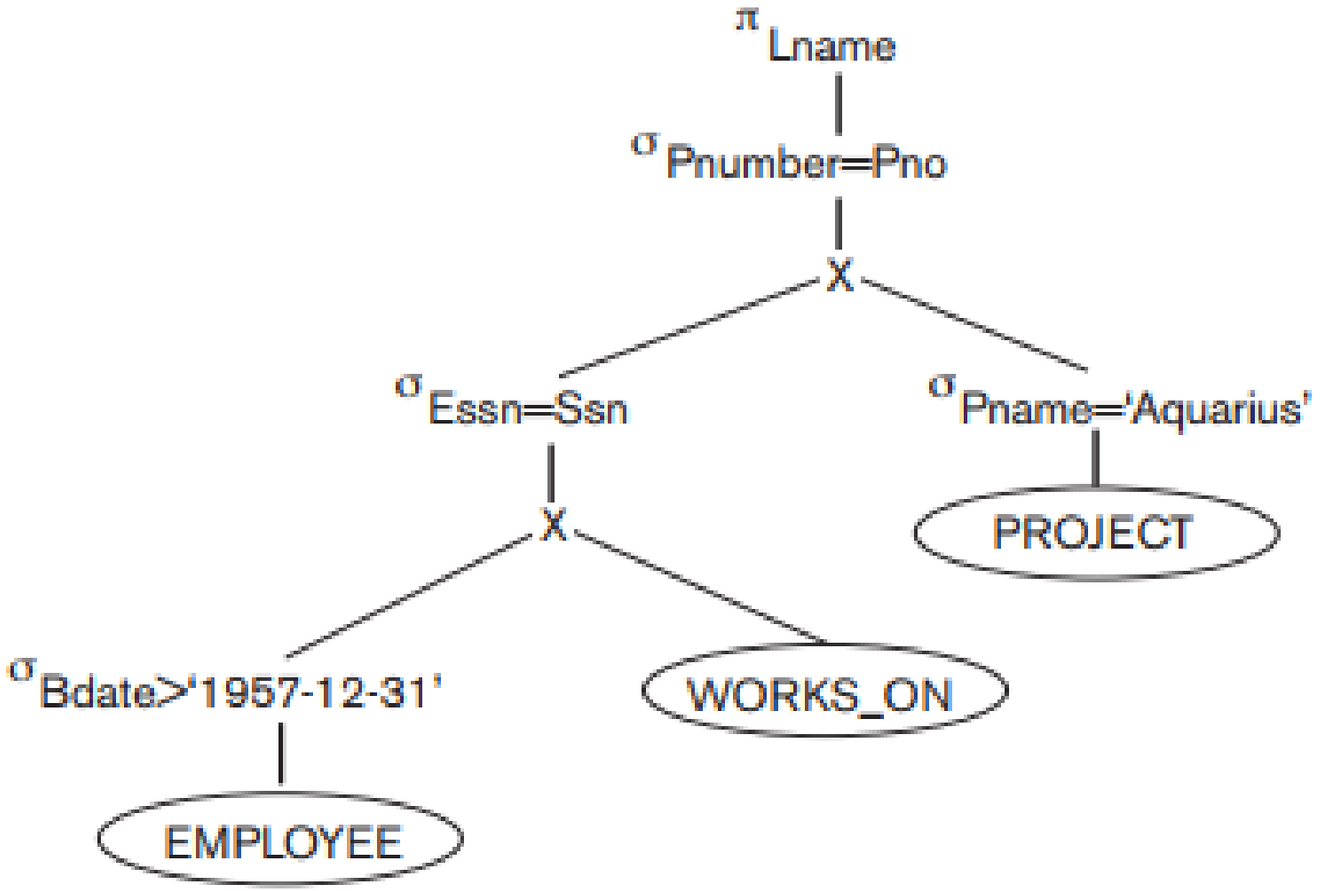
- Combine cartesian products with selects when possible

- Move projects down the tree as far as possible

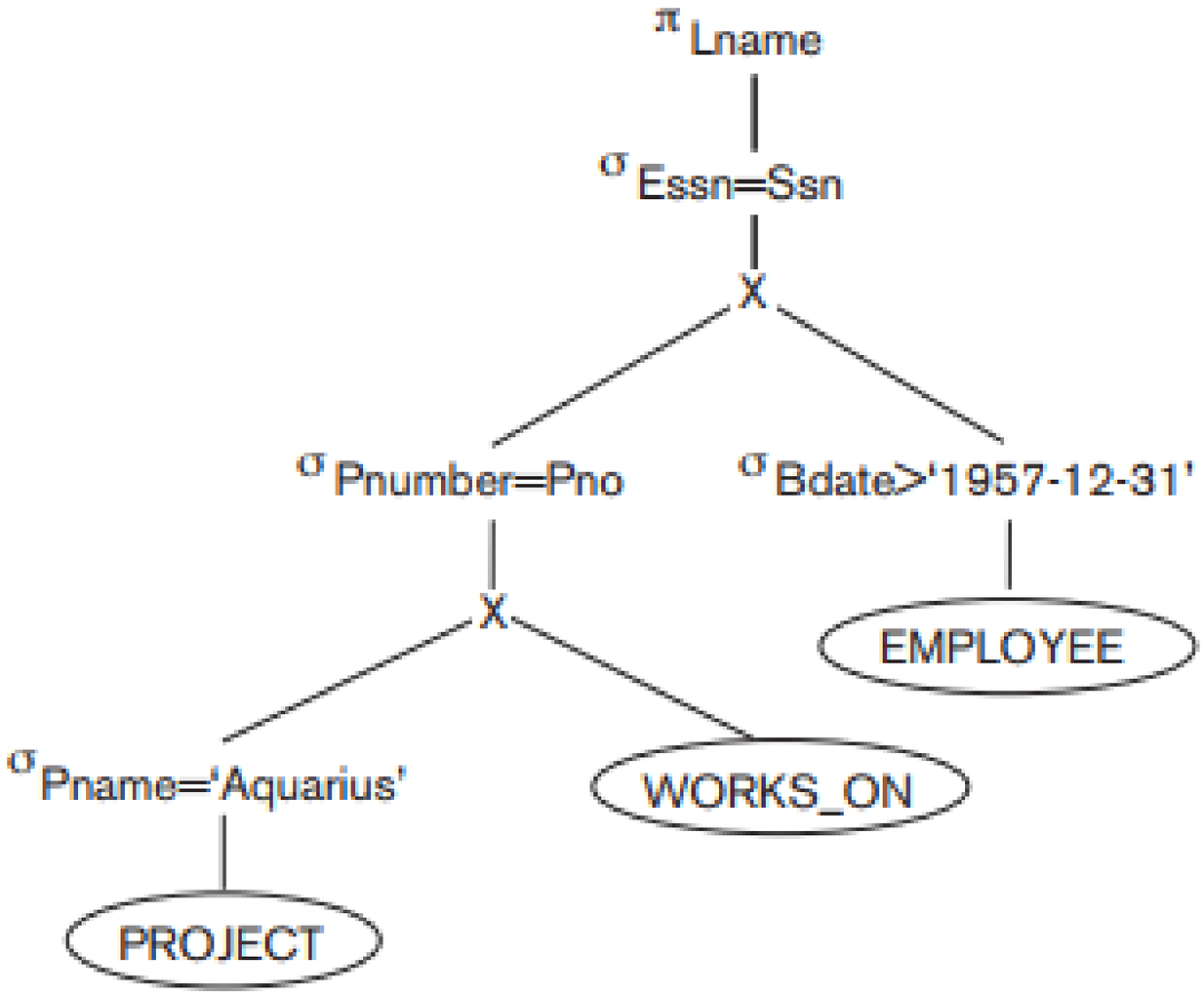
Heuristic Example



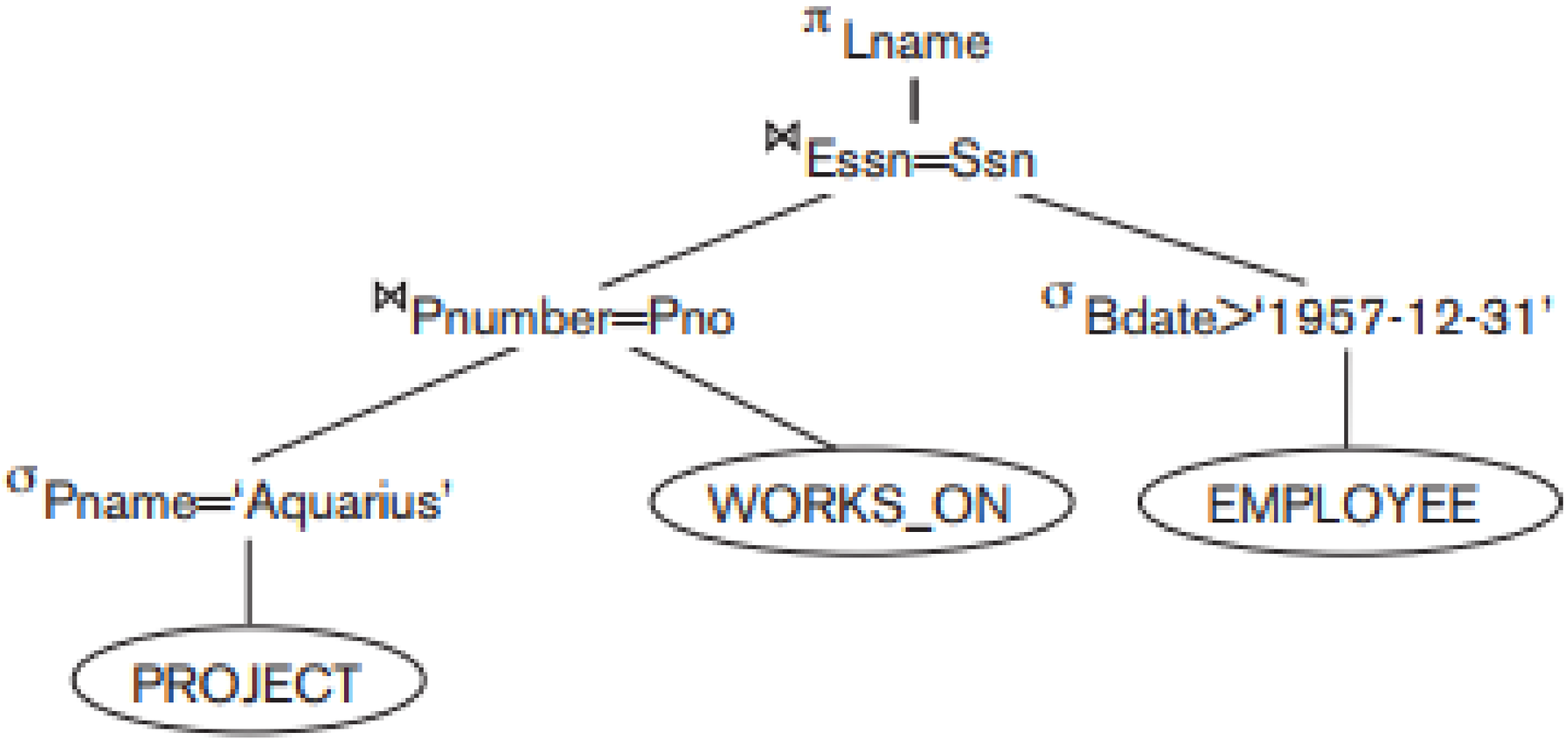
Heuristic Example



Heuristic Example



Heuristic Example



Heuristic Example