

Comando EXPLAIN sem alteração no modelo e sem melhoria na QUERY 2

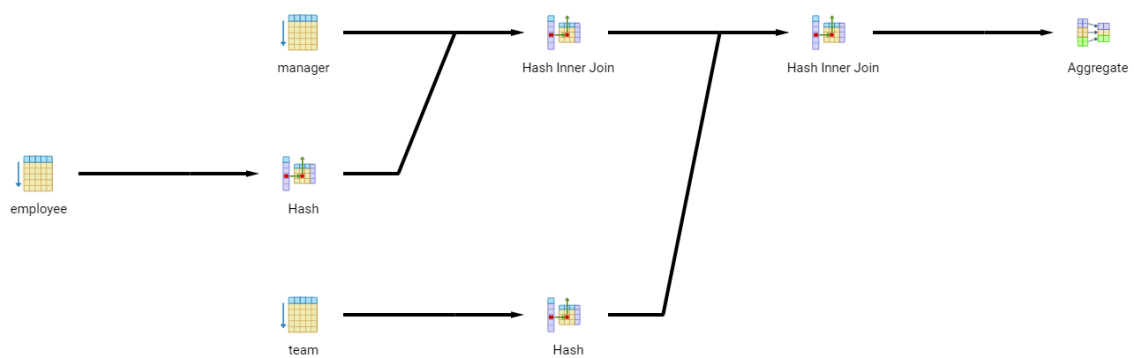
```
explain SELECT AVG(E.salary)
FROM employee as E NATURAL JOIN team as T NATURAL JOIN manager as M
WHERE E.idt_employee = M.fk_Employee_idt_employee
```

Consulta executada com o comando EXPLAIN

1.	→ Aggregate
2.	→ Hash Inner Join Hash Cond: (e.idt_team = t.idt_team)
3.	→ Hash Inner Join Hash Cond: (m.fk_employee_idt_employee = e.idt_employee)
4.	→ Seq Scan on manager as m
5.	→ Hash
6.	→ Seq Scan on employee as e
7.	→ Hash
8.	→ Seq Scan on team as t

	QUERY PLAN
	text
1	Aggregate (cost=118.50..118.51 rows=1 width=32)
2	-> Hash Join (cost=63.20..112.12 rows=2550 width=32)
3	Hash Cond: (e.idt_team = t.idt_team)
4	-> Hash Join (cost=26.20..68.40 rows=2550 width=36)
5	Hash Cond: (m.fk_employee_idt_employee = e.idt_employee)
6	-> Seq Scan on manager m (cost=0.00..35.50 rows=2550 width=4)
7	-> Hash (cost=17.20..17.20 rows=720 width=40)
8	-> Seq Scan on employee e (cost=0.00..17.20 rows=720 width=40)
9	-> Hash (cost=22.00..22.00 rows=1200 width=4)
10	-> Seq Scan on team t (cost=0.00..22.00 rows=1200 width=4)

Plano de consulta executada com o comando EXPLAIN



Árvore de consulta executada com o comando EXPLAIN

Comando EXPLAIN com alteração no modelo e sem melhoria na QUERY 2

```

/* Query 2 seleciona a média salarial geral dos diretores, gestores, especialistas e analistas,
* além de retornar a quantidade de funcionarios em cada tabela;
*/
explain SELECT COUNT (d.idt_employee) as qty_Director, AVG(d.salary) as mean_salary_Director,
COUNT (m.idt_employee) as qty_Manager, AVG(m.salary) as mean_salary_Manager,
COUNT (es.idt_employee) as qty_Especialist, AVG(es.salary) as mean_salary_Especialist,
COUNT (a.idt_employee) as qty_Analyst, AVG(a.salary) as mean_salary_Analyst
FROM employee as e
LEFT JOIN director as d ON d.idt_employee = e.idt_employee
LEFT JOIN manager as m ON m.idt_employee = e.idt_employee
LEFT JOIN especialista as es ON es.idt_employee = e.idt_employee
LEFT JOIN analyst as a ON a.idt_employee = e.idt_employee

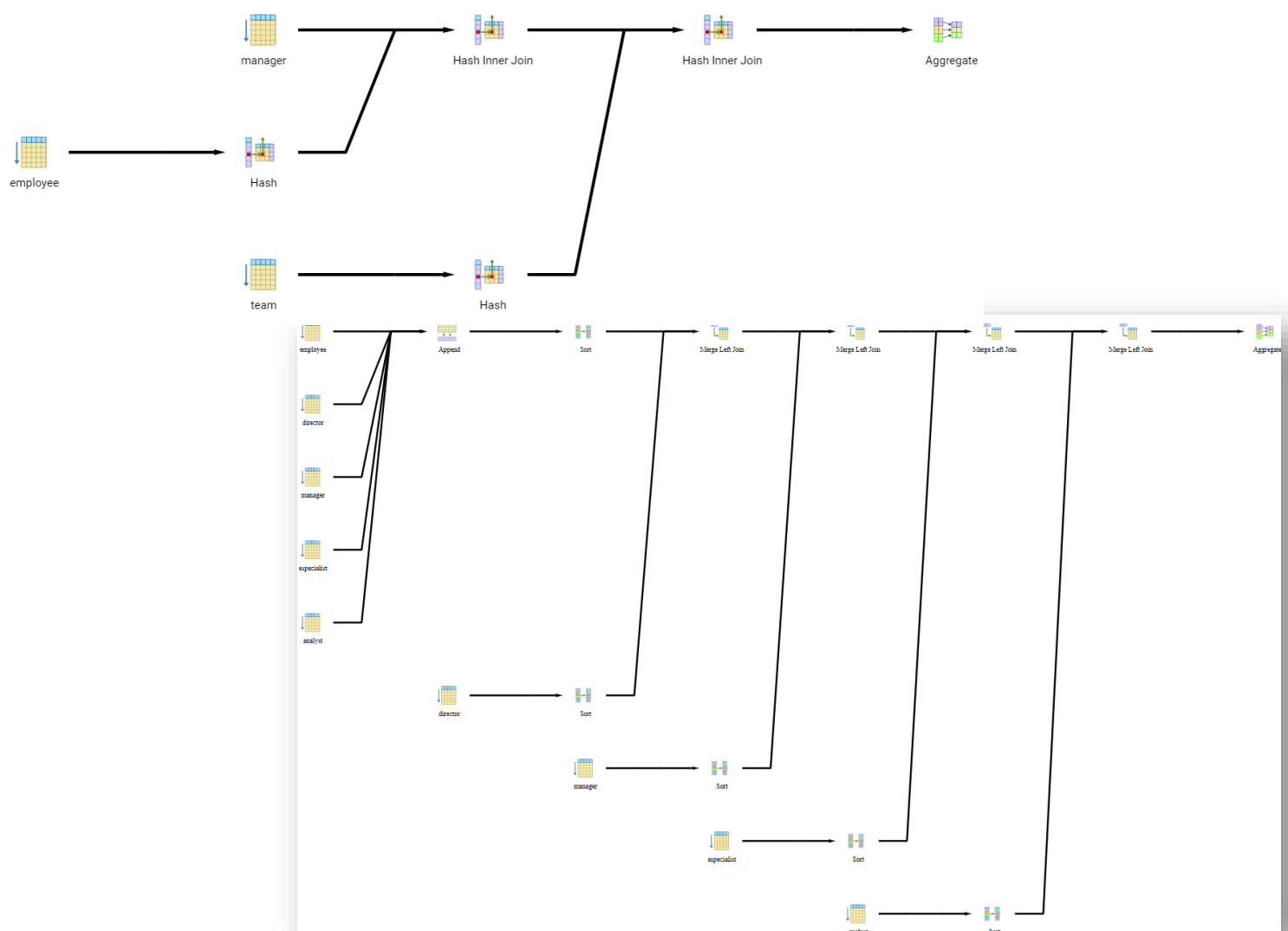
```

Consulta alterada e executada com o comando EXPLAIN

QUERY PLAN	text
1	Aggregate (cost=118.50..118.51 rows=1 width=32)
2	-> Hash Join (cost=63.20..112.12 rows=2550 width=32)
3	Hash Cond: (e.idt_team = t.idt_team)
4	-> Hash Join (cost=26.20..68.40 rows=2550 width=36)
5	Hash Cond: (m.fk_employee_idt_employee = e.idt_employee)
6	-> Seq Scan on manager m (cost=0.00..35.50 rows=2550 width=4)
7	-> Hash (cost=17.20..17.20 rows=720 width=40)
8	-> Seq Scan on employee e (cost=0.00..17.20 rows=720 width=40)
9	-> Hash (cost=22.00..22.00 rows=1200 width=4)
10	-> Seq Scan on team t (cost=0.00..22.00 rows=1200 width=4)

QUERY PLAN	text
1	Aggregate (cost=873.18..873.19 rows=1 width=160)
2	-> Merge Left Join (cost=242.03..645.77 rows=11370 width=144)
3	Merge Cond: (e.idt_employee = a.idt_employee)
4	-> Merge Left Join (cost=214.34..430.80 rows=6688 width=112)
5	Merge Cond: (e.idt_employee = es.idt_employee)
6	-> Merge Left Join (cost=186.64..292.95 rows=3934 width=76)
7	Merge Cond: (e.idt_employee = m.idt_employee)
8	-> Merge Left Join (cost=158.94..200.46 rows=2314 width=40)
9	Merge Cond: (e.idt_employee = d.idt_employee)
10	-> Sort (cost=131.25..134.65 rows=1361 width=4)
11	Sort Key: e.idt_employee
12	-> Append (cost=0.00..60.41 rows=1361 width=4)
13	-> Seq Scan on employee e (cost=0.00..0.00 rows=1 width=4)
14	-> Seq Scan on director e_1 (cost=0.00..13.40 rows=340 width=4)
15	-> Seq Scan on manager e_2 (cost=0.00..13.40 rows=340 width=4)
16	-> Seq Scan on especialista e_3 (cost=0.00..13.40 rows=340 width=4)
17	-> Seq Scan on analyst e_4 (cost=0.00..13.40 rows=340 width=4)
18	-> Sort (cost=27.70..28.55 rows=340 width=36)
19	Sort Key: d.idt_employee
20	-> Seq Scan on director d (cost=0.00..13.40 rows=340 width=36)
21	-> Sort (cost=27.70..28.55 rows=340 width=36)
22	Sort Key: m.idt_employee
23	-> Seq Scan on manager m (cost=0.00..13.40 rows=340 width=36)
24	-> Sort (cost=27.70..28.55 rows=340 width=36)
25	Sort Key: es.idt_employee
26	-> Seq Scan on especialista es (cost=0.00..13.40 rows=340 width=36)
27	-> Sort (cost=27.70..28.55 rows=340 width=36)
28	Sort Key: a.idt_employee
29	-> Seq Scan on analyst a (cost=0.00..13.40 rows=340 width=36)

Comparação entre os planos de consultas antes x depois da alteração do modelo



Comparação entre as árvores de consultas antes x depois da alteração do modelo

O plano de execução que foi aplicado na primeira consulta “realizada na Parte 1 do trabalho” seleciona a média salarial dos gestores apresentando função de agregação através da cláusula “WHERE”. Faz uma busca sequencial (“seq scan”) varrendo as tabelas “Employee” e “Manager”.

Após a alteração do modelo, no artefato “A” como mencionado, implementamos a herança na tabela “Employee” então vimos uma oportunidade de também alteramos a consulta para retornar além da média salarial dos gestores, as médias salariais dos diretores, especialistas e analistas juntamente com a quantidade de funcionários cadastrados em cada cargo. Além disso, ao contrário da primeira consulta, esta não faz uso da cláusula “WHERE” possibilitando uma outra análise. Se formos comparar com uma consulta que não tenha a cláusula “WHERE” (no caso a nossa segunda consulta) esse processo seria mais custoso por não ocorrer nenhum tipo de seleção, mas o que acontece neste caso é a ocorrência de muitas junções (“joins”) visto que utilizamos muitas tabelas para separar as visualizações. Poderíamos ter mantido a consulta da mesma forma que estava na “Parte 1” do trabalho, porém, decidimos realizar modificações para nos permitir visões, lados opostos das análises comuns.