**Query #1**

SELECT \*

FROM emp, dept

WHERE emp.deptno = dept.deptno AND emp.job = 'ENGINEER';

Cambiare l’obiettivo di ottimizzazione dalla modalità ALL ROWS (best throughput) alla modalità

FIRST\_ROWS (best response time) attraverso l’uso di hint (/\*+ FIRST\_ROWS(n) \*/). n è una

variabile numerica intera che può assumere valori maggiori o uguali a 1. Assegnare diversi valori

ad n e verificare come variano il piano d’esecuzione e i costi delle diverse operazioni.

ALL ROWS: piano esecuzione ottimizzato in modo da minimizzare il tempo di esecuzione di tutta

la query.

FIRST\_ROWS(n): piano esecuzione ottimizzato in modo da minimizzare il tempo di esecuzione per

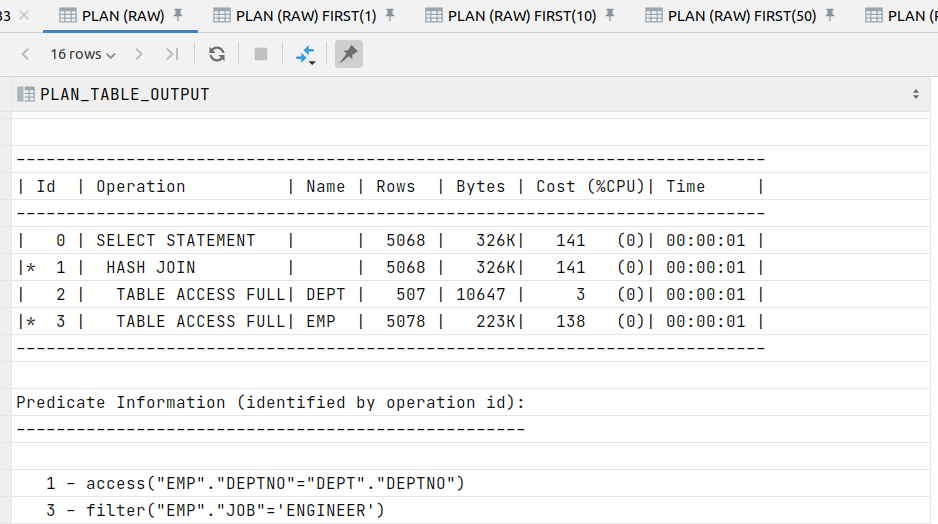
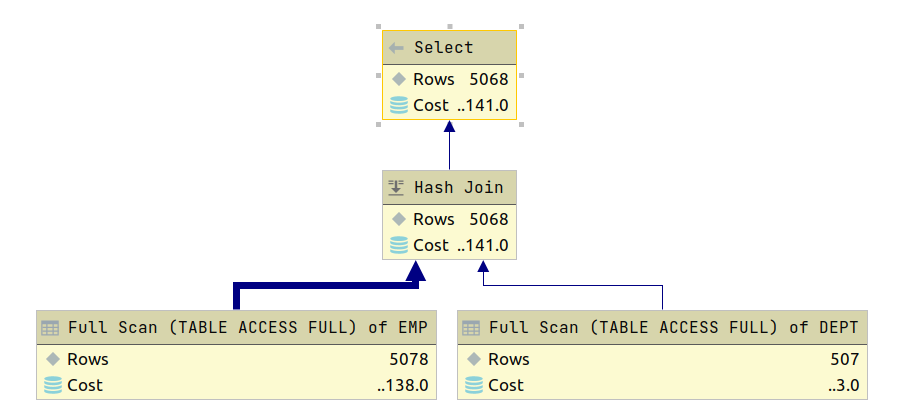
i primi n record del risultato.

SELECT /\*+ FIRST\_ROWS(n) \*/ \*

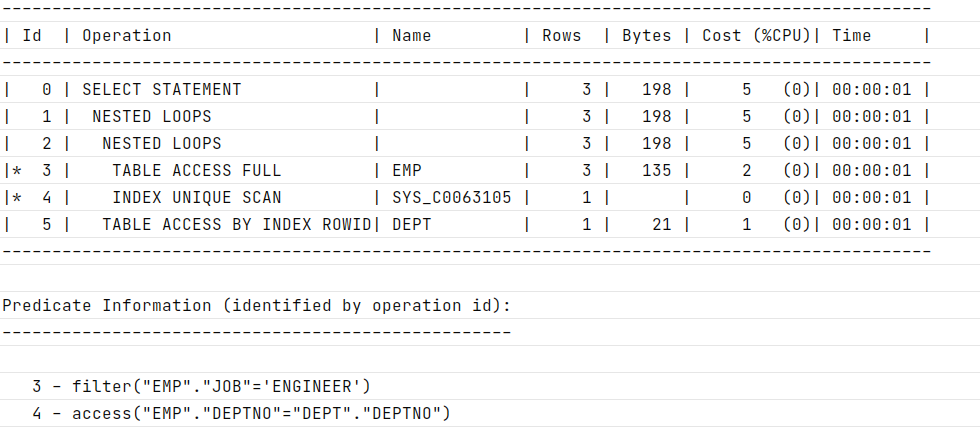
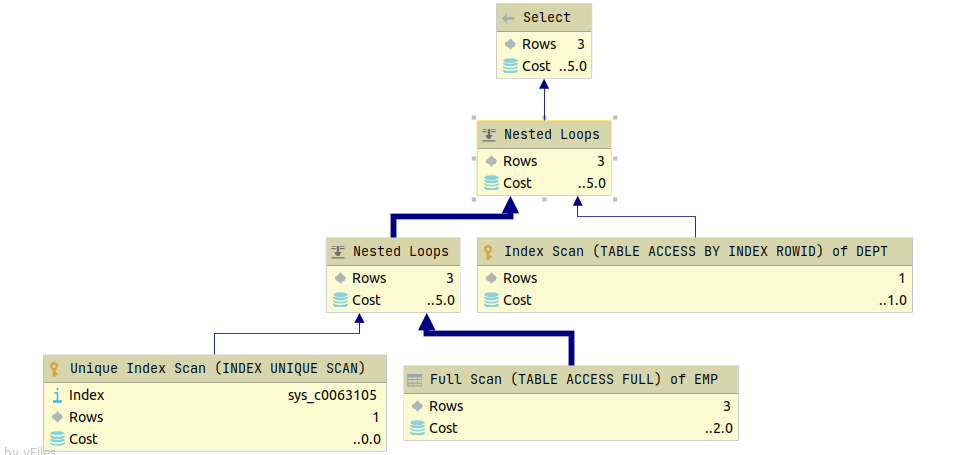
FROM emp, dept

WHERE emp.deptno = dept.deptno AND emp.job = 'ENGINEER';

**ALL ROWS**



**FIRST\_ROWS(1)**



**FIRST\_ROWS(10,50,150,2500,5000)**

Piano uguale a all\_rows

first\_rows(10)->TAF-> 12

first\_rows(50)->TAF-> 52

first\_rows(150)->TAF-> 152

hash join->563

first\_rows(2500)->TAF-> 2506

hash join->2511

first\_rows(5000)->TAF-> 5011

hash join->5001

**Query #2**

Confrontare i costi di hash join e nested loop usando l’hint USE / NO\_USE\_HASH

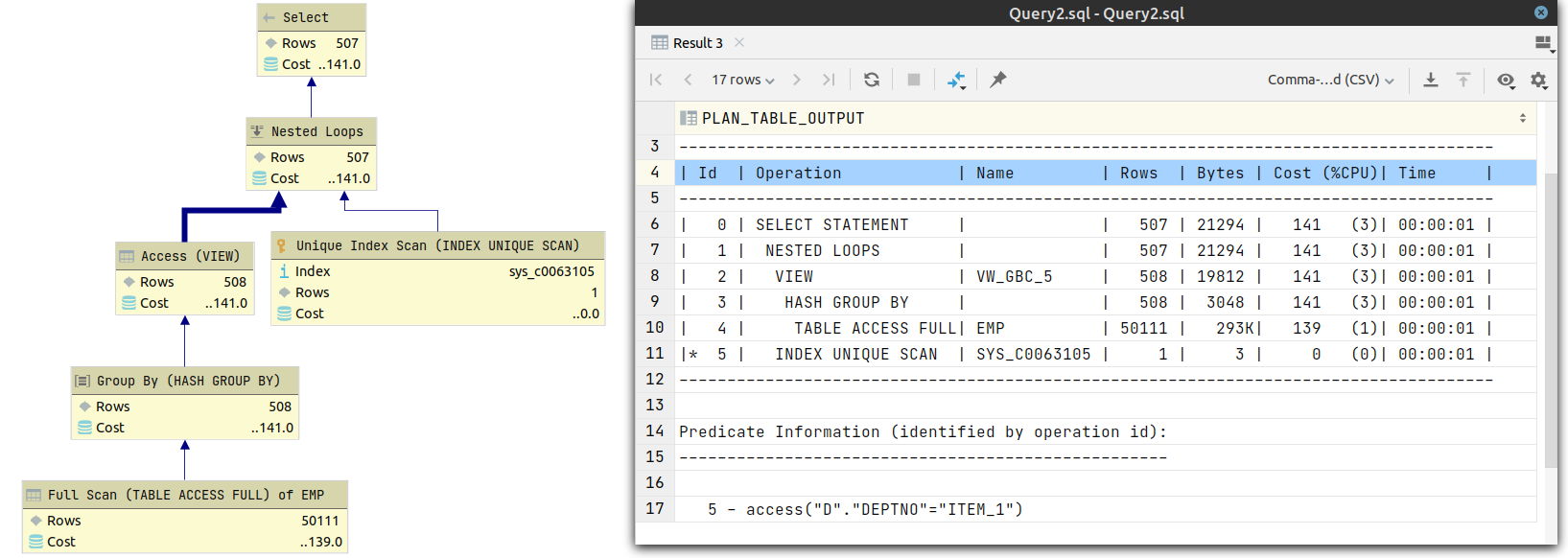
SELECT /\*+ NO\_USE\_HASH(e d) \*/ d.deptno, AVG(e.sal)

FROM emp e, dept d

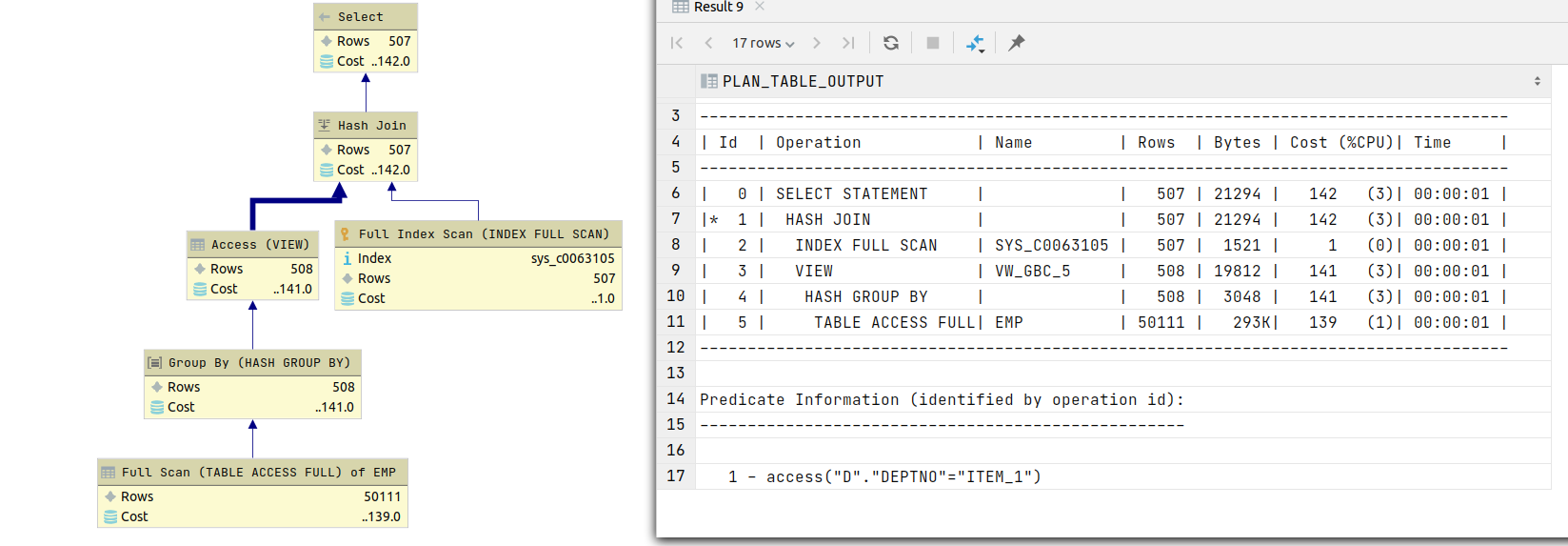
WHERE d.deptno = e.deptno

GROUP BY d.deptno;

**Default=No\_Use\_Hash(e d)**



**Use\_Hash(e d)**

****

**Query #3**

Disabilitare il metodo hash join mediante l’uso di hint (/\*+ NO\_USE\_HASH(e d) \*/)

SELECT /\*+ NO\_USE\_HASH(e d) \*/ ename, job, sal, dname

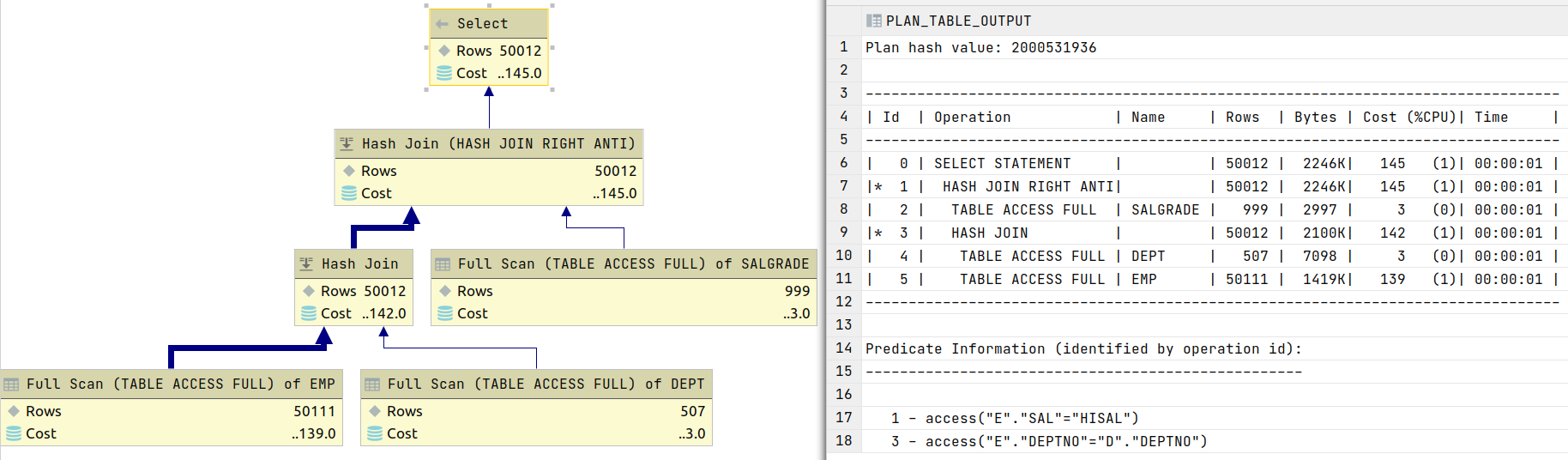
FROM emp e, dept d

WHERE e.deptno = d.deptno

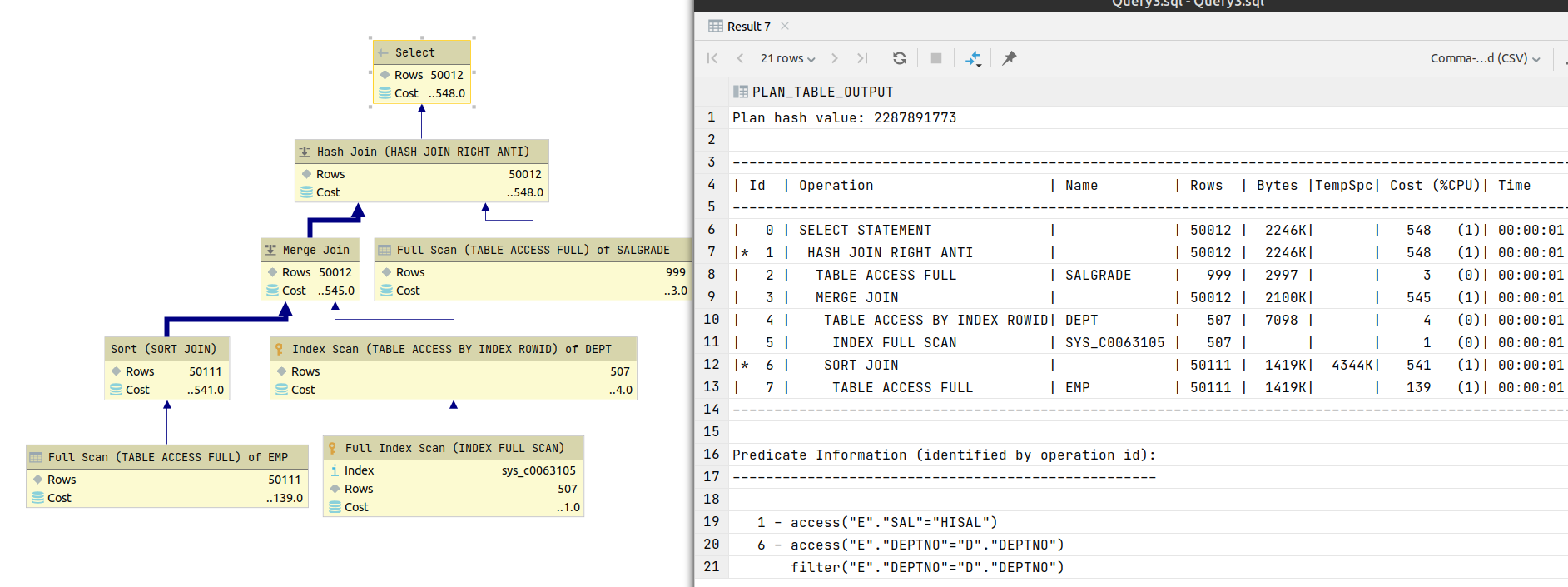
AND NOT EXISTS

(SELECT \* FROM salgrade WHERE e.sal = hisal);

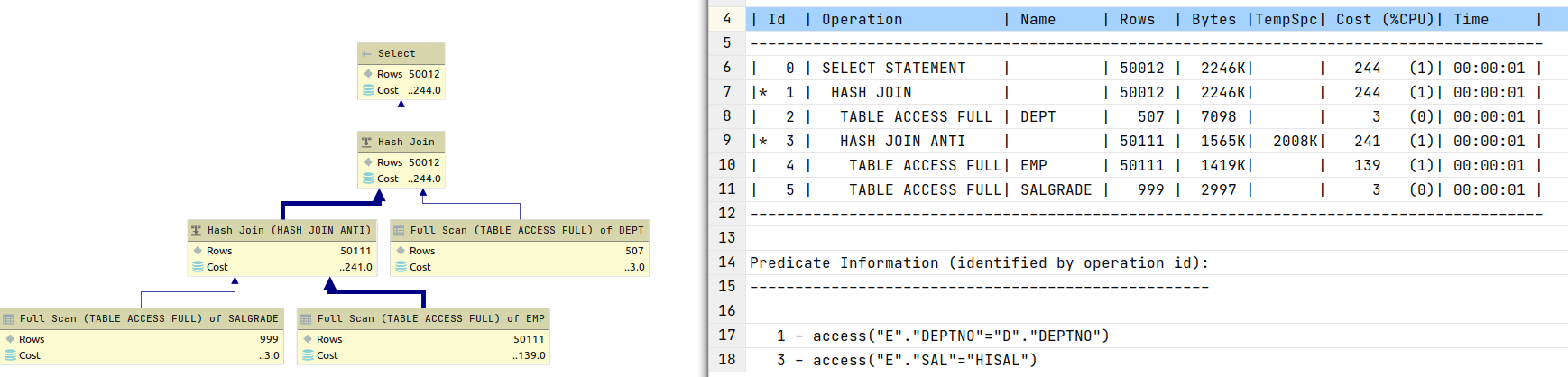
**Default=Use\_Hash(e d)**



**No\_Use\_Hash(e d)**



**/\*+ ORDERED \*/(prima join tra emp e salgrade, e poi join con dept)**



**Query #4**

Si definiscano una o più strutture secondarie (indici) che permettano l’ottimizzazione della seguente

query. Si analizzi con particolare attenzione il cambiamento nel piano di esecuzione creando due indici

sugli attributi interessati dall’interrogazione.

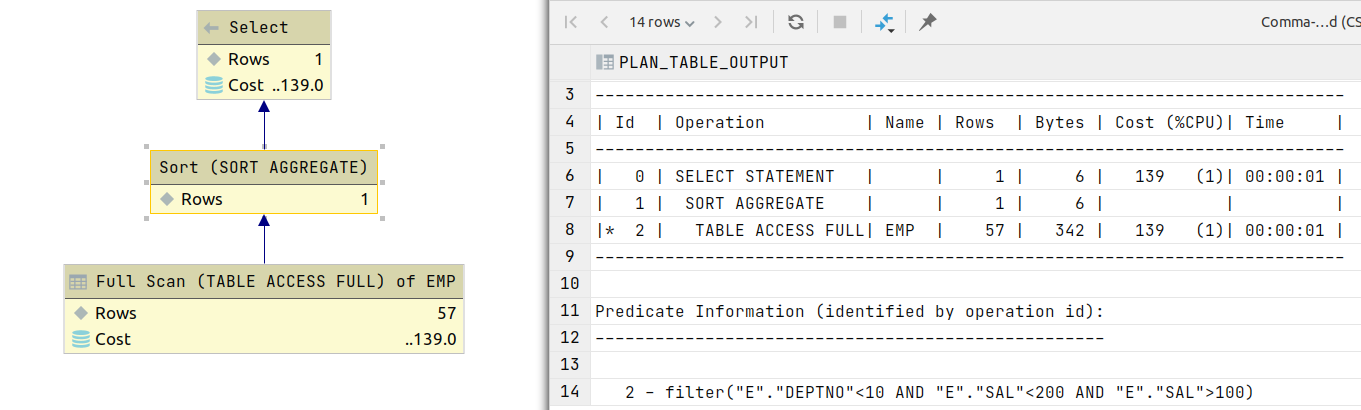
select avg(e.sal)

from emp e

where e.deptno < 10 and

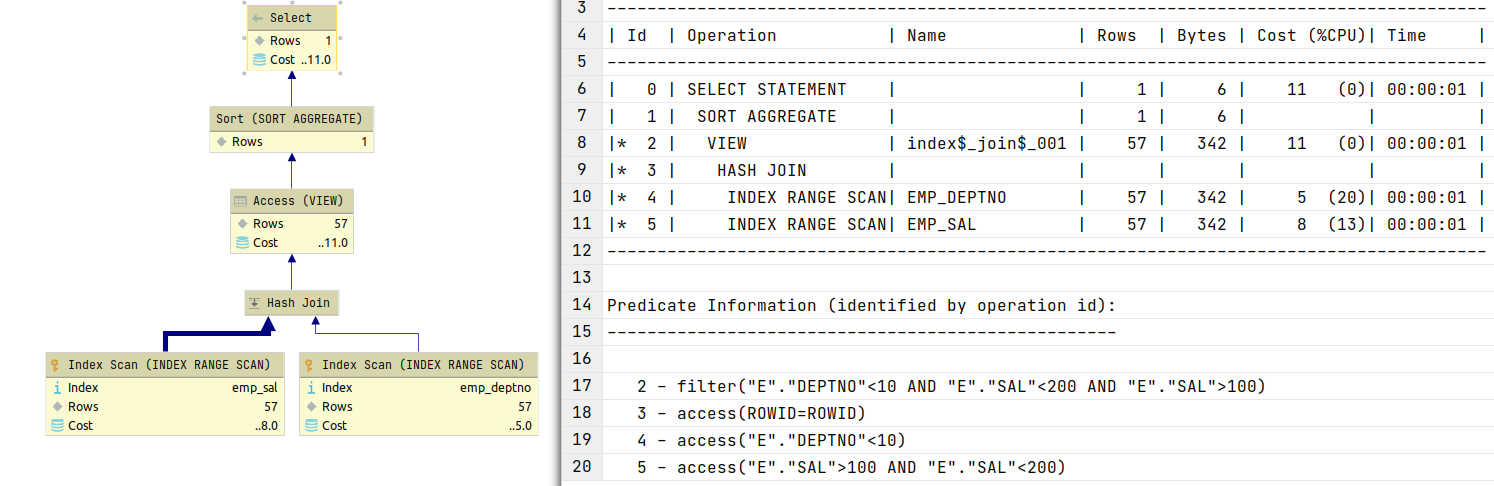
e.sal > 100 and e.sal < 200;

**Default oppure deptno B-tree index oppure sal B-tree index**

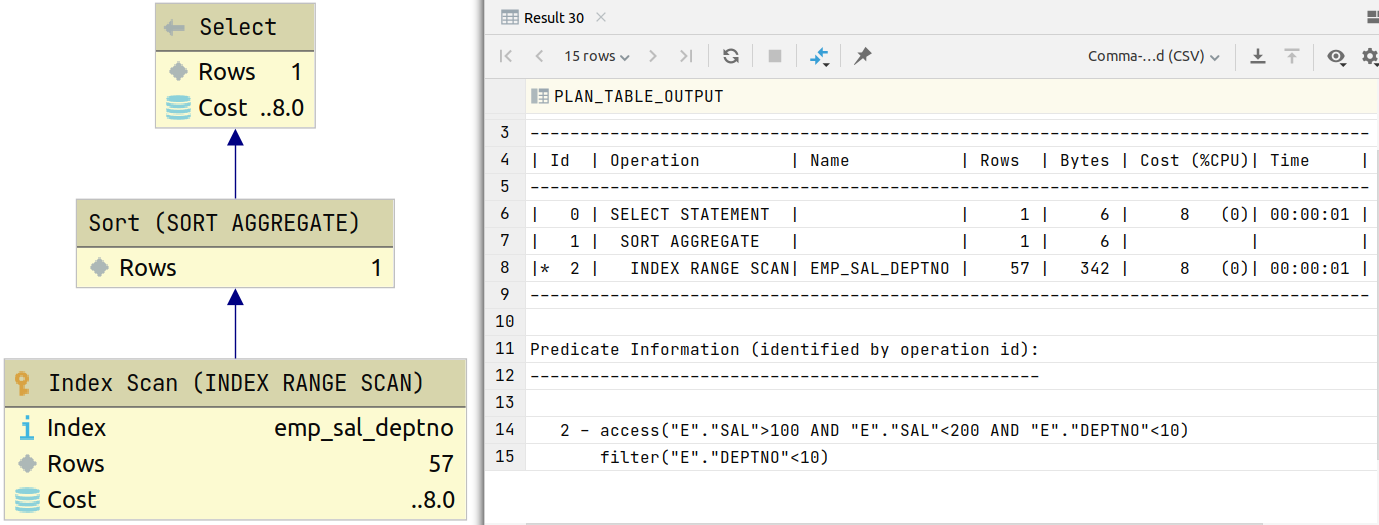


(SORT AGGREGATE does not actually involve a sort. It is used when aggregates are being computed across the whole set of rows)

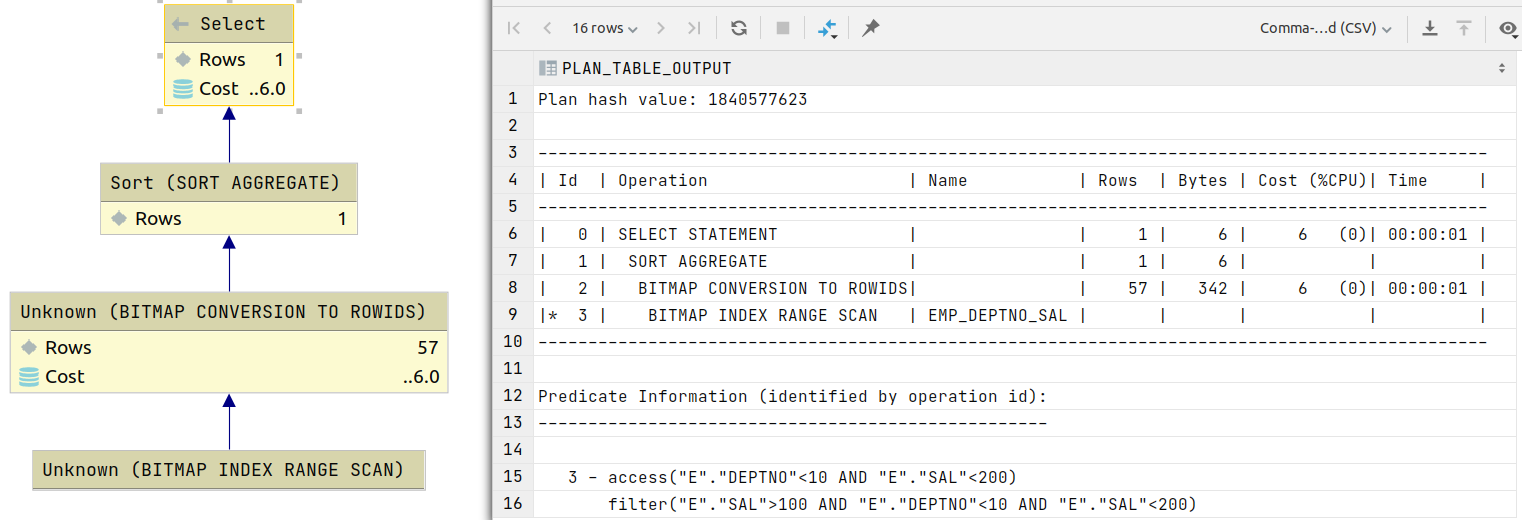
**Deptno B-tree index + sal B-tree index**

****

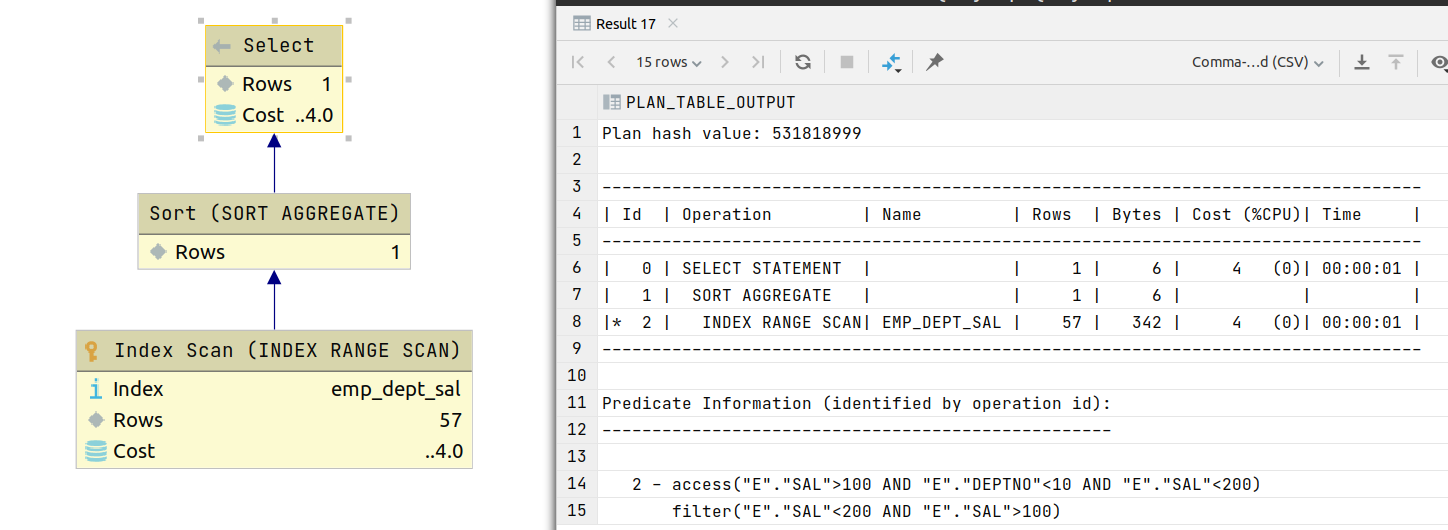
**Sal,deptno B-tree index**

****

**Deptno,sal BITMAP index (deptno più selettivo!)**

****

**Deptno,sal B-tree index (deptno più selettivo!)**

****

**Query #5**

Si definiscano una o più strutture secondarie (indici) che permettano l’ottimizzazione della seguente

query:

select dname

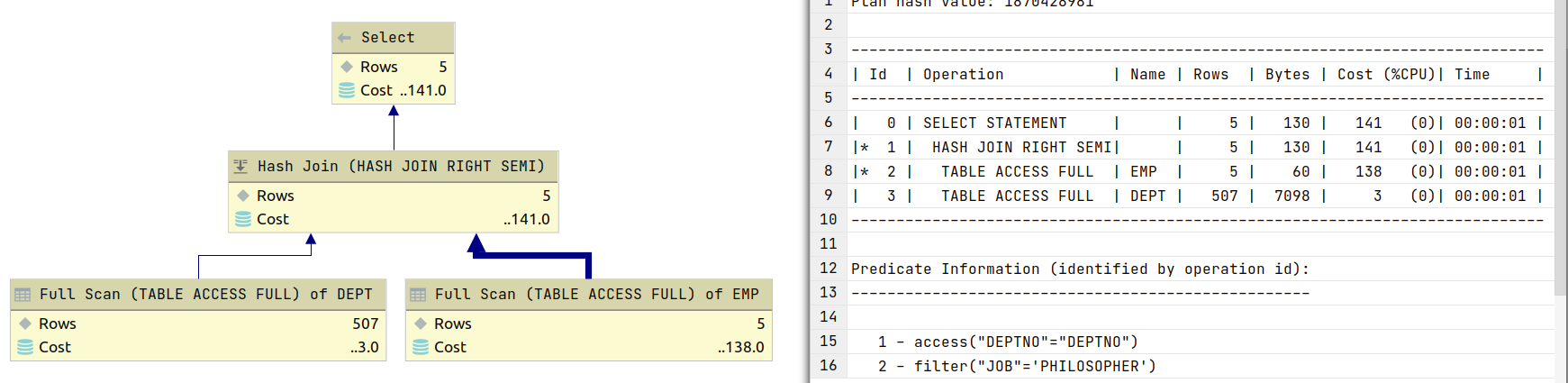
from dept

where deptno in (select deptno

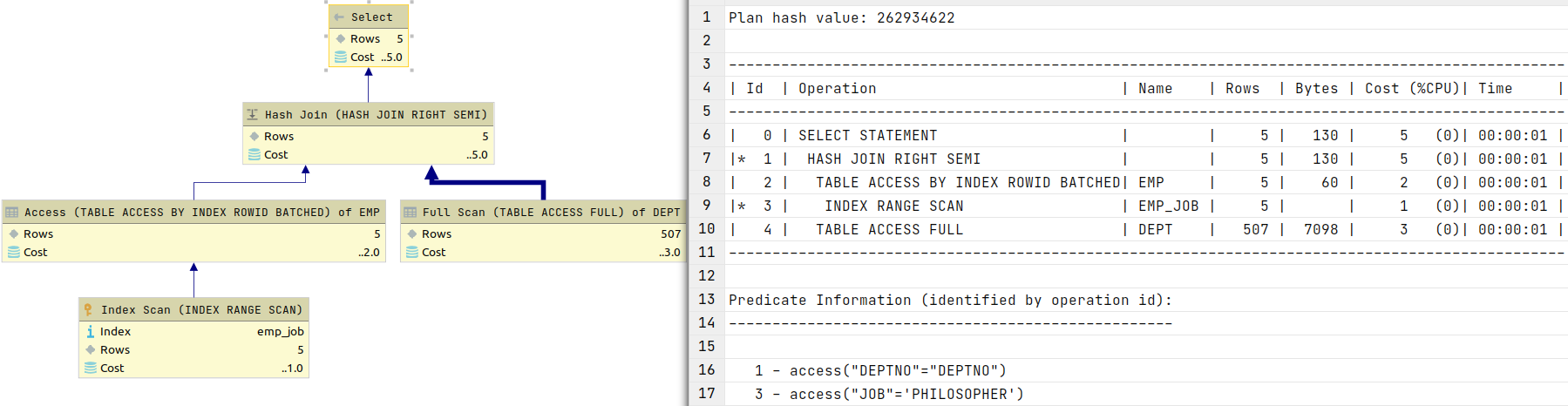
from emp

where job = 'PHILOSOPHER');

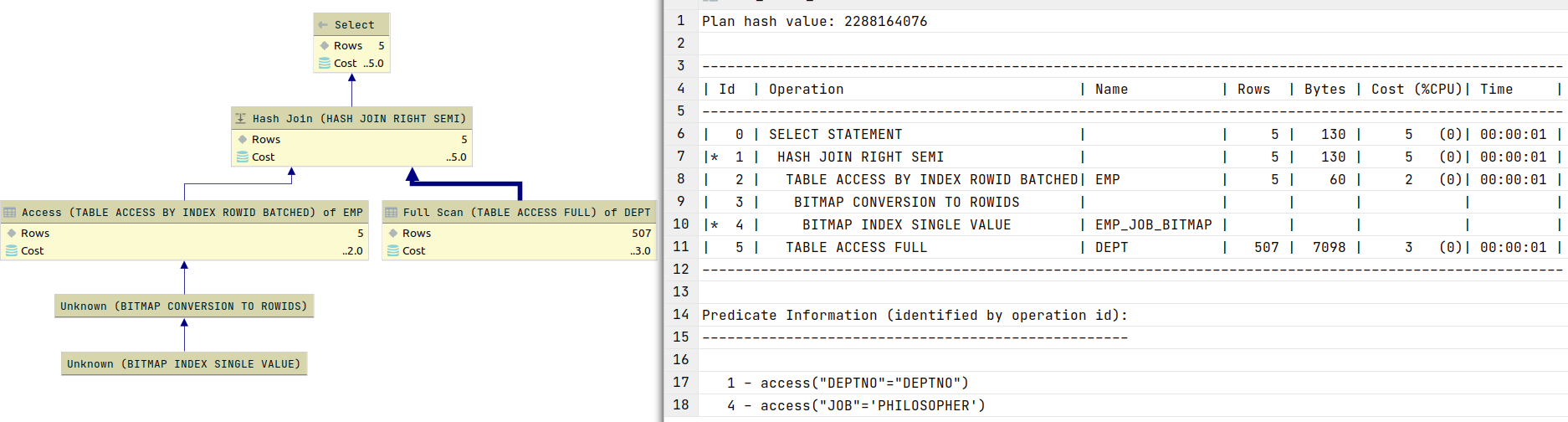
**Default o indice su dept(dname)**

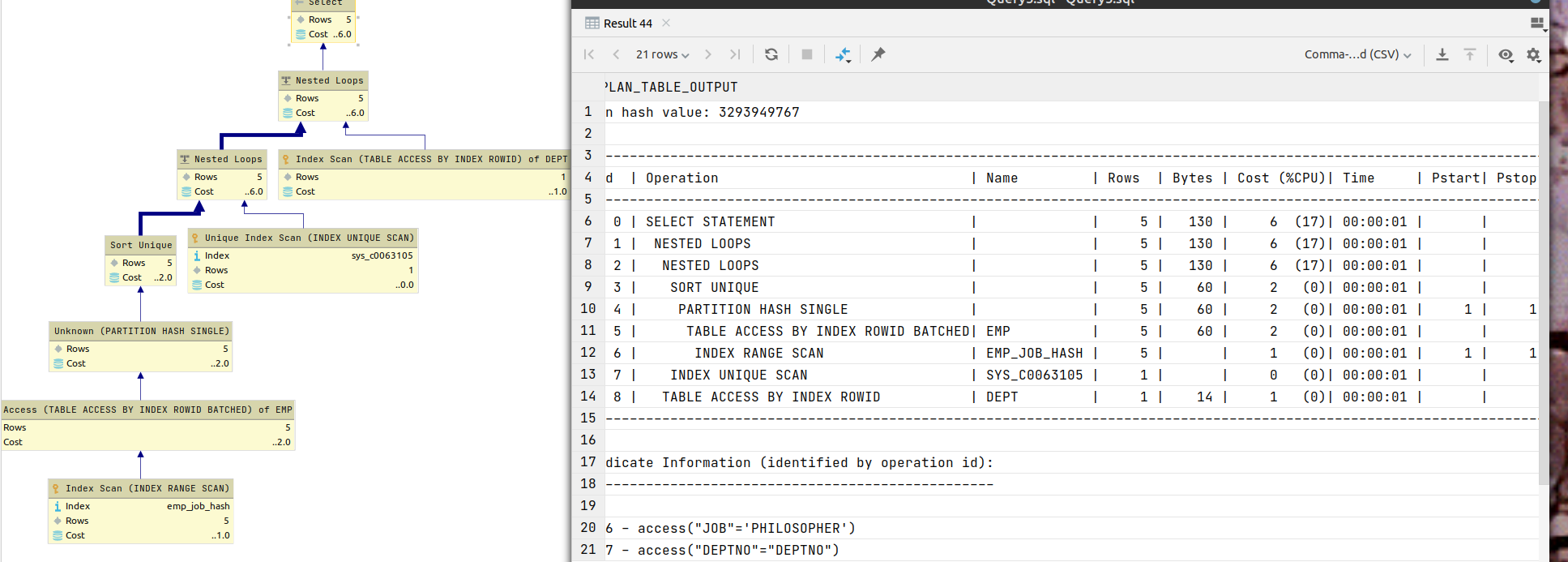


**Job B-tree index**

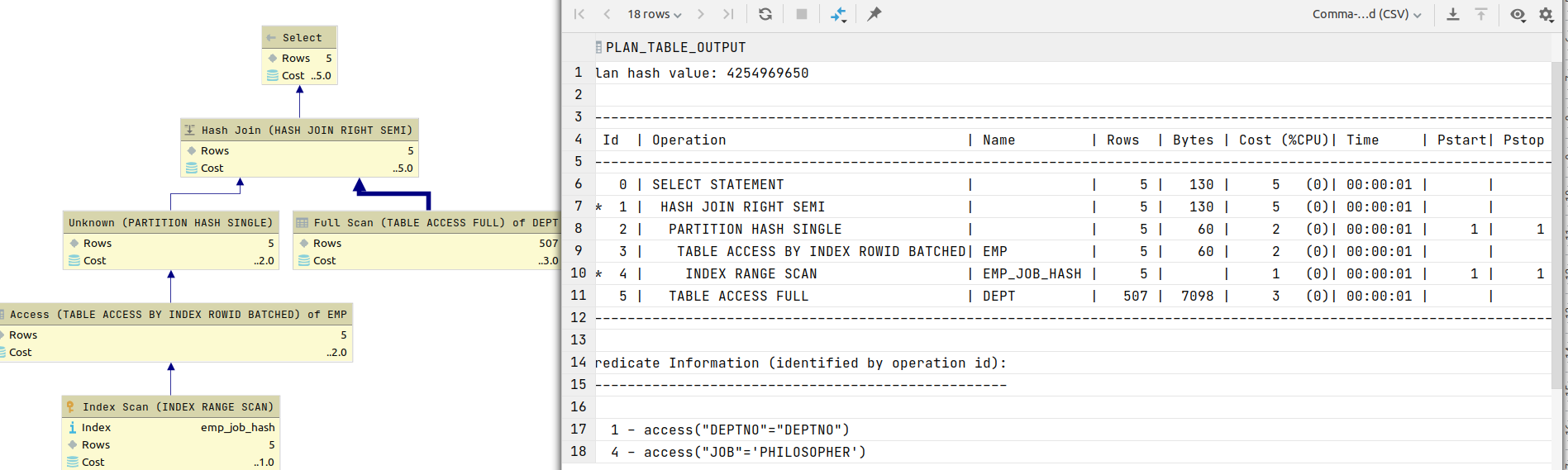
****

**Job BITMAP index**

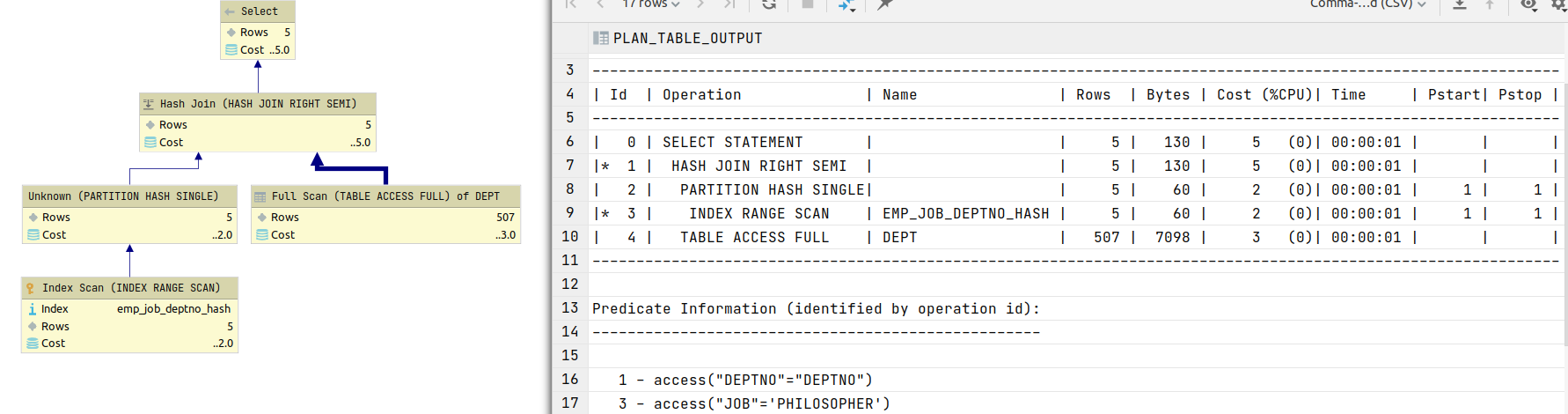
****

**Job HASH index Nested Loop**

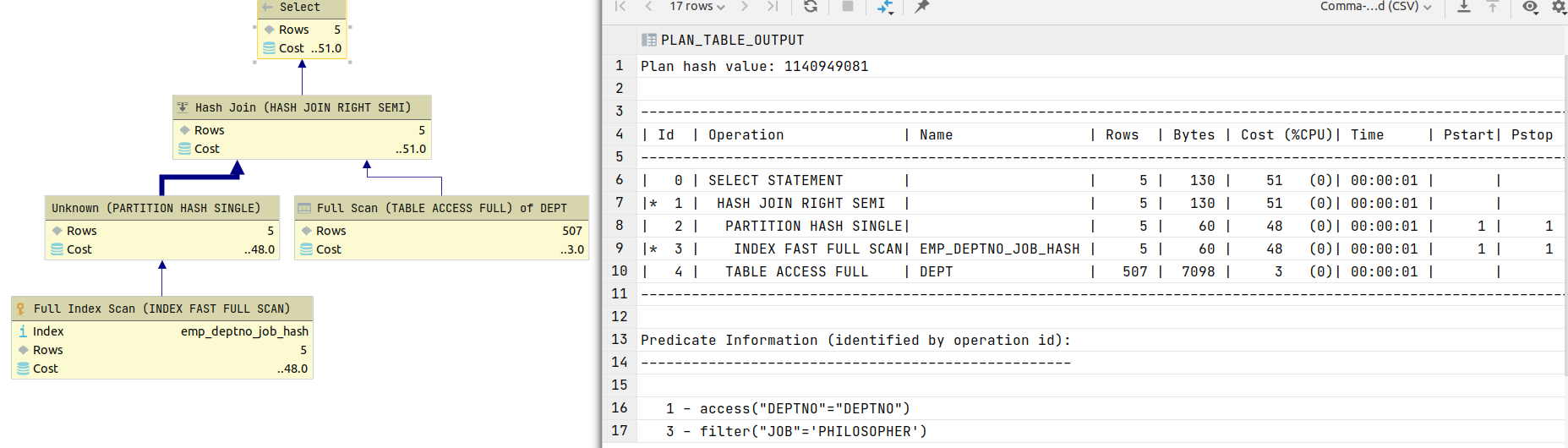
**Job HASH index Hash Join**

****

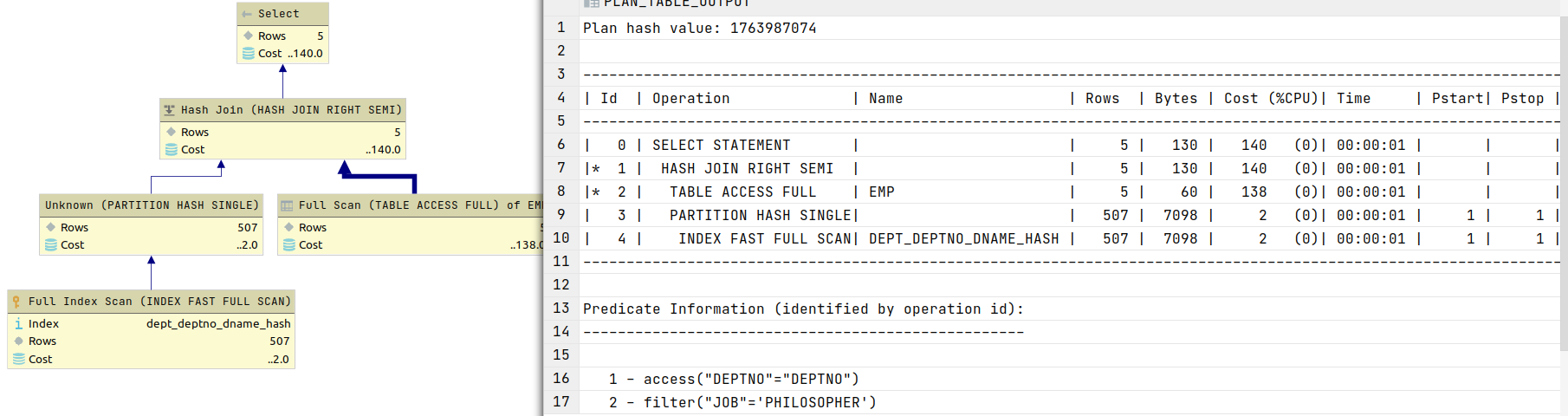
**Job,Deptno HASH index Hash Join**

****

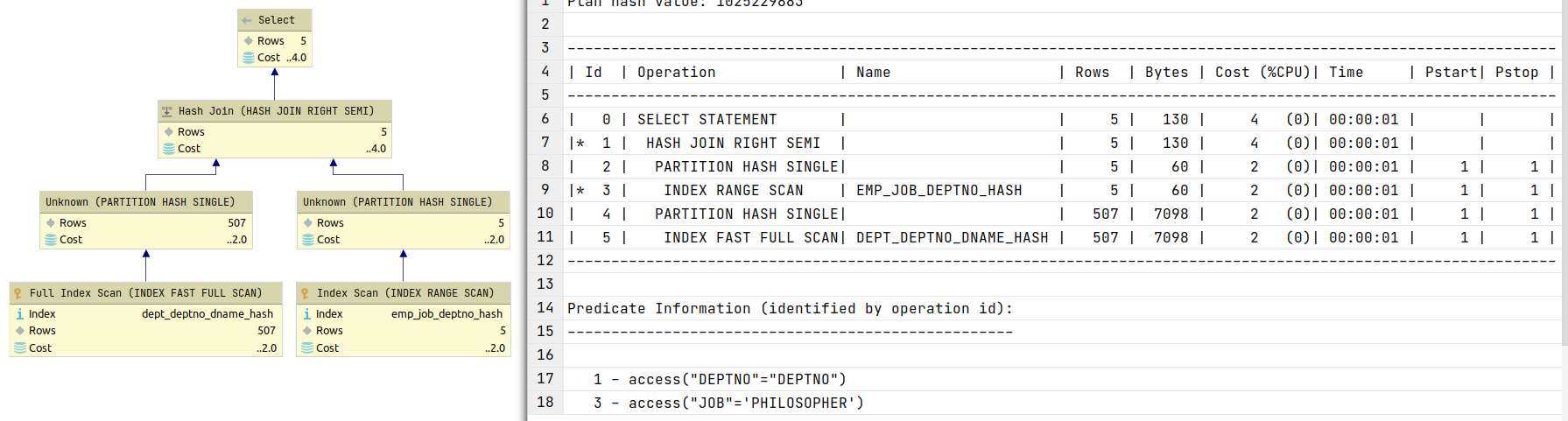
**Deptno,Job HASH index Hash Join**

****

**Deptno,Dname HASH index Hash Join**

****

**Job,Deptno HASH index + Deptno,Dname HASH index Hash Join**

****

**Query #6**

Si definiscano una o più strutture secondarie (indici) che permettano l’ottimizzazione della seguente

query (rimuovere eventuali indici già esistenti per confrontare le performance della query con e senza

indici):

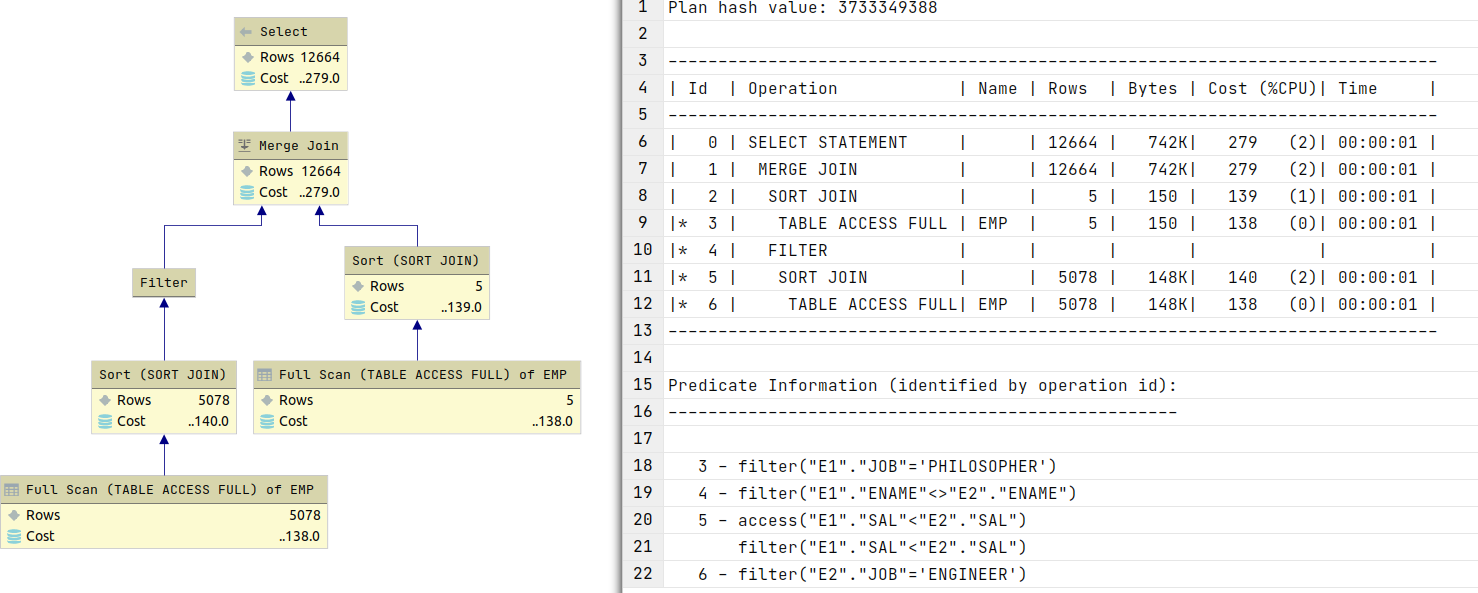
select e1.ename, e1.empno, e1.sal, e2.ename, e2.empno, e2.sal

from emp e1, emp e2

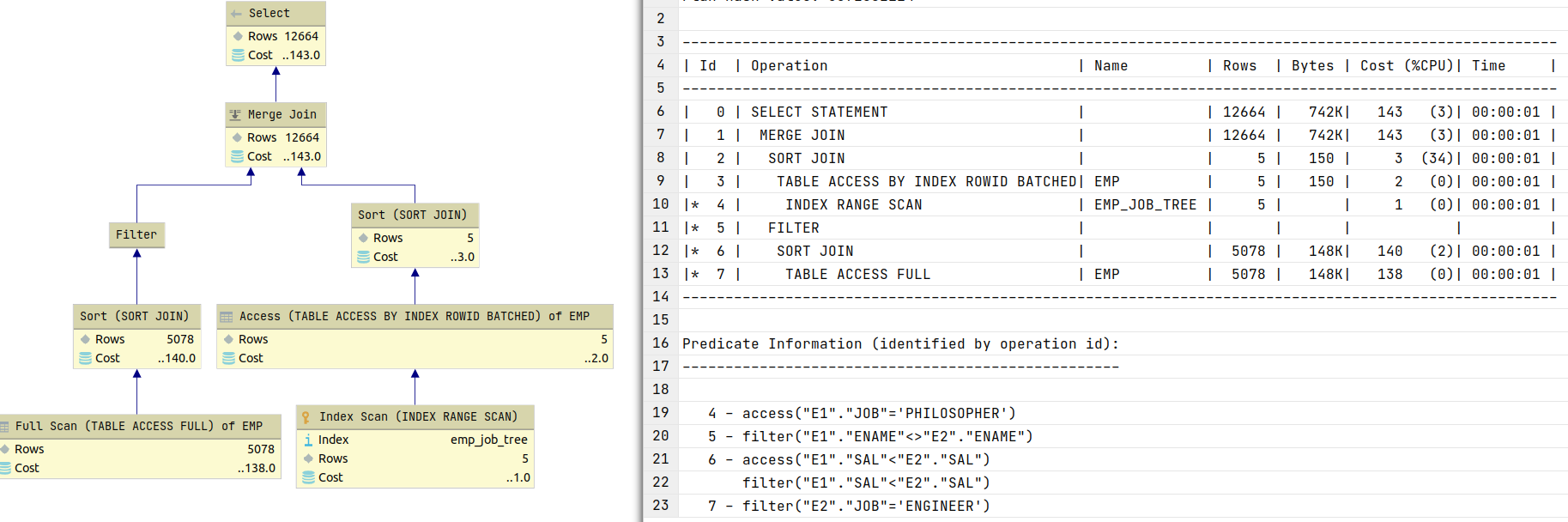
where e1.ename <> e2.ename and e1.sal < e2.sal

and e1.job = 'PHILOSOPHER' and e2.job = 'ENGINEER';

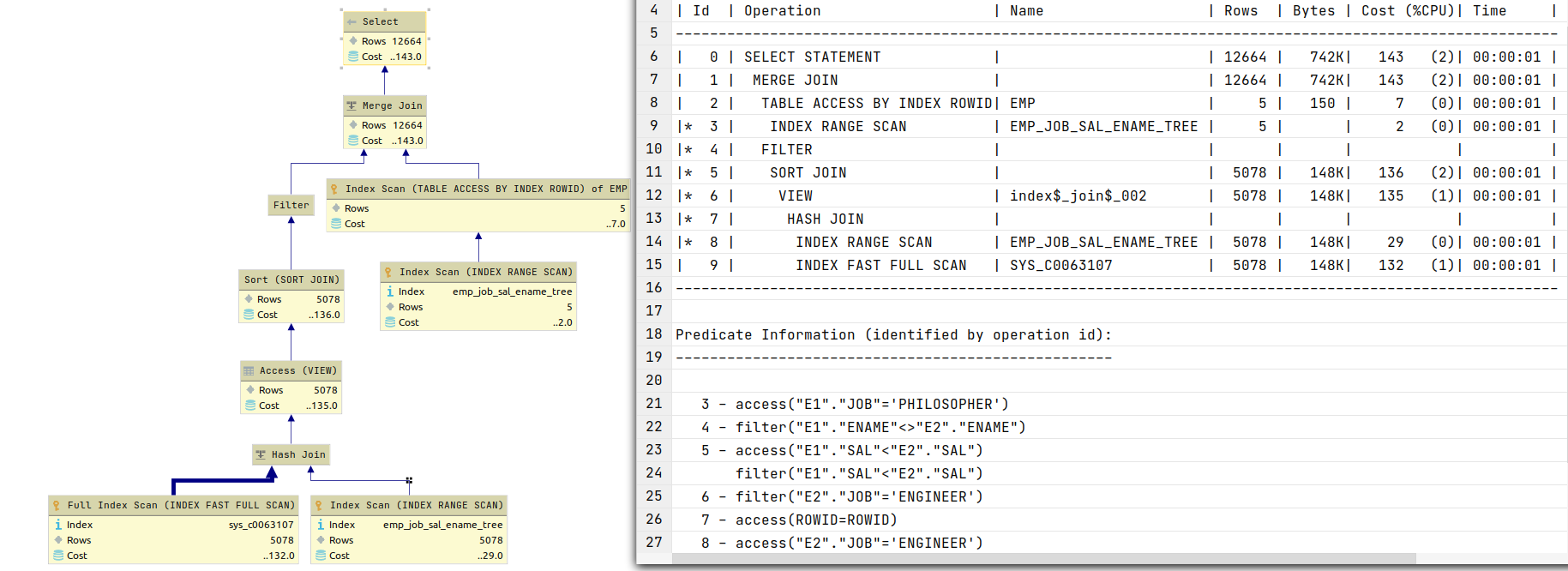
**Default o SAL + Ename B-tree index**



**Job B-Tree index**

****

**Job,Sal,Ename B-Tree index**

****