|  |
| --- |
| EPAM Systems, RD Dep.  MTN.\*NIX.07 Oracle DB. Introduction to DWH |
| MTN.\*NIX.07 Labs - Access and Join Methods Part 2 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| REVISION HISTORY | | | | | |
| Ver. | Description of Change | Author | Date | Approved | |
| Name | Effective Date |
| 1.0 | Initial status of document | [**Kiryl Bucha**](mailto:Kiryl_bucha@epam.com) | 16-JAN-2012 |  |  |
|  |  |  |  |  |  |

*Contents*

[1. Auto Trace & Explain Plan 3](#_Toc315038644)

[1.1. Task 1: Auto Trace configuration training 3](#_Toc315038645)

[2. Join Methods 3](#_Toc315038646)

[2.1. Task 2: Nested Loops Joins 4](#_Toc315038647)

[2.2. Task 3: Sort-Merge Joins 4](#_Toc315038648)

[2.3. Task 4: Hash Joins 4](#_Toc315038649)

[2.4. Task 5: Cartesian Joins 4](#_Toc315038650)

[2.5. Task 6: Left/Right Outer Joins 4](#_Toc315038651)

[2.6. Task 7: Full Outer Join 4](#_Toc315038652)

[2.7. Task 8: Semi Joins 4](#_Toc315038653)

[2.8. Task 9: Anti Joins 5](#_Toc315038654)

[2.9. Task 10: Prepare summary table 5](#_Toc315038655)

# Auto Trace & Explain Plan

# Join Methods

## Task 2: Nested Loops Joins

SELECT \* */\*+ USE\_NL(e d) \*/*

FROM emp e, dept d

WHERE E.DEPTNO = D.DEPTNO

and E.COMM between 1 and 100;



## Task 3: Sort-Merge Joins

SELECT \* */\*+ USE\_MERGE(e d) \*/*

FROM emp e, dept d

WHERE E.DEPTNO = D.DEPTNO;



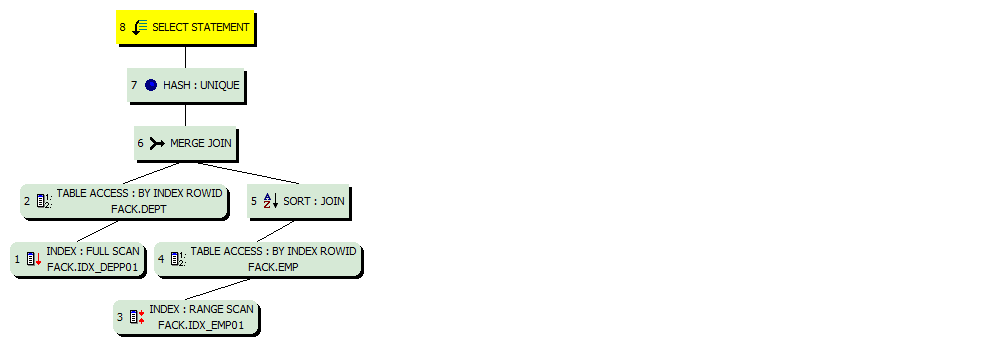
## Task 4: Hash Joins

SELECT DISTINCT \* */\*+ USE\_HASH(e d) \*/*

FROM emp e, dept d

WHERE E.DEPTNO = D.DEPTNO

and E.EMPNO >100;



## Task 5: Cartesian Joins

SELECT */\*+ USE\_HASH(t1 t2) \*/* distinct \*

FROM t3 t1, t4 t2*/\*+ USE\_HASH(t1, t2) \*/*



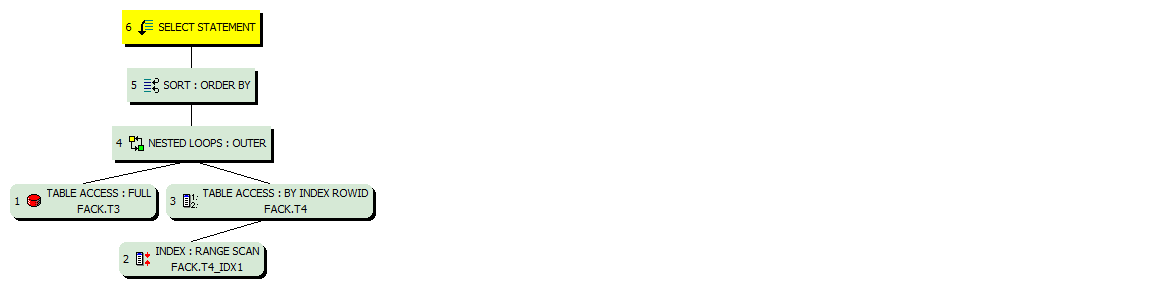
## Task 6: Left/Right Outer Joins

SELECT */\*+ use\_nl(t3 t4) \*/* \*

FROM t3 left join t4 on T3.ID = T4.ID

where T3.T\_PAD !=T3.ID/5

order by 1;



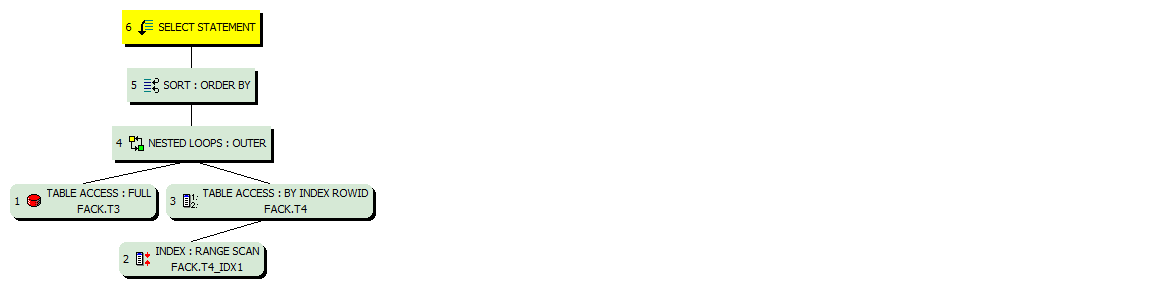
## Task 7: Full Outer Join

SELECT */\*+ use\_nl(t3 t4) \*/* \*

FROM t3 full join t4 on T3.ID = T4.ID

where T3.T\_PAD !=T3.ID/5

order by 1;



## Task 8: Semi Joins

SELECT \*

FROM t3

WHERE t3.id IN

( SELECT id FROM t4 );



**Note:**  If you would like change in execution plan the type of join method use oracle performance hints.

1. SEMIJOIN – perform a semi-join (the optimizer gets to pick which kind)
2. NO\_SEMIJOIN – obviously means don’t perform a semi-join
3. NL\_SJ – perform a nested loops semi-join (deprecated as of 10g)
4. HASH\_SJ – perform a hash semi-join (deprecated as of 10g)
5. MERGE\_SJ – perform a merge semi-join (deprecated as of 10g)

## Task 9: Anti Joins

SELECT \*

FROM t3

WHERE t3.id not IN

( SELECT id FROM t4 );



## Task 10: Prepare summary table

**Task:** Make comparison of all possible variant of join methods and join access methods and fill the table below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Join Access “A” | Join Access “B” | Nested Loop | Hash Join | Sort-Merge Join | Anti-Join | Semi-Join |
| Small Table | Small Table |  |  |  |  |  |
| Small Table | Indexed Small Table |  |  |  |  |  |
| … |  |  |  |  |  |  |