**Legal Notice:** This document contains privileged and/or confidential information and may not be disclosed, distributed or reproduced without the prior written permission of Datamola.

U1M5.LW.Access and Join Methods Part 2

#### Kiryl Bucha

Contents

[1. Auto Trace & Explain Plan 3](#_Toc43832198)

[1.1. Task 1: Auto Trace configuration training 3](#_Toc43832199)

[2. Join Methods 4](#_Toc43832200)

[2.1. Task 2: Nested Loops Joins 4](#_Toc43832201)

[2.2. Task 3: Sort-Merge Joins 4](#_Toc43832202)

[2.3. Task 4: Hash Joins 5](#_Toc43832203)

[2.4. Task 5: Cartesian Joins 5](#_Toc43832204)

[2.5. Task 6: Left/Right Outer Joins 5](#_Toc43832205)

[2.6. Task 7: Full Outer Join 5](#_Toc43832206)

[2.7. Task 8: Semi Joins 5](#_Toc43832207)

[2.8. Task 9: Anti Joins 6](#_Toc43832208)

[2.9. Task 10: Prepare summary table 6](#_Toc43832209)

# 1. Auto Trace & Explain Plan

## 1.1. Task 1: Auto Trace configuration training

Below all possible variants of SQL plus utilities autotrace:

set autotrace off

set autotrace on

set autotrace traceonly

set autotrace on explain

set autotrace on statistics

set autotrace on explain statistics

set autotrace traceonly explain

set autotrace traceonly statistics

set autotrace traceonly explain statistics

set autotrace off explain

set autotrace off statistics

set autotrace off explain statistics

**NOTE:** If you received next error: Check PLUSTRACE role is enabled. Please make next steps:

1. Run next script connected as sysdba:

# @ $ORACLE\_HOME/sqlplus/admin/plustrce.sql;

1. Grant role PLUSTRACE to $UserName$

# grant plustrace to $UserName$;

**Task Results:**

Expected:

Summary table with all result and text description of analyses this results.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Auto Trace Configuration Options | Expected Results | Description |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 2. Join Methods

**The Main Task** is to create SQL and prepare execution plan of statements with join methods on Task Topics (Task 2 - 9)

**Task Results:**

There are several tasks below with the same main expected result points:

* Create SQL using next tables: scott.emp, scott.dept
* Create additional needed Tables and Indexes
* Prepare screenshots of execution plan

## 2.1. Task 2: Nested Loops Joins

**Example:**

# SELECT \*

FROM scott.emp e, scott.dept d

WHERE e.deptno = d.deptno

AND d.deptno = 10

**Task:** Prepare SQL **explain plan** using: SQL plus Auto Trace Utility.

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

## 2.2. Task 3: Sort-Merge Joins

**Task:** Prepare SQL **explain plan** using: SQL plus Auto Trace Utility.

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

## 2.3. Task 4: Hash Joins

**Task:** Prepare SQL **explain plan** using software: Oracle SQL Developer.

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

## 2.4. Task 5: Cartesian Joins

**Task:** Prepare SQL **explain plan** using software: Oracle SQL Developer.

## 2.5. Task 6: Left/Right Outer Joins

**Tasks:**

* Prepare SQL **trace protocol** using software: Oracle SQL Developer.
* Create SQL left outer join
* Create SQL right outer join

## 2.6. Task 7: Full Outer Join

**Task:** Prepare SQL **explain plan** using: SQL plus Auto Trace Utility.

## 2.7. Task 8: Semi Joins

**Task:** Prepare All possible variants of SEMI JOIN SQL **explain plan** using: SQL plus Auto Trace Utility.

**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)

## 2.8. Task 9: Anti Joins

**Task:** Prepare All possible variants of ANTI JOIN SQL **explain plan** using: SQL plus Auto Trace Utility.

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

1. ANTIJOIN – perform an anti-join (the optimizer gets to pick which kind)
2. USE\_ANTI – older version of ANTIJOIN hint
3. NL\_AJ – perform a NESTED LOOPS anti-join (deprecated as of 10g)
4. HASH\_AJ – perform a HASH anti-join (deprecated as of 10g)
5. MERGE\_AJ – perform a MERGE anti-join (deprecated as of 10g)

## 2.9. 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 |  |  |  |  |  |
| … |  |  |  |  |  |  |