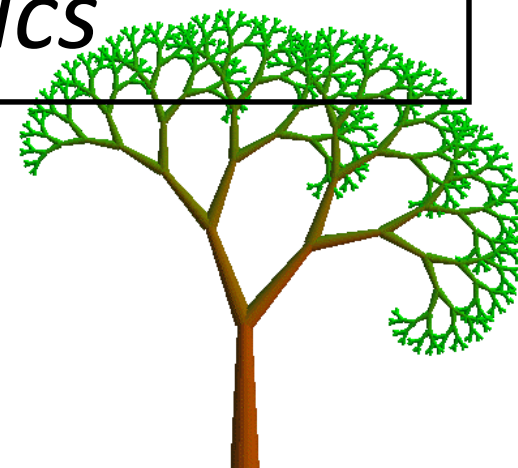




Query Tuning Foundations

Execution Plan

RowSource Statistics





Execution Plan

- The execution plan of a SQL statement is composed of small building blocks called “row sources.”
- The combination of row sources for a statement is called the execution plan.
- By using parent-child relationships, the explain plan can be displayed in a tree (Text or Graphical)

SQLs : Non-Recursive & Recursive

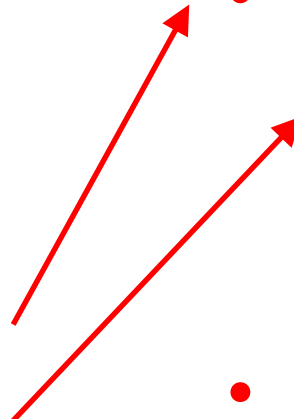
- User SQL
- User SQL
- User SQL
- Commit



- User SQL
 - Internal SQL
 - Internal SQL
- User SQL
- User SQL
 - Internal SQL
 - Internal SQL
 - Internal SQL
 - Internal SQL
- Commit

Non-Recursive = (depth = 0)

Recursive = (depth > 0)





SQLs : Database Calls

- User SQL
 - Internal SQL
 - Internal SQL
- 
- User SQL
 - **Parse**
 - Internal SQL
 - Parse
 - **Exec**
 - Fetch
 - Internal SQL
 - Exec
 - Fetch
 - Exec
 - **Fetch**



SQLs : Waits Events

- SQL
 - Parse
 - Exec
 - Fetch
 - Fetch
- SQL

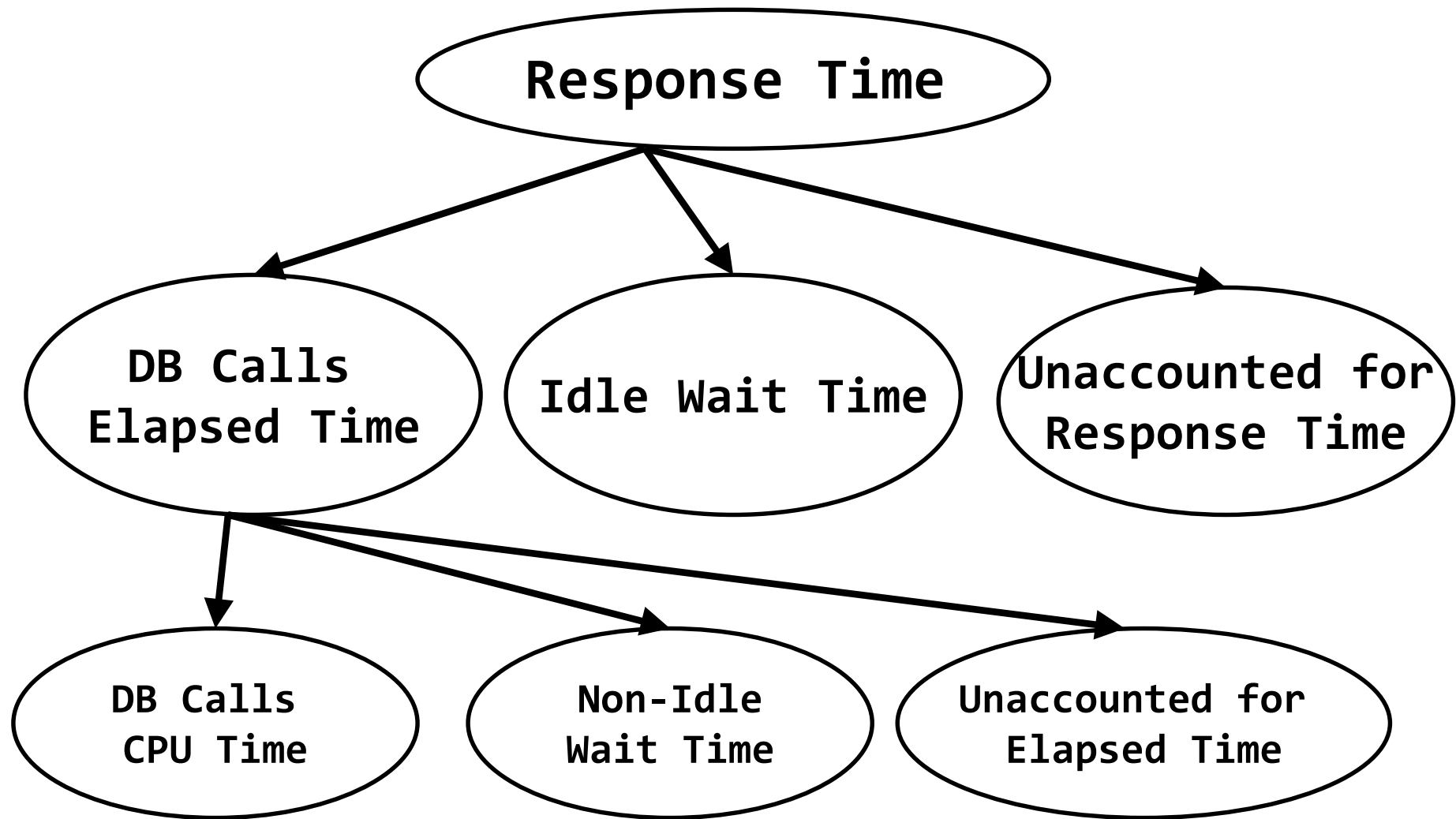


- SQL
 - Parse
 - Exec
 - Fetch
 - Db file sequential read
 - Db file sequential read
 - SQL*Net message to client
 - Fetch
- SQL*Net message to client
- SQL

Non Idle Within a DB call
Idle = between DB calls



Response Time Components





Response Time Components

- Elapsed Time (from DB Calls)
 - CPU Time (from DB Calls)
 - Non-Idle Wait Time (within DB Calls)
 - Elapsed Time Unaccounted-for
- Idle Wait Time (between DB Calls)
- Response Time Unaccounted-for



Zoom RAW: Trace (01) 10046

(Entête / Pied)

Dump file c:\oracle\product\10.2.0\admin\fry\udump\fry_ora_3460.trc

Oracle Database 10g Enterprise Edition Release 10.2.0.4.0 - Production...

Windows XP Version V5.1 Service Pack 3

Windows thread id: 3460, image: ORACLE.EXE (SHAD)

*** 2009-08-21 10:41:34.250

*** ACTION NAME:() 2009-08-21 10:41:34.234 *DBMS_APPLICATION_INFO*

*** MODULE NAME:(SQL*Plus) 2009-08-21 10:41:34.234 *DBMS_APPLICATION_INFO*

*** SERVICE NAME:(SYS\$USERS) 2009-08-21 10:41:34.234

*** SESSION ID:(150.22) 2009-08-21 10:41:34.234 *N° session tracée*

Vérifier la fin du fichier :

*** DUMP FILE SIZE IS LIMITED TO 10240000 BYTES *** ????



Zoom RAW: Trace 10046 (DB calls)

PARSING IN CURSOR #1 len=68 dep=0 uid=47 oct=3 lid=47 tim=1045378263 hv=3063997299 ad='1d72e198'

select 'Bonjour, nous sommes Le '||sysdate from dept where deptno=10

END OF STMT

PARSE #1: c=0,e=91,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,tim=1045378258

BINDS #1:

EXEC #1: c=0,e=154,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,tim=1045379118

WAIT #1: nam='SQL*Net message to client' ela= 4 driver id=1111838976 #bytes=1 p3=0 obj#=-1 tim=1045379259

FETCH #1: c=0,e=44,p=0,cr=1,cu=0,mis=0,r=1,dep=0,og=1,tim=1045379488

WAIT #1: nam='SQL*Net message from client' ela= 247 driver id=1111838976 #bytes=1 p3=0 tim=1045379900

FETCH #1: c=0,e=3,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=0,tim=1045380059

WAIT #1: nam='SQL*Net message to client' ela= 3 driver id=1111838976 #bytes=1 p3=0 obj#=-1 tim=1045380194

WAIT #1: nam='SQL*Net message from client' ela= 175987 driver id=1111838976 #bytes=1 p3=0 tim=1047140216

STAT #1 id=1 cnt=1 pid=0 pos=1 obj=47187 op='INDEX UNIQUE SCAN PK_DEPT (cr=1 pr=0 pw=0 time=26 us)'

XCTEND rlbk=0, rd_only=1



Detail : PARSE, EXEC, FETCH

PARSE #1: c=0, e=91, p=0, cr=0, cu=0, mis=0, r=0, dep=0, og=1, tim=1045378258

EXEC #1: c=0, e=154, p=0, cr=0, cu=0, mis=0, r=0, dep=0, og=1, tim=1045379118

FETCH #1: c=0, e=44, p=0, cr=1, cu=0, mis=0, r=1, dep=0, og=1, tim=1045379488

c= 0 CPU time in microseconds, rounded to nearest 1/100 of a second
e= 154 Elapsed time in microseconds
p=0 Physical reads
cr=0 consistent reads
cu=0 current mode gets
mis=0 Library Cache misses. If 1, means we just had a hard parse.
r=0 rows processed, .i.e., returned
dep=0 recursive depth
og=1 Optimizer mode: 1= All_Rows; 2= First_Rows, 3=Rule, 4= Choose.
tim= 1045379118 The ubiquitous timestamp, in microseconds. Time represents the time at the end of the call.

Zoom RAW : Trace 10046 (Others - 39817.1)

PARSING IN CURSOR #1 len=68 dep=0 uid=47 oct=3 lid=47 tim=1045378263 hv=3063997299 ad='1d72e198'

select 'Bonjour, nous sommes Le '||sysdate from dept where deptno=10

END OF STMT

PARSE #1:c=0,e=91,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,tim=1045378258

BINDS #1:

EXEC #1:c=0,e=154,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,tim=1045379118

WAIT #1: nam='SQL*Net message to client' ela= 4 driver id=1111838976 #bytes=1 p3=0 obj#=-1 tim=1045379259

FETCH #1:c=0,e=44,p=0,cr=1,cu=0,mis=0,r=1,dep=0,og=1,tim=1045379488

WAIT #1: nam='SQL*Net message from client' ela= 247 driver id=1111838976 #bytes=1 p3=0 tim=1045379900

FETCH #1:c=0,e=3,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=0,tim=1045380059

WAIT #1: nam='SQL*Net message to client' ela= 3 driver id=1111838976 #bytes=1 p3=0 obj#=-1 tim=1045380194

WAIT #1: nam='SQL*Net message from client' ela= 175987 driver id=1111838976 #bytes=1 p3=0 tim=1047140216

STAT #1 id=1 cnt=1 pid=0 pos=1 obj=47187 op='INDEX UNIQUE SCAN PK_DEPT (cr=1 pr=0 pw=0 time=26 us)'

XCTEND rlbk=0, rd_only=1



Detail : PARSING IN CURSOR

- Appears when a statement is not already printed in the trace when a PARSE, EXEC, FETCH operation is about to be printed

```
PARSING IN CURSOR #1 len=68 dep=0 uid=47 oct=3 lid=47 tim=1045378263  
hv=3063997299 ad='1d72e198'  
select 'Bonjour, nous sommes le '||sysdate from dept where deptno=10  
END OF STMT
```

CURSOR #1 Gets reused within this session. Enables association of events with each cursor.

len=68 Length of the SQL text

dep=0 Depth.

oct=3 Oracle Command Type. Type of SQL operation. 3= SELECT,
see V\$SESSION in the Database Reference

lid=47 User ID of the user who first parsed the statement

tim= 1045378263 microsecond time since some arbitrary value. Not clock or date time.

hv= 3063997299 Hash value of SQL string. Used to get hash bucket for Library Cache. Not
necessarily unique, unless combined with address. Repeatable value across
platforms for a given Oracle version.

ad= '1d72e198'Address of this object's handle in the Library Cache.

Combined with hash value, this gives an absolute unique identifier of this object
in the Library Cache. Established dynamically when cursor is first created.



Detail : Binds if level 4 or 12 chosen

BINDS #1: Bind#0

oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=08228e44 bln=22 avl=04 flg=05
value=47186

oacdty =2	Data type
mxl=22(22)	Maximum length, private max length (may not always apply)
mal=00	Total # of array elements
scl=00	Scale, for datetime and interval types
pre=00	Precision, also for datetime and interval types
oacflg=13	Bitmask for various things (internal uses)
fl2=1	Extension of oacflg
size=24	Size and offset identify location of value for this bind variable
kxsbbbf=0555ded8	UGA buffer pointer
bln=22	Buffer length
avl=02	Value length
flg=05	Bitmask, (internal uses)
value=2	Actual value of bind variable



Detail : Wait Events

- Wait events always occur **BEFORE** the action it relates to is reported
- If you see the event printed, it already happened
- Always look for the following call with the same cursor # as the wait event

WAIT #1: nam='SQL*Net message from client' ela= 247 driver id=1111838976 #bytes=1 p3=0 tim=1045379900

nam='SQL*Net...'	Name of event, see v\$event_name
ela= 247	Elapsed time, in microseconds
p1=1111838976	P2, P3 are specific to the event
p2=1	Refer to WebIV for event of interest
p3=0	



Zoom RAW : Trace 10046

(Wait Events – Idle vs NonIdle)

```
PARSING IN CURSOR #2 len=57 dep=0 uid=47 oct=3 lid=47 tim=2769298867 hv=4052346060 ad='1d6c14d0'
select 'Hi, I come from '||deptno||'    '||dname from dept
END OF STMT
PARSE #2:c=0,e=5683,p=0,cr=3,cu=0,mis=1,r=0,dep=0,og=1,tim=2769298863
BINDS #2:
EXEC #2:c=0,e=211,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,tim=2769299946
WAIT #2: nam='SQL*Net message to client' ela= 11 driver id=1111838976 #bytes=1 p3=0 obj#=-1 tim=2769300088
WAIT #2: nam='db file sequential read' ela= 61034 file#=4 block#=298163 blocks=1 obj#=47186 tim=2769361787
WAIT #2: nam='db file scattered read' ela= 1517 file#=4 block#=298164 blocks=5 obj#=47186 tim=2769363749
FETCH #2:c=0,e=63720,p=6,cr=3,cu=0,mis=0,r=1,dep=0,og=1,tim=2769364000
WAIT #2: nam='SQL*Net message from client' ela= 379 driver id=1111838976 #bytes=1 p3=0 obj#=47186 tim=2769364551
WAIT #2: nam='SQL*Net message to client' ela= 3 driver id=1111838976 #bytes=1 p3=0 obj#=47186 tim=2769365035
FETCH #2:c=0,e=209,p=0,cr=5,cu=0,mis=0,r=3,dep=0,og=1,tim=2769365222
WAIT #2: nam='SQL*Net message from client' ela= 5951350 driver id=1111838976 #bytes=1 p3=0 obj#=47186
tim=2775316803
STAT #2 id=1 cnt=4 pid=0 pos=1 obj=47186 op='TABLE ACCESS FULL DEPT (cr=8 pr=6 pw=0 time=63692 us)'
XCTEND rlbk=0, rd_only=1
```



Zoom : Trace 10046 (Recursive)

PARSING IN CURSOR #1 len=210 **dep=1** uid=0 oct=3 lid=0 tim=2769294703 hv=864012087 ad='21268d5c'

select /*+ rule */ bucket_cnt, row_cnt, cache_cnt, null_cnt, timestamp#, sample_size, minimum, maximum,
distcnt, lowval, hival, density, col#, spare1, spare2, avgcln from hist_head\$ where obj#=:1 and intcol#=:2

END OF STMT

PARSE #1:c=0,e=28,p=0,cr=0,cu=0,mis=0,r=0,**dep=1**,og=3,tim=2769294695

BINDS #1:

Bind#0

oacdtty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00 fl2=0001 ... value=47186

Bind#1 ...

EXEC #1:c=0,e=1708,p=0,cr=0,cu=0,mis=0,r=0,**dep=1**,og=3,tim=2769297372

FETCH #1:c=0,e=122,p=0,cr=3,cu=0,mis=0,r=1,**dep=1**,og=3,tim=2769297623

STAT #1 id=1 cnt=1 pid=0 pos=1 obj=255 op='TABLE ACCESS BY INDEX ROWID HIST_HEAD\$ (cr=3 pr=0 time=127 us)'

STAT #1 id=2 cnt=1 pid=1 pos=1 obj=257 op='INDEX RANGE SCAN I_HH_OBJ#_INTCOL# (cr=2 pr=0 pw=0 time=77 us)'

=====

PARSING IN CURSOR #2 len=57 **dep=0** uid=47 oct=3 lid=47 tim=2769298867 hv=4052346060 ad='1d6c14d0'

select 'Hi, I come from '||deptno||' '||dname from dept

END OF STMT

PARSE #2:c=0,e=5683,p=0,cr=3,cu=0,mis=1,r=0,**dep=0**,og=1,tim=2769298863



10046 Tracing: Potential Problems

- Due to Collection Methods
 - Trace activated during long running call
 - Compute unaccounted time for first call in trace
 - Trace deactivated during long running call
 - Compute missing time at end of trace
 - Trace activated/deactivated during a recursive call
 - Look for missing child calls associated with a parent; or, missing parents associated with a child
- Due to unaccounted time



10046 Tracing: Potential Problems

- Due to Architecture
 - Shared Server and Connection Pooling
- Set DBMS_SESSION.SET_IDENTIFIER at the beginning of your session (via logon trigger, OCI or JDBC)
- Instrument your code using DBMS_APPLICATION_INFO
- Turn on Statistics and tracing using DBMS_MONITOR
- Use TRCSESS if required to consolidate trace files
- Use RAW trace file, tkprof or another parser to format the output file



DBMS_XPLAN summary

```
select t1.*  
from emp t2, emp t1  
where t1.empno = t2.empno  
      and t1.sal > 500  
      and t2.empno between 5000 and 9000
```

With Statistics_level=ALL



DBMS_XPLAN summary

The first section of output is as follows:

SQL_ID 392agcpnsb5qy, child number 0

```
select t1.* from emp t2 , emp t1 where
  t1.empno = t2.empno and t1.sal > 500 and
  t2.empno between 5000 and 9000
```



DBMS_XPLAN summary

The second section shows

the hash value of the execution plan and, in a table, the execution plan itself.

Plan hash value: 866206949

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	

0	SELECT STATEMENT		14	574	2 (0)	
1	NESTED LOOPS		14	574	2 (0)	
*	TABLE ACCESS BY INDEX ROWID	EMP	14	518	2 (0)	
*	INDEX RANGE SCAN	PK_EMP	14		1 (0)	
*	INDEX UNIQUE SCAN	PK_EMP	1	4	0 (0)	



DBMS_XPLAN summary

The third section shows

the query block names and the object aliases:

Query Block Name / Object Alias (identified by operation id):

```
1 - SEL$1
2 - SEL$1 / T1@SEL$1
3 - SEL$1 / T1@SEL$1
4 - SEL$1 / T2@SEL$1
```



DBMS_XPLAN summary

The fourth section is available

only as of Oracle Database 10g Release 2. It shows the set of hints that are necessary to force that particular execution plan. This set of hints is called outline.

Outline Data

```
/*+  BEGIN_OUTLINE_DATA
      IGNORE_OPTIM_EMBEDDED_HINTS
      OPTIMIZER_FEATURES_ENABLE('10.2.0.4')
      ALL_ROWS
      OUTLINE_LEAF(@"SEL$1")
      INDEX_RS_ASC(@"SEL$1" "T1"@"SEL$1" ("EMP"."EMPNO"))
      INDEX(@"SEL$1" "T2"@"SEL$1" ("EMP"."EMPNO"))
      LEADING(@"SEL$1" "T1"@"SEL$1" "T2"@"SEL$1")
      USE_NL(@"SEL$1" "T2"@"SEL$1")
      END_OUTLINE_DATA
*/
```



DBMS_XPLAN summary

The fifth section shows

which predicates are applied. For each of them, it is shown where (line) and how (access or filter) they are applied.

Predicate Information (identified by operation id):

```
2 - filter("T1"."SAL">500)
3 - access("T1"."EMPNO">=5000 AND "T1"."EMPNO"<=9000)
4 - access("T1"."EMPNO"="T2"."EMPNO")
    filter(("T2"."EMPNO">=5000 AND "T2"."EMPNO"<=9000))
```



DBMS_XPLAN summary

The following section shows

which columns are returned as output when each operation is executed. It is available as of Oracle Database 10g only.

Column Projection Information (identified by operation id):

- 1 - "T1"."EMPNO"[NUMBER,22], "T1"."ENAME"[VARCHAR2,10], "T1"."JOB"[VARCHAR2,9],
"T1"."MGR"[NUMBER,22], "T1"."HIREDATE"[DATE,7], "T1"."SAL"[NUMBER,22],
"T1"."COMM"[NUMBER,22], "T1"."DEPTNO"[NUMBER,22]
- 2 - "T1"."EMPNO"[NUMBER,22], "T1"."ENAME"[VARCHAR2,10], "T1"."JOB"[VARCHAR2,9],
"T1"."MGR"[NUMBER,22], "T1"."HIREDATE"[DATE,7], "T1"."SAL"[NUMBER,22],
"T1"."COMM"[NUMBER,22], "T1"."DEPTNO"[NUMBER,22]
- 3 - "T1".ROWID[ROWID,10], "T1"."EMPNO"[NUMBER,22]



DBMS_XPLAN summary

Finally, there is a section

providing notes and warnings about the optimization phase, the environment, or the SQL statement itself:

Note

- Warning: basic plan statistics not available.

These are only collected when:

- * hint 'gather_plan_statistics' is used for the statement or
- * parameter 'statistics_level' is set to 'ALL', at session or system level