

SQL Trace в Oracle

Egor Starostin, DBA

July 11, 2019

- моя первая версия Oracle 7.2, в 1996
- DBA с 1998
- <http://oracledba.ru>
- twitter: @egorst
- <https://github.com/egorst/sqltrace-slides/raw/master/sqltrace.pdf>

О чём этот доклад?

“... все компиляторы должны предоставлять программистам возможность понять, какие части их кода выполняются дольше всего”

Donald Knuth, "Structured Programming with go to Statements", 1974

Не искать ключи под фонарём



Oracle Diagnostic Events (не те, что Wait Events)

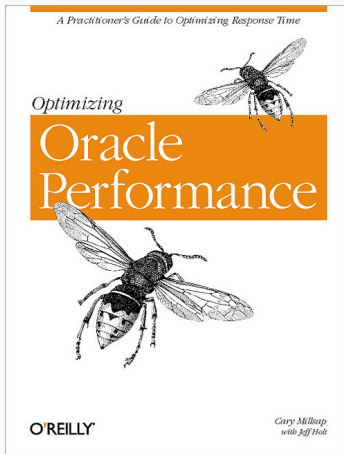
- `$ORACLE_HOME/rdbms/mesg/oraus.msg`

```
/ Pseudo-error debugging events:  
/   Error codes 10000 .. 10999 are reserved for debug event codes  
/   that are not really errors.  
[...]  
10046, 00000, "enable SQL statement timing"
```

- EVENT: 10046 "enable SQL statement tracing (including binds/waits)"
(Doc ID 21154.1)

В начале была книга

Optimizing Oracle Performance
by Cary Millsap and Jeff Holt
O'Reilly Media, 2003



Enable trace

- **по-старинке**

```
alter session set events '10046 trace name context forever, level 8'
```

- **в своей сессии**

```
exec dbms_session.session_trace_enable(waits, binds, plan_stat)
```

- **в другой сессии**

```
exec dbms_monitor.session_trace_enable(sid, serial#, waits, binds, plan_stat)
-- select * from dba_enabled_traces;
```

- **по service_name/module/action**

```
exec dbms_monitor.serv_mod_act_trace_enable( -
    service_name, module_name, action_name, waits, binds, instance_name, plan_stat)
```

- **точно**

```
alter system set events 'sql_trace[sql: sql_id=9tz4qu4rj9rdp|grwydz59pu6mc] -
    wait=true, binds=false, plan_stat=adaptive'
```

```
alter system set events 'sql_trace{process: pname=smon|p0000} wait=true'
```

```
alter system set events 'sql_trace[sql: sql_id=9tz4qu4rj9rdp]{process:12345}'
```

Disable trace

```
alter session set events '10046 trace name context off'
```

```
exec dbms_monitor.session_trace_disable(sid, serial#)
```

```
alter system set events 'sql_trace[sql: sql_id=9tz4qu4rj9rdp|grwydz59pu6mc] off'
```


Как сделать трассировку еще проще

- `dbms_application_info.set_action(action_name)`
- `dbms_application_info.set_module(module_name, action_name)`
- `dbms_session.set_identifier(client_id)`

Dude, Where's My Trace?

```
select value from v$diag_info where name = 'Default Trace File';
```

```
select tracefile from v$process where addr=(  
    select paddr from v$session where sid=sys_context('userenv','sid')  
);
```

```
alter system set "_trace_files_public"=true scope=spfile; -- on dev machine
```

```
select translate(payload, 'x'||chr(10), 'x') payload  
from v$diag_trace_file_contents  
where adr_home || '/trace/' || trace_filename = (  
    select tracefile  
    from v$process  
    where addr = (  
        select paddr from v$session where sid = sys_context('userenv','sid')  
    )  
)  
order by line_number;
```

И какой уровень трассировки выставлять?

- Обычно 8-ой (waits=true)
- 12-й (binds=true) и +16-й (plan_stat='all_executions') уровни изредка могут и тормозить

И насколько всё тормозит от трассировки?

На -10% и менее процентов

SQL запрос с высоты птичьего полета

Стадия запроса

Trace Level

• PARSE	1
• BINDS	4
• EXEC	1
• WAIT	8
• FETCH	1
• STAT	1, 16, 32, 64
• CLOSE	1

Что в trace-файле?

- Разбивка запроса на стадии PARSE / EXEC / FETCH / BINDS / WAIT / STAT

- Информация о CPU Time, Elapsed Time, Timestamp

PARSE #1: c=311, e=311, p=0, cr=0, cu=0, mis=1, r=0, dep=1, og=4, plh=0, tim=745332470

- Информация об иерархии запросов

PARSE #1: c=311, e=311, p=0, cr=0, cu=0, mis=1, r=0, dep=1, og=4, plh=0, tim=745332470

- Информация о dba_objects.object_id

WAIT #1: nam='db file scattered read' ela= 9426 file#=3 block#=23883 blocks=5
obj#=21722 tim=745332470

Что делать с trace-файлом?

- **tkprof**

`tkprof sort=prsela,exeela,fchela file.trc file.out`

- **Method R Software**

<https://method-r.com/software/>

- **tvd\$xtat**

<https://antognini.ch>

- **orasrp**

<http://oracledba.ru/orasrp>

Что можно извлечь из trace-файла? Flat Profile

Session Flat Profile

=====						
Event Name	% Time	Seconds	Calls	Time Per Call		
				Avg	Min	Max
db file sequential read	47.4%	1.0842s	1,959	0.0006s	0.0003s	0.0105s
unaccounted-for time	21.9%	0.5005s				
FETCH calls [CPU]	17.3%	0.3961s	2,347	0.0002s	0.0000s	0.3812s
PARSE calls [CPU]	10.8%	0.2464s	13	0.0190s	0.0000s	0.2360s
PGA memory operation	1.1%	0.0250s	326	0.0001s	0.0000s	0.0198s
EXEC calls [CPU]	0.9%	0.0211s	59	0.0004s	0.0000s	0.0098s
Disk file operations I/O	0.6%	0.0136s	3	0.0045s	0.0000s	0.0135s
SQL*Net message from client [idle]	0.1%	0.0014s	2	0.0007s	0.0004s	0.0010s
asynch descriptor resize	0.0%	0.0000s	1	0.0000s	0.0000s	0.0000s
SQL*Net message to client	0.0%	0.0000s	2	0.0000s	0.0000s	0.0000s

Total	100.0%	2.2883s				

Что можно извлечь из trace-файла? Call Graph

Session Call Graph

SQL Hash Value	Branch %	Total %	Self Seconds	Cumulative Seconds	Calls	Similar	Command Type
2466234169	94.6%	100.0%	2.1518s	2.2745s	3	0	SELECT
3433193208	1.8%	1.8%	0.0404s	0.0404s	4	0	SELECT
1411673910	1.1%	1.1%	0.0251s	0.0251s	2,283	0	SELECT
3461447171	0.2%	1.0%	0.0045s	0.0218s	3	0	SELECT
4126578894	0.8%	0.8%	0.0174s	0.0174s	40	0	SELECT
2195068792	0.9%	0.9%	0.0208s	0.0208s	3	0	SELECT
2045374727	0.1%	0.5%	0.0019s	0.0111s	7	0	SELECT
1641257687	0.4%	0.4%	0.0090s	0.0090s	2	0	SELECT
4126578894	0.0%	0.0%	0.0003s	0.0003s	2	0	SELECT
4126578894	0.1%	0.1%	0.0017s	0.0017s	27	0	SELECT
3849548163	0.0%	0.0%	0.0006s	0.0006s	22	0	SELECT
1641257687	0.0%	0.0%	0.0004s	0.0004s	5	0	SELECT
3901160166	0.0%	0.0%	0.0004s	0.0004s	7	0	SELECT
2880999359	0.0%	0.0%	0.0003s	0.0003s	9	0	SELECT
2217940283	100.0%	0.0%	0.0004s	0.0004s	2	0	ALTER SESSION

Что можно извлечь из trace-файла? Event Histograms

db file sequential read

value	Distribution	count
0.000128s		0
0.000256s	████████████████████████████████████████	1,300
0.000512s	████████████████	546
0.001024s	@	95
0.002048s		9
0.004096s		3
0.008192s		6
0.016384s		0

- Бесполезен для сессий в прошлом

Плюсы SQL Trace

- Непосредственное понимание того, что происходит в сессии
- Не нужна никакая дополнительная лицензия
- Возможность "reverse engineering"

Вопросы?