Triaging a Database Slowdown

NYOUG
December 6, 2007

Michael S. Abbey — Database Officer Oracle Practice

What You Need to Get Started

• Patience
• Diplomacy
• Warmth
• Kindness
• Tolerance
• Sense of humour
• Same old same old
• It's probably the CxO
• Swaddle your users
• Positive outlook
• High threshold
• Lighten up the dialogue

Michael S. Abbey — Triaging a Database Slowdown
No no no no no …

- Why are you calling me again; it was not the database the last 2,328,281,000,029,182,727 times!
- If I can find the time, I will check it out …
- Did you check the app-tier?
- Don't look at me; it's those b_____y developers.
- Call Miguel or Honorée

What You Really Need

- Metrics (if possible)
  - is the load average "high"
  - is the hourglass there a lot longer than usual?
  - on a scale of 1 to 10, how bad is the slowdown
- What is the caller doing with what system when the slowdown is noticed?
Help Translate

• There's something wrong with the database

• Can you work with development, if necessary, to find out why the applications are so slow?

Scenario

• CPU maxed out
• Queries seem to take forever
• Reporting performance is noticeably slower than usual
• Taking forever to log in
• Hourglass city
**UNIX**

`vmstat 2 15 : system/user/idle CPU`

---

**top** *(topas AIX)*: who are the biggest consumers
Windows

A Locking Problem
**In General** 10g/11g

```sql
select sid, blocking_session, username, blocking_session_status
from v$session
order by blocking_session;
```

Best MetaLink document

*Michael S. Abbey — Triaging a Database Slowdown*

**In General**

```
SQL> select * from dba_blockers;
HOLDING_SESSION
----------------------
1201
```

```
SQL> desc dba_blockers
Name          Type
------------- ------------
HOLDING_SESSION NUMBER
```

*Michael S. Abbey — Triaging a Database Slowdown*
In General

SQL> desc dba_waiters
Name Type
---------------- ------------------
WAITING_SESSION NUMBER
HOLDING_SESSION NUMBER 1201
LOCK_TYPE VARCHAR2(26)
MODE_HELD VARCHAR2(40)
MODE_REQUESTED VARCHAR2(40)
LOCK_ID1 NUMBER
LOCK_ID2 NUMBER

So, What is 1201 Up To?

select sql_text
from v$sqlarea
where address =
    (select sql_address
     from v$session
     where sid = 1201);
Caveats About That Query

- There may be more than 1 SQL statement associated with SID 1201
  - *single row subquery returns more than one row*
  - may have to change "=" to "in"
- May return *no rows selected*

Armed With / Can Get

- SID of blocking session
- SID of waiting session(s)
- SQL text associated with that blocking session
- SQL text associated with waiting session(s)
- Information about the environment of the blocking session from V$SESSION
  → USERNAME  MACHINE  OSUSER
Bad SQL
(more long term)

The Developers' Treaty (1983)

The undersigned agree that neither party will assess any blame on one another, make any negative comments about each other's code, nor make any remarks that are intended to or may be construed by a third party to be critical of said parties' code; furthermore …

Section 3-D; subsection 12; exhibit 34-C12
Bad SQL

- Inefficient logical reads
  - can be more expensive than physical reads
  - physical reads deemed "too expensive" before advances in storage technology
- Logical vs. physical
  - high buffer gets equates to the hourglass
  - minimize (almost at all costs)

The Plague of the Oracle Database

Buffer busy waits occur when a session wants to access a database block in the buffer cache but it cannot as the buffer is busy. The two main cases where this can occur are:

- another session is reading the block into the buffer
- another session holds the buffer in an incompatible mode to a new request
Latch free waits occur when a process waits for a latch that is currently busy (held by another process); the wait time increases exponentially.

- in willing-to-wait mode a process will spin and try to get the same latch again
- in no-wait mode a process will "give-up" and request a different latch if one is unavailable

Run 12 Times in 2 Minutes

select event, count(*)
from v$session_wait
where event in ('latch free',
               'buffer busy waits')
group by event;
Just Remember

• These 2 wait events are the nature of the beast with Oracle
• They appear at least 50% (if not more) of the time in $v$session_wait
• Persistent double-digits for one or both over an interactive running of the query may be cause for concern
<table>
<thead>
<tr>
<th>latch free</th>
<th>buffer busy waits</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

**Longer Term Triage**

- Where and what is the hot data
  - querying the data dictionary
  - turning on table monitoring (SYS.DBA_TAB_MODIFICATIONS)
- How's my buffer cache
  - bigger is not always better
  - should I have multiple block-sized buffer cache
Where?

```sql
select d.tablespace_name,
       sum(x.count) total_waits,
       sum(x.time)  time_waited
from sys.x$kcbfwait  x,
     sys.dba_data_files  d
where x.inst_id = userenv('Instance')
and x.count > 0
and x.indx + 1 = d.file_id
group by d.tablespace_name
order by 3 desc;
```

What's Happening?

```sql
select decode(state, 0,'Free',
             1,'Read and Modified',
             2,'Read and Not Modified',
             3,'Currently Being Read',
             'Other' ), count(*)
from sys.x$bh
group by decode(state,0,'Free',
                1,'Read and Modified',
                2,'Read and Not Modified',
                3,'Currently Being Read',
                'Other' )
```
Finding Culprits

- Buffer gets (STATSPACK)
- Consistent gets (AUTOTRACE)
- sqlplus
  - explain plan
  - set autotrace exp
- Today's small problems are tomorrow's disasters

Lessons to be Learned

- Your less-technical colleagues will be looking for a quick fix
- A "well-behaved" database all of a sudden performs poorly may be a contributor but not the culprit
- Queries may start behaving differently as data volumes change … hints and the cost-based optimizer

Michael S. Abbey — Triaging a Database Slowdown
Michael S. Abbey — Triaging a Database Slowdown
michael.abbey@ntirety.com

fenderpbs

*Michael S. Abbey — Triaging a Database Slowdown*