

## ❁ 52. RAC 환경에서 많이 발생하는 HW enqueue의 원인과 해결 방법

HW enqueue: High Water Mark enqueue의 약자

HWM를 올릴려면 이 enqueue를 확보 해야 HWM를 올릴 수 있다.

데이터를 insert 하려면 포맷된 디스크가 있어야 한다.

block -> extent -> segment(table, index) -> tablespace -> DB

=> 동시에 같은 테이블에 insert를 하는 세션이 다수 있으면 그중에 하나의 insert 문의 세션만 HW enqueue를 확보해서 HWM를 위로 올릴 수 있고 나머지 세션들은 모두 HW enqueue로 waiting 하게 된다.

=> 이러한 HW enqueue가 RAC 환경에서는 더 자주 발생한다. 왜냐하면 양쪽 노드에서 여러 세션들이 동시에 같은 테이블에 데이터를 입력하는 경우가 있기 때문에 하나의 인스턴스에서 보다 여러 노드일 때가 더 HW enqueue가 발생할 확률이 높다.

처방전

1. insert가 이루어지는 테이블이 속한 tablespace의 segment 관리 방법을 automatic segment space management 방식으로 설정해야 한다. (기본값이 이 방식으로 세팅 되어져 있다.)

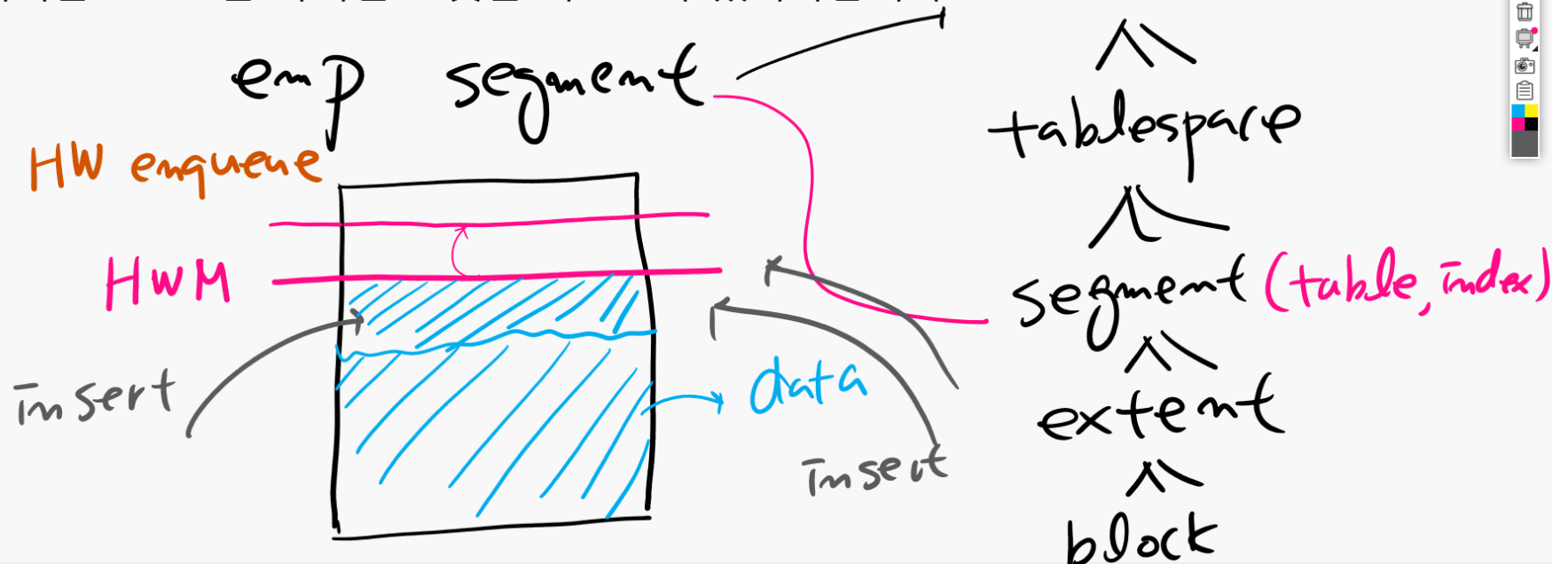
2. \_bump\_high\_water\_mark\_count 파라미터의 숫자값을 늘린다.

=> \_bump\_high\_water\_mark\_count: HWM를 올릴 때 한 번에 포맷할 수 있는 블록의 개수

- High Water Mark 가 무엇입니까 ? <--- 자주 물어보는 면접질문

답변: 포맷된 디스크와 포맷되지 않은 디스크의 경계선 입니다.

데이터를 insert 를 하려면 포맷된 디스크가 있어야합니다.



실습.

1. t\_log\_buffer\_space 테이블이 속한 테이블 스페이스의 segment 속성이 auto인지 manual인지 확인 (sql 1 owi)

```
select table_name, tablespace_name
from user_tables
where table_name='T_LOG_BUFFER_SPACE';
```

```
select tablespace_name, segment_space_management
from dba_tablespaces
where tablespace_name='OWI_TBS';
```

=> segment space management가 auto이면 오라클이 알아서 HW enqueue가 발생 안하겠금  
\_bump\_high\_water\_mark\_count를 자동 조절한다.

```
racdb1(OWI) > select table_name, tablespace_name
from user_tables
where table_name='T_LOG_BUFFER_SPACE';
```

TABLE_NAME	TABLESPACE_NAME
T_LOG_BUFFER_SPACE	OWI_TBS

```
racdb1(OWI) > select tablespace_name, segment_space_management
from dba_tablespaces
where tablespace_name='OWI_TBS';
```

TABLESPACE_NAME	SEGMENT_SPACE_MANAGEMENT
OWI_TBS	AUTO

2. 테이블 스페이스를 하나 만드는데 segment space 관리를 manual로 해서 생성하시오.  
create tablespace ts77 datafile '+data' size 10m  
segment space management manual;

3. t\_log\_buffer\_space 테이블을 ts77 테이블 스페이스로 이동시킨다.  
alter table t\_log\_buffer\_space move tablespace ts77;

```
select table_name, tablespace_name
from user_tables
where table_name='T_LOG_BUFFER_SPACE';
```

```
racdb1(OWI) > alter table t_log_buffer_space move tablespace ts77;
Table altered.

racdb1(OWI) > select table_name, tablespace_name
from user_tables
where table_name='T_LOG_BUFFER_SPACE';
```

TABLE_NAME	TABLESPACE_NAME
T_LOG_BUFFER_SPACE	TS77

4. (터미널 창 2번) HW enqueue가 발생하는지 모니터링을 준비한다.

@event

5. (터미널 창 1번) HW enqueue 관련 부하를 일으킨다.

@exec

log\_buffer\_space

```
racdb1(OWI) > @event

SID EVENT
-----
368 KJC: Wait for msg sends to complete
140 enq: HW - contention
143 enq: HW - contention
374 enq: HW - contention
376 enq: HW - contention
16 enq: TX - row lock contention
142 gc current multi block request
252 os thread startup
```

6. 사진을 한 번 찍는다.

@snap

7. (해결 방법) 다시 segment space management가 auto로 되어져 있는 owi\_tbs 테이블 스페이스로 t\_log\_buffer\_space 테이블을 move한다.

alter table t\_log\_buffer\_space move tablespace owi\_tbs;

문제 1. enqueue 중에 SQ enqueue 또는 HW enqueue를 가지고 테스트 해서 튜닝 전과 튜닝 후의 비교 레포트를 생성하시오.

```
select sequence_name, cache_size, order_flag
from user_sequences;
```

- SQ enqueue

튜닝 전

alter sequence seq\_sq\_enqueue cache 2;

@snap

@exec

sq\_enqueue

@snap

튜닝 후

alter sequence seq\_sq\_enqueue cache 50;

@snap

@exec

sq\_enqueue

@snap

@?/rdbms/admin/awrddrpt.sql

html

57

59

59

60

report\_sq\_enqueue.html

=> 만약 report에 sq에 관련된 항목이 나오지 않는다면 owi 유저를 drop하고 owi\_tbs 테이블 스페이스도 drop 한 다음, 양쪽 instance들을 내렸다가 올리고 다시 @install 한다.

```

48 28 Mar 2024 13:00 1
49 28 Mar 2024 14:00 1
50 28 Mar 2024 14:24 1
51 28 Mar 2024 14:25 1
52 28 Mar 2024 14:27 1
53 28 Mar 2024 15:00 1
54 28 Mar 2024 15:08 1
55 28 Mar 2024 16:00 1
56 28 Mar 2024 16:27 1
57 28 Mar 2024 16:34 1
58 28 Mar 2024 16:34 1
59 28 Mar 2024 16:36 1
60 28 Mar 2024 16:37 1

Specify the First Pair of Begin and End Snapshot Ids
~~~~~
Enter value for begin_snap: █

```

## Top Timed Events

- Events with a "-" did not make the Top list in this set of snapshots, but are displayed for comparison purposes

1st						2nd					
Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time	Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time
enq: SQ - contention	Configuration	42,372	97.84	2.31	15.82	enq: SQ - contention	Configuration	12,924	32.81	2.54	24.70
CPU time			20.22		3.27	CPU time			17.78		13.39
control file sequential read	System I/O	318	12.69	39.90	2.05	log file parallel write	System I/O	108	4.52	41.88	3.40
log file parallel write	System I/O	169	12.07	71.42	1.95	control file sequential read	System I/O	291	3.86	13.28	2.91
db file sequential read	User I/O	502	10.07	20.06	1.63	db file sequential read	User I/O	75	1.92	25.57	1.44
gcs log flush sync	Other	264	3.82	14.47	0.62	library cache: mutex X	Concurrency	470	1.45	3.09	1.09
row cache lock	Concurrency	32,526	2.42	0.07	0.39	gcs log flush sync	Other	64	0.92	14.37	0.69
row cache process	Other	212,537	2.37	0.01	0.38	gc cr block busy	Cluster	6	0.43	70.87	0.32
enq: WF - contention	Other	19	2.12	111.41	0.34	reliable message	Other	2	0.30	148.18	0.22
control file parallel write	System I/O	13	1.99	153.32	0.32	control file parallel write	System I/O	7	0.29	41.93	0.22
-library cache: mutex X	Concurrency	109	0.30	2.73	0.05	-row cache lock	Concurrency	1,839	0.24	0.13	0.18
-reliable message	Other	1	0.00	4.48	0.00	-row cache process	Other	34,327	0.10	0.00	0.07
-gc cr block busy	Cluster	1	0.00	1.21	0.00	-enq: WF - contention	Other	16	0.01	0.79	0.01

- HW enqueue

튜닝 전

```
alter table t_log_buffer_space move tablespace ts77;
```

@snap

@exec

```

-----
log_buffer_space
-----

```

@snap

튜닝 후

```
alter table t_log_buffer_space move tablespace owi_tbs;
```

@snap

@exec

```

-----
log_buffer_space
-----

```

@snap

@?/rdbms/admin/awrddrpt.sql

```

-----
html
69
70

```

71

72

report\_hw\_enqueue.html

-----

=> tablespace 공간이 부족하다면 추가해준다.

alter tablespace ts77 add datafile (사이즈를 따로 안 적으면 default 값이 100m)

```
65 28 Mar 2024 16:54 1
66 28 Mar 2024 16:54 1
67 28 Mar 2024 16:55 1
68 28 Mar 2024 16:56 1
69 28 Mar 2024 17:00 1
70 28 Mar 2024 17:00 1
71 28 Mar 2024 17:00 1
72 28 Mar 2024 17:01 1
```

Specify the First Pair of Begin and End Snapshot Ids

~~~~~

Enter value for begin\_snap: 69

First Begin Snapshot Id specified: 69

Enter value for end\_snap: 70

First End Snapshot Id specified: 70

## Top Timed Events

- Events with a "-" did not make the Top list in this set of snapshots, but are displayed for comparison purposes

| 1st                           |               |       |         |              |          | 2nd                          |               |       |         |              |          |
|-------------------------------|---------------|-------|---------|--------------|----------|------------------------------|---------------|-------|---------|--------------|----------|
| Event                         | Wait Class    | Waits | Time(s) | Avg Time(ms) | %DB time | Event                        | Wait Class    | Waits | Time(s) | Avg Time(ms) | %DB time |
| buffer busy waits             | Concurrency   | 6,730 | 277.21  | 41.19        | 47.19    | control file sequential read | System I/O    | 327   | 28.28   | 86.47        | 48.92    |
| control file sequential read  | System I/O    | 464   | 101.56  | 218.89       | 17.29    | gc current block busy        | Cluster       | 23    | 6.21    | 270.16       | 10.75    |
| cursor: pin S wait on X       | Concurrency   | 27    | 55.07   | 2,039.56     | 9.37     | control file parallel write  | System I/O    | 14    | 5.35    | 382.47       | 9.26     |
| enq: HW - contention          | Configuration | 6,438 | 54.26   | 8.43         | 9.24     | log file parallel write      | System I/O    | 77    | 5.21    | 67.60        | 9.01     |
| log file parallel write       | System I/O    | 120   | 38.43   | 320.26       | 6.54     | db file sequential read      | User I/O      | 62    | 3.43    | 55.35        | 5.94     |
| log buffer space              | Configuration | 36    | 35.66   | 990.43       | 6.07     | gcs log flush sync           | Other         | 158   | 1.89    | 11.96        | 3.27     |
| log file sequential read      | System I/O    | 56    | 33.41   | 596.53       | 5.69     | CPU time                     |               |       | 0.73    |              | 1.26     |
| enq: TX - row lock contention | Application   | 77    | 30.54   | 396.59       | 5.20     | local write wait             | User I/O      | 22    | 0.28    | 12.84        | 0.49     |
| log file sync                 | Commit        | 11    | 25.82   | 2,347.17     | 4.39     | log file sync                | Commit        | 2     | 0.26    | 128.66       | 0.45     |
| gcs log flush sync            | Other         | 1,231 | 25.63   | 20.82        | 4.36     | gc current block 2-way       | Cluster       | 162   | 0.13    | 0.82         | 0.23     |
| -db file sequential read      | User I/O      | 132   | 17.44   | 132.14       | 2.97     | enq: HW - contention         | Configuration | 2     | 0.00    | 0.90         | 0.00     |
| -control file parallel write  | System I/O    | 41    | 14.24   | 347.40       | 2.42     | -                            |               |       |         |              |          |
| -CPU time                     |               |       | 8.28    |              | 1.41     | -                            |               |       |         |              |          |
| -gc current block busy        | Cluster       | 1     | 2.23    | 2,234.18     | 0.38     | -                            |               |       |         |              |          |
| -gc current block 2-way       | Cluster       | 413   | 0.30    | 0.73         | 0.05     | -                            |               |       |         |              |          |
| -local write wait             | User I/O      | 1     | 0.00    | 2.02         | 0.00     | -                            |               |       |         |              |          |