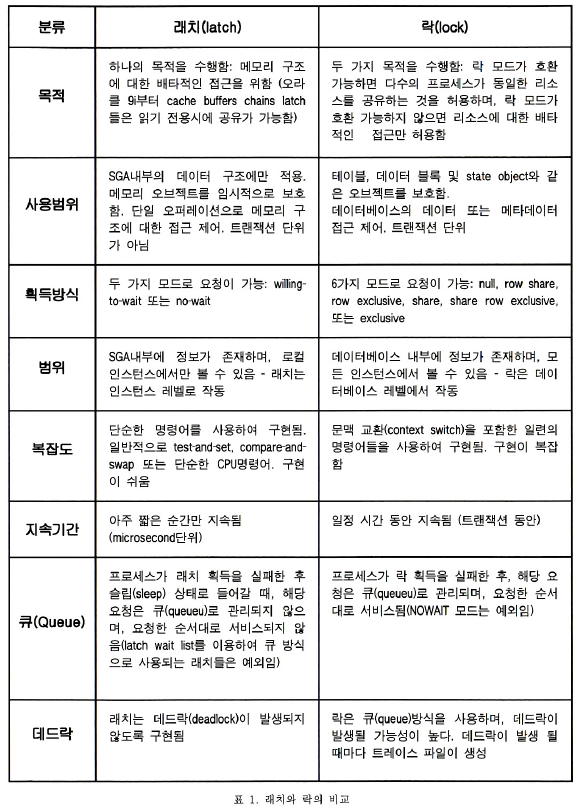
**래치와 락**

- 오라클의 동기화 매커니즘

오라클은 거대한 동기화 머신이다. 오라클은 래치와 락이라는 두 개의 동기화 매커니즘을 이용해 리소스를 보호한다. 래치와 락의 존재 이유는 동시 작업으로부터 오라클의 자원을 보호하는 것이다.



**래치 (Latch)**

- 래치란?

오라클 매뉴얼에서는 가벼운 락으로 분류하고 있다. 즉, 락 중에서도 아주 빠른 속도로 작동하게끔 구현된 경량화 된 락이라는 것이다. 그러나 보통은 래치와 락을 전혀 다른 객체로 분류하는 것이 일반적이다. 래치는 유닉스에서 제공하는 뮤텍스와 대단히 유사하다. 뮤텍스는 특정 리소스에 대한 배타적인 접근을 가능하게 하는 동기화 메커니즘이다. 래치는 뮤텍스와 같이 아주 빠르고 가볍게 특정 리소스에 대한 동기화를 구현하기 위해 고안된 객체이다. 특히 오라클은 래치 획득의 순서를 보장하지 않는다.

- 래치가 보호하는 리소스

SGA이다. SGA에 접근하는 모든 프로세스는 반드시 해당 영역을 관장하는 래치를 획득한 후에만 접근이 허용된다. 래치를 획득하는데 실패한 프로세스는 래치를 획득할 수 있을 때까지 특정 대기 이벤트를 기다리게 된다. 또한 락 자체도 SGA의 Shared Pool 영역에 존재하는 메모리 구조체이기 때문에, 래치를 통해서 보호를 받는다.

- 래치 동작 메커니즘

1) 래치 레벨: 오라클은 래치 획득 과정에서의 데드락을 원천적으로 방지하기 위해 모든 래치에 레벨을 부여한다. 레벨은 0~16까지 17가지의 값이 존재한다. 또한 자식 래치들은 항상 부모 래치와 동일한 레벨 값을 갖는다. 또한 한 개 이상의 래치를 보유한 상태에서 또 다른 래치를 획득하고자 하는 프로세스는 반드시 가장 최근에 획득한 래치보다 높은 레벨의 래치 만을 획득해야 한다. 만일 어떤 이유로 인해서 현재 보유한 래치보다 낮거나 같은 레벨의 래치를 획득하려면 No-Wait 모드로 획득하게 함으로써 데드락을 방지한다. 예를 들어, 0레벨의 래치가 필요하면 2레벨의 래치는 데드락을 피하기 위해 No-Wait모드로 획득을 시도하고, 실패하면 다시 래치를 해제한 후 다시 0,1의 순서로 획득을 시도한다.

2) 래치 모드: 래치는 기본적으로 Exclusive 모드를 사용한다. 따라서 한 순간에 하나의 프로세스만이 래치를 보유할 수 있다. 하지만 특정 래치에 대해서는 Shared 모드를 사용하기도 한다.

3) 래치 획득: 래치를 획득하고 자하는 프로세스는 기본적으로 Willing-to-wait 모드를 사용한다. Willing-to-wait모드란 래치를 획득할 수 있을 때까지 대기한다는 의미이다. 래치 경합이 발생하는 상황에서의 높은 CPU 사용율은 실제 작업을 하는 과정에서 발생하는 것이 아니라, 래치 획득을 위해 스핀을 수행하는 과정에서 유발된다. 스핀을 시도하고도 래치를 획득하는데 실패한 프로세스는 슬립 상태로 바뀐다. 슬립 상태란 프로세스가 cpu 사용을 포기한다는 의미이다. 또한 cpu가 하나밖에 없는 시스템에서는 스핀을 수행하지 않는다.

4) 래치 Cleanup: 공유 메모리 훼손을 막기 위한 메커니즘. 만일 프로세스가 래치를 보유한 채로 종료되면 PMON은 각 래치마다 구현되어 있는 Cleanup Function을 호출한다. 각 래치의 Cleanup Function은 래치 복구영역에 기록된 정보를 이용해서 공유메모리 영역을 원래 상태로 복구하고, 이 작업이 끝나면 PMON은 래치를 해지한다.

- 래치 관련 다이나믹 뷰

1) V$LATCH: 모든 래치들의 통계에 대한 합계치 정보를 제공

2) V$LATCH\_PARENT: 독립 래치에 대한 통계 값을 제공한다.

3) V$LATCH\_CHILDREN: 모든 개별 자식 래치들에 대한 통계 값을 제공한다.

GETS: Willing-to-wait모드에서 슬립하기 전의 래치 요청 회수

MISSES: Willing-to-wait모드에서 슬립하기 전의 래치 획득 실패 회수

SPIN\_GETS: Willing-to-wait모드에서 슬립하기 전의 스핀단계에서의 래치 획득 성공 회수

SLEEPS: Willing-to-wait모드에서 슬립 회수

IMEEDIATE\_GETS: No-Wait모드에서 래치 획득 성공 회수

IMMEDIATE\_MISSES: No-Wait모드에서 래치 획득 실패 회수

WAIT\_TIME: 래치를 획득하기 위해 대기한 시간 (Microsecond 단위)

4) V$LATCH\_MISSES: 래치 미스가 발생한 경우 오라클 커널 코드의 어떤 영역에서 발생했는지에 대한 정보를 가지고 있다.

- 일반적인 래치 관련 대기 이벤트들

1) latch: cache buffers chains - 버퍼 캐시에서 특정 부분을 탐색하고자 하는 프로세스는 cache buffers chains 래치를 획득해야 한다. 이 과정에서 경합이 나타나면 발생한다.

2) latch: cache buffers lru chain - 버퍼 캐시에서 프리 버퍼와 더티 버퍼를 탐색하고자 하는 프로세스가 많을 때 발생한다.

3) latch: shared pool - Shared Pool의 힙 영역에서 새로운 블록을 할당받고자 하는 경우 발생한다.

4) latch: library cache - Library Cache 영역을 탐색하고자 하는 프로세스가 많을 때 발생한다.

5) latch: redo copy - DML에 의한 변동 사항을 리두 버퍼에 기록하고자 하는 프로세스가 많을 때 발생한다.

**락 (Lock)**

- 락의 분류

1) Enqueue Lock: User Type Lock (TX, TM, UL), System Type Lock (CP, US, CI, TC, JS, ...)

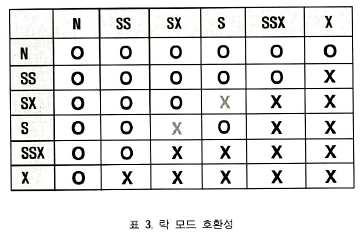
2) 일반 락: row cache lock, library cache lock, library cache pin, buffer lock

- 락이 보호하는 리소스

락은 데이터베이스 전체를 보호한다. 테이블, 트랜잭션, 언두 세그먼트, 테이블스페이스, 잡 등 데이터베이스 단위에서 필요한 대부분의 리소스들은 락의 보호를 필요로 한다. Enqueue 락 경합이 발생한 경우 V$LOCK 뷰나 V$SESSION\_WAIT 뷰의 P1, P2, P3 에서도 같은 정보를 얻을 수 있으나, 일반 락의 경우는 별도의 뷰가 없이 V$SESSION\_WAIT를 통해 어떤 대기 현상이 발생했는지 확인해서 발생 여부 및 경합 여부를 알 수 있다.

- 락 동작 메커니즘

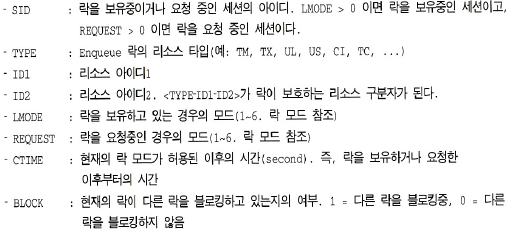
1) 락 모드: 오라클은 락을 획득하는 모드를 다양하게 구성한다. (0: None, 1: Null, 2: SS/RS, 3: SX/RX, 4: Shared, 5: SSX/SRX, 6: Exclusive(X))



2) 락 획득: Enqueue락 획득에 실패한 프로세스는 자신을 Enqueue 리소스의 대기 목록에 등록한다. 현재 락을 보유한 프로세스는 락의 사용이 끝나면 락을 해제하고, Enqueue 리소스의 대기 목록에서 다음 프로세스를 깨워준다. 락 획득에 실패한 프로세스는 대기목록에 대기상태에 들어갈 때 특정 타임아웃 값을 지니는 알람을 설정한다. 또한 락은 데드락의 가능성이 항상 존재한다. 따라서 락을 대기하는 프로세스는 정해진 시간이 되면, 즉 타임아웃이 발생하면 깨어나서 데드락이 발생했는지 여부를 체크한다.

- 락 관련 다이나믹 뷰

1) V$LOCK이 Enqueue 락 관련한 최고의 뷰이다. 중요한 칼럼들은 아래와 같다.



2) V$LOCKED\_OBJECT: 현재 시스템의 모든 트랜잭션에 의해 획득 중인 TM락에 대한 정보를 제공한다. 락이 걸린 테이블을 조회할 때 유용하다

3) V$ENQUEUE\_LOCK: V$LOCK뷰와 거의 동일하지만, 실제 Enqueue 현상, 즉 블로킹이 발생한 경우 대기 세션에 대한 정보만 조회된다. 즉, 어떤 세션이 락을 대기하기 위해 큐에서 대기하고 있는지 정보만 나온다.

4) Enqueue 락이 아닌 경우 V$LOCK 뷰와 같은 통합적인 뷰는 존재하지 않는다.

5) DBA\_DDL\_LOCK, DBA\_KGLLOCK, X$KGLLK: 라이브러리 캐시 락 관련 뷰

6) DBA\_KGLLOCK, X$KGLPN: 라이브러리 캐시 핀 관련 뷰

7) V$ROWCACHE\_PARENT: row cache lock 관련 뷰

-일반적인 락 관련 대기 이벤트들

1) row cache lock: row cache lock에 의한 경합

2) buffer busy waits, read by other session: buffer lock에 의한 경합

3) library cache lock: library cache lock에 의한 경합

4) library cache pin: library cache pin에 의한 경합

5) DFS lock handle: 글로벌 락인 경우(RAC)인 경우에 관찰된다.