**OWI 란?**

- 오라클은 어떤 작업을 수행하는 과정에서 원하는 리소스를 획득하지 못하는 경우에 리소스에 대한 점유가 해제될 때까지 리소스와 관련된 이벤트를 대기한다. 이처럼 프로세스가 겪는 대기 현상을 기록하고 관찰하는 일련의 기능과 인터페이스, 그리고 방법론을 통칭하여 OWI, 즉 Oracle Wait Interface라고 부른다.

대기 이벤트는 P1, P2, P3 라는 세 개의 파라미터를 통해 현재 대기하는 리소스를 표현한다. 대기 이벤트의 P1, P2, P3의 값은 V$SESSION\_WAIT 뷰나 V$SESSION 뷰와 같은 다이나믹 뷰나 SQL Trace 파일 등을 통해 관찰 가능하다. 각 이벤트마다 P1, P2, P3의 의미는 각각 다르며, V$EVENT\_NAME 뷰를 통해 확인 가능하다.

오라클은 대기횟수, 타임아웃횟수, 대기시간이라는 세 가지 값을 이용해 대기 현상을 표현한다. 특히, 래치와 같은 자원의 경우, 래치 획득 시도 횟수가 매우 빈번하기 때문에, 대기횟수로 경합여부를 판단하는 것은 의미가 없으며, 대기시간을 통해 경합 정도를 가장 정확하게 분석할 수 있다.

Response Time = Service Time + Wait Time 이 된다. Service Time과 Wait Time을 낮추면 자연스럽게 응답시간이 낮아지고, 사용자의 성능 만족도는 증가한다.

**OWI의 특징**

- OWI는 문제 지향적이다. 오라클은 문제가 생길 때마다, 즉 원하는 자원을 획득하는데 실패할 때마다 특정 이벤트를 대기함으로써 현재 어떤 문제가 발생했고, 왜 발생했는지를 보고한다. AWR의 성능 이력 관리 기능을 이용하면, 과거 특정 시점의 문제까지도 OWI에 기반해서 분석할 수 있다.

- OWI는 정량적이다. OWI에서의 사실은 특정 이벤트의 대기횟수, 타임아웃횟수, 대기시간으로 표현된다. OWI는 문제가 왜 발생했는지 뿐만 아니라, 구체적으로 어떤 정도의 성능 문제를 유발하고 있는지를 표현해준다.

- OWI는 징후학적이다. 즉, 대기 현상은 오라클이 내부의 성능문제를 외부로 드러내는 징후이다.

- OWI는 개선 중이다. 오라클은 대기 이벤트를 더욱 세분화시키고, 더욱 많은 다이나믹 뷰들을 추가하고 있다. 이러한 세분화는 성능 문제를 더욱 정확하게 묘사하고, 직관적으로 분석할 수 있도록 도와준다. 또한 대기현상을 보고하는 다이나믹 뷰들은 더욱 더 많은 종류의 정보를 담고, 더 사용하기 편한 방식으로 진화하고 있다.

**OWI 툴**

- OWI 다이나믹 뷰

1) V$EVENT\_NAME: 오라클에 정의된 모든 대기 이벤트에 대한 참조성 정보를 제공한다.

2) V$SYSTEM\_EVENT: 인스턴스 가동 후 지금까지 모든 세션에서 발생한 대기 이벤트의 누적 정보를 담고 있다.

3) V$SESSION\_EVENT: 현재 접속되어 있는 각 개별 세션 별 대기이벤트의 누적 정보를 담고 있다.

4) V$SESSION\_WAIT: 각 세션이 현재 대기하고 있는 이벤트에 대한 상세정보를 제공한다. 누적 정보가 아닌 실시간 정보이므로, 반복적으로 조회를 해야 의미 있는 정보를 얻을 수 있다.

5) V$SESSION\_WAIT\_HISTORY: V$SESSION\_WAIT 정보를 세션별로 최대 10개까지 누적해서 제공한다.

6) V$SYSTEM\_WAIT\_CLASS: V$SYSTEM\_EVENT와 동일한 차원의 값을 클래스 기준으로 제공한다.

7) V$SESSION\_WAIT\_CLASS: V$SESSION\_EVENT와 동일한 차원의 값을 클래스를 기준으로 제공한다.

8) V$EVENT\_HISTOGRAM: 각 대기 이벤트에 대해서 대기 시간의 구간 (버킷)별로 대기 횟수를 보여준다.

- 기타 중요 다이나믹 뷰

1) V$SESSION: 세션 정보

2) V$ACTIVE\_SESSION\_HISTORY: 세션 히스토리 정보

3) V$PROCESS: 프로세스 정보

4) V$TRANSACTION: 트랜잭션 정보

5) V$LATCH, V$LATCH\_PARENT, V$LATCH\_CHILDREN, V$LATCH\_HOLDER: 래치 정보

6) V$SQL: SQL 정보

7) V$LIBRARYCACHE, X$KGLLK, X$KGLPN: 라이브러리 캐시 정보

8) V$ROWCACHE, V$ROWCACHE\_PARENT: 로우 캐시 정보

9) V$SGASTAT: SGA 정보

10) V$SEGMENT\_STATISTICS: 세그먼트 레벨 통계 정보

11) V$SESS\_TIME\_MODEL, V$SYS\_TIME\_MODEL: Time Model 정보

12) V$BH, X$BH: 버퍼 캐시 정보

- Extended SQL Trace

SQL Trace는 기본적으로 10046 진단 이벤트를 통해 활성화되며, LEVEL 1,4,8,12 의 4가지 레벨로 제어할 수 있다. SQL의 성능 진단을 위해서는 레벨 12를 사용하며, 이 때 SQL 문장의 실행정보와 바인드 변수 값, 대기 정보를 제공한다.

- oradebug와 덤프

oradebug 툴과 덤프 파일들은 비록 OWI와 직접적인 관계는 없지만 대기 현상들을 분석하고 이해하는데 매우 유용한 툴들이다. Oradebug는 DBA를 위한 가장 강력한 툴이다. 오라클을 디버깅을 위한 것으로 사용되지만, 적절히 사용하면 오라클을 관리하거나 내부 매커니즘을 테스트하는데 큰 도움이 된다. 덤프 수행은 SGA, PGA, 라이브러리 캐시, row cache, buffers, enqueue, latch, heap, hang analyze 등에서 가능하다. 이러한 덤프 파일들은 오라클의 내부 구조를 이해하는데 큰 도움을 주며, 기술지원팀에서 장애를 진단하는데 널리 사용된다.

**AWR (Automatic Workload Repository)**

- 오라클 성능 관리에서 가장 미약했던 부분인 성능이력관리 부분을 크게 강화시킨 것이다. AWR을 사용하면 오라클 성능 관리에 필요한 대부분의 데이터를 수집할 수 있으며, 수집된 데이터는 DBA\_HIST\_XXX뷰를 통해 조회 가능하다. AWR에서 스냅샷이 저장되는 기본 주기가 30분인 점을 감안하면 AWR로 특정 시간대의 장애 문제를 해석하는 것은 다소 어려워 보인다.

**OWI 데이터 수집**

- Logoff Trigger

- SQL을 이용한 주기적인 수집

- AWR

- DMA (Direct Memory Access): 원하는 데이터를 원하는 주기로 저장할 수 있음.