**I/O와 OWI**

- I/O 개요

I/O 작업 레이어는 다음과 같이 나눌 수 있다.

1) 어플리케이션 레이어: select / insert / update / delete / truncate

2) 오라클 메모리 레이어: Buffer cache | PGA

3) 오라클 세그먼트 레이어: Datafile, tempfile, Tablespace, Segment

4) OS/디바이스 레이어: Async I/O, Direct I/O, Raw Device, RAID, ….

I/O 성능 문제의 원인을 파악하는 것과 해결책을 찾는 과정은 항상 1,2,3,4의 순서를 따라야한다.

- 어플리케이션 레이어

어플리케이션을 효과적으로 구현하여, 불필요한 I/O를 최소화해야 한다. 비효율적인 어플리케이션을 두고 시스템을 튜닝한다는 것은 거의 불가능하다. 오라클은 병렬 쿼리, 병렬DML, Nologging, Direct load, Direct read 등 I/O를 효율적으로 처리하는 다양한 기법들을 제공한다.

- 오라클 메모리 레이어

오라클이 버퍼 캐시를 효과적으로 사용하기 위해 제공하는 기능들은 다음과 같다.

1) Touch Count 기반의 효율적은 LRU 알고리즘을 제공한다.

2) Buffer Pinning

3) 다중 버퍼 풀 기능

4) 다양한 블록 사이즈 제공

5) Direct Path I/O

- 오라클 세그먼트 레이어

오라클 8i부터 제공되는 LMT와 9i부터 제공되는 ASSM을 사용하면 익스텐트 및 세그먼트 공간의 부적절한 관리에서 오는 성능 문제를 대부분 해결할 수 있다. 또한 대용량 테이블은 파티션을 이용해서 관리하는 것이 유리한 경우가 많다.

- OS/디바이스 레이어

오라클은 가능하면 비동기 I/O를 사용할 것을 권장한다. 비동기 I/O가 안된다면 Direct I/O를 사용하는 것이 바람직하다. 이것보다 유의해야 할 점은 비효율적으로 과다하게 I/O를 수행하는 어플리케이션은 위 두개가 무조건적인 해결책은 아니라는 것이다. 어떠한 경우 어플리케이션의 적절한 튜닝이 정답일 수도 있다.

- Direct path I/O

오라클의 I/O는 기본적으로 SGA를 경유한다. SGA를 경유하지 않고, PGA에 바로 데이터를 올리는 I/O작업이 있는데 이것을 Direct path I/O라고 한다. 다음과 같은 경우 direct path I/O를 사용한다.

1) 정렬작업을 위해 정렬 세그먼트를 읽고 쓰는 경우

2) 병렬 쿼리를 위해 데이터 파일을 읽는 경우

3) PDML이나 CTAS를 위해 데이터파일을 쓰는 경우

4) NOCACHE 속성으로 생성된 LOB 세그먼트를 읽고 쓰는 경우

5) I/O 시스템이 데이터를 읽어서 오라클에 반환하는 속도보다 훨씬 빠른 속도로 버퍼를 요구할 때(readahead I/O)

Direct Path I/O는 버퍼 캐시를 경유하지 않기 때문에 동기화에 따른 오버헤드가 없다. 따라서 경합으로 인한 성능저하 현상이 발생하지 않는다. 정렬 작업과 같은 경우를 제외하고는 튜닝 작업을 통해 direct path I/O에서의 대기 횟수 및 대기 시간을 줄이는 것은 불가능하다.