


[이전 포스팅](#)에서 HWM (High Water Mark) 에 대해 따로 다루기로 했었는데 그만큼 HWM 가 데이터 저장 및 스캔과 관련해서 중요한 역할을 하기 때문이다.


Segment 와 Extent

우선 HWM 는 Segment의 구성요소이자 Extent 와도 중요한 관계가 있으므로 다시 짚고 넘어가자.

[Segment]

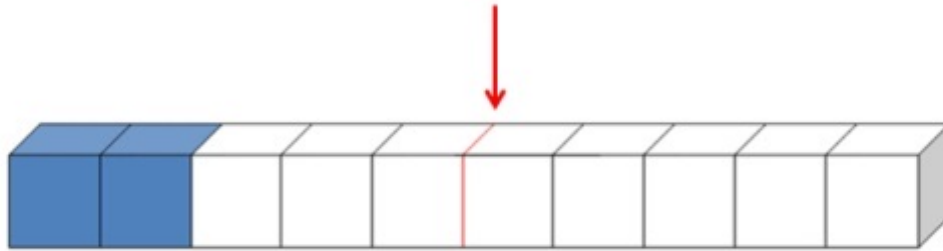
1. 테이블스페이스 내에 특정 유형의 논리적 저장구조로 할당된 영역
2. 테이블, 인덱스 등의 오브젝트가 세그먼트에 포함된다.
-  3. 저장 공간을 가지는 오브젝트만을 세그먼트라고 함 (≠ View, Sequence, Synonym)

[Extent]

1. 하나 이상의 연속된 데이터 블록의 모임
2. 세그먼트에 공간을 할당하는 단위
-  3. 하나의 데이터 파일에만 존재

자세한 것은 위 포스팅을 보거나, 간단하게 Segment는 테이블 같은 오브젝트, Extent 는 이런 Segment에 공간을 할당하는 단위 정도로 생각해도 좋다.

HWM (High Water Mark)



HWM

이미지에서 전체 블록은 Segment에 할당된 영역, 각각의 블록은 Data Block에 해당하고, 5개의 Block 묶음은 하나의 Extent에 해당한다. 그리고 빨간 색으로 표시된 부분이 바로 HWM 이다. 그럼 HWM이 무슨 역할을 하는 걸까?

[HWM]

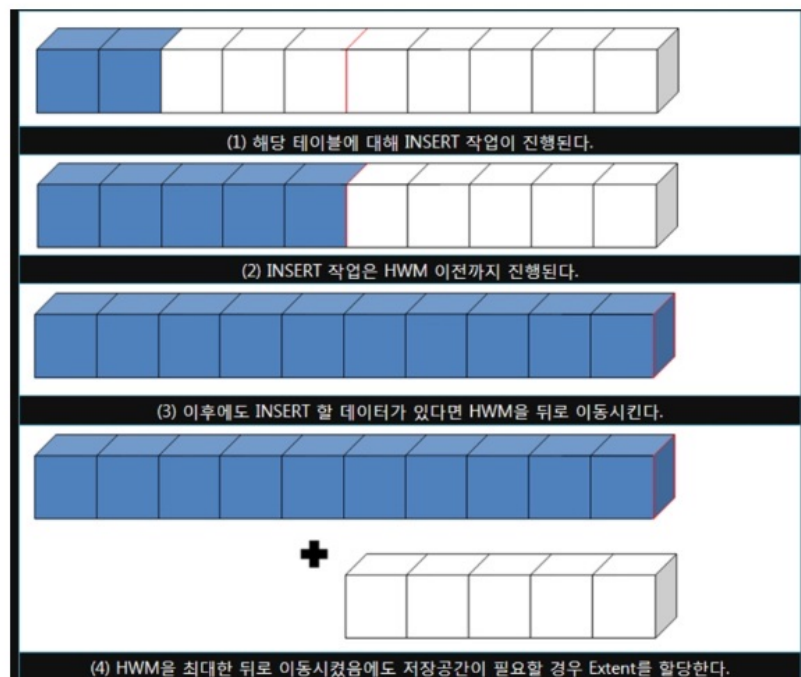
1. Extent 확장의 기준, 모든 세그먼트에 하나씩 존재
2. 1번에 5개의 데이터 블록 단위로 HWM 이동
3. HWM 이전 블록에만 저장 가능
4. Full Scan 수행 시, HWM 앞의 모든 데이터 블록 액세스
5. Data가 적는데 풀 스캔 시간이 오래 걸리면 세그먼트 축소 필요

위에서 언급한 것을 살펴보자.

데이터는 HWM 이전 블록까지만 저장 가능하
고 한 번에 5개의 데이터 블록, 즉 Extent 단위
로 HWM을 이동시킨다고 했다. 이 말은 HWM
이 있는 위치까지 우선 데이터를 저장 후 저장공
간이 꽉 차게 되면, 무작정 추가공간을 할당해서
저장하는 것이 아니라 우선 HWM 을 이동시키
고 해당 공간에 데이터를 저장하게 된다.

↑ 핵심!!

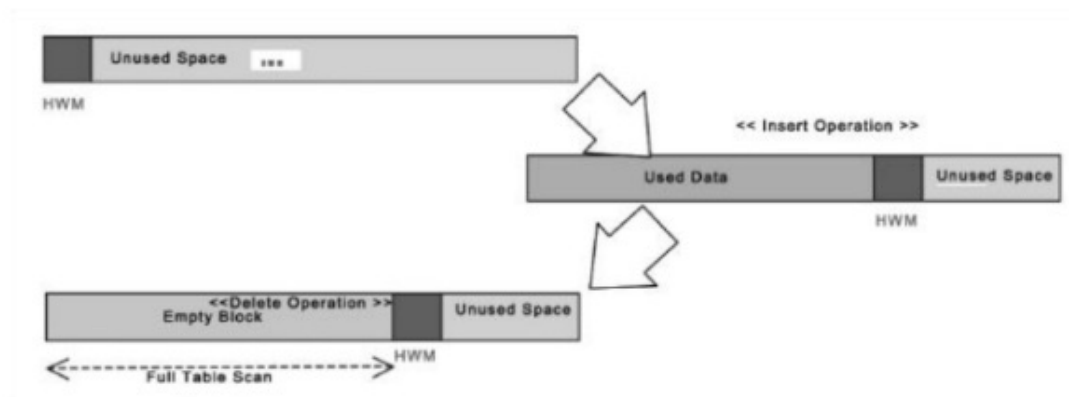
아래 그림을 보자.



HWM는 관리자가 별도로 초기화하거나 축소시키지 않으면 늘어나기만 하고 줄어들지 않는다
는 것을 알아두어야 하는데, 이 점을 꼭 알아두어야 하는 이유는 데이터 풀스캔 시 데이터 스캔의 범위 기준이 바로 HWM 이기 때문이다.

DB 사용 목적에 따라 차이가 있겠으나 당연히 데이터는 추가와 삭제가 발생하며, HWM를 이동시켜야 할 정도로 많아지기도 했다가 많은 공간이 비게 될 정도로 데이터를 삭제시키는 경우도 있을 것이다. 그런데 데이터베이스는 이런 데이터의 많고 적음에 상관없이 HWM 까지의 데이터블록 전체를 스캔하게 된다. 심지어 데이터가 0건인 경우라도 HWM 가 1억 Row의 데이터가 있던 때를 기준으로 설정되어 있다면 그만큼의 탐색 시간이 소요된 뒤 0건의 조회 결과를 출력하게 되는 것이다.

아래 이미지에서 Empty Block 일 때도 HWM
까지 Full Scan 이 발생하고 있음을 잘 보여주
고 있다.



이처럼 사용하지 않는 공간이 많으면 공간낭비
뿐만 아니라 조회 성능이 떨어지는 문제도 발생
하는 것이다. 그래서 이런 비효율적인 부분들
을 제거하기 위해서 주기적으로 통계정보 등을
바탕으로 재구성해주어야 할 테이블이나 인덱스
를 확인해주는 것이 필요하다.

Delete, Truncate 차이점

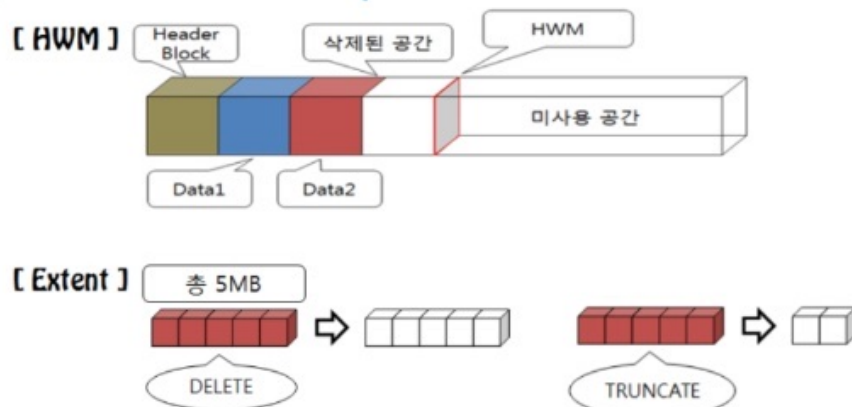
이런 Segment의 축소에 대한 것도 많은 내용을 필요로 하므로 다음 포스팅에서 다루기로 하고, HWM 를 조절할 수 있는 방법 중 하나인 Truncate와 Delete 의 차이점에 대해서만 간단히 확인해보고 가자. 특정 테이블의 전체 데이터를 삭제하기 위해서 다음 두 가지 명령어를 사용할 수 있다.

- 1. delete from [table_name] + commit
- 2. truncate table [table_name]

명령을 수행하면 테이블의 데이터가 모두 삭제
가 되어 있을 것이므로 HWM 에 대해 모른다면
그냥 같은 의미로 받아들일 수도 있을 것이다.

하지만 내부적으로 보면 이 둘은 큰 차이가 있
다. (HWM 외에도 다른 차이점들이 있지만 여기
서는 일단 넘어가자.)

Delete, Truncate 차이



[Truncate]

1. 자동 Commit
2. HWM를 초기화 시킴
3. MINEXTENTS 설정값만큼 제외하고 남은 EXTENT 는 모두 할당 해제
4. 전체 데이터 삭제 시 Delete보다 Truncate를 사용하는 것이 성능 면에서 유리

HWM 에 관해서만 보더라도 delete는 기존에 할당된 영역 및 HWM 의 위치가 그대로인 반면, truncate 는 HWM 의 위치를 초기화시키고, MINEXTENT 설정값만큼의 공간만 남긴 뒤 모두 할당 해제시킨다.

따라서 위에서 우리가 이미 알아본 바와 같이, 이후 동일 테이블에 똑같은 SQL문으로 조회를 시도했을 때 delete 로 삭제했을 경우와 truncate 로 삭제했을 경우의 성능이 다르게 나올 것이다.