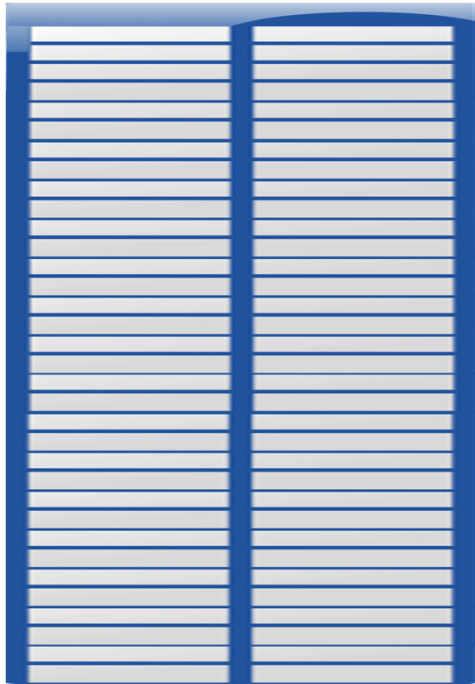


효율적인 데이터 저장을 위한 오라클 솔루션

Partition, ACO, ADO

시간이 흐름에 따라 Table 사이즈 증가



- Table은 점점 사이즈가 증가
- 적은 수의 Table이 대부분의 Data를 차지
 - 80:20 rule very common
- 테이블의 크기는 성능에 영향
 - Indexing이 만능은 아님
 - Data maintenance는 서비스 영향도가 큼
- DB 논리모델을 해치지 않는 해결 방법은?

The Concept of Partitioning

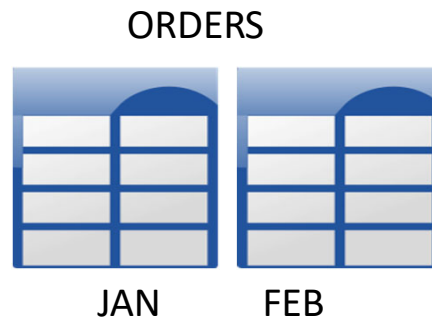
Simple Yet Powerful

파티션이란 테이블에 있는 특정 컬럼값을 기준으로 데이터를 분할해 저장해 놓은 테이블



Large Table

관리하기 어려움

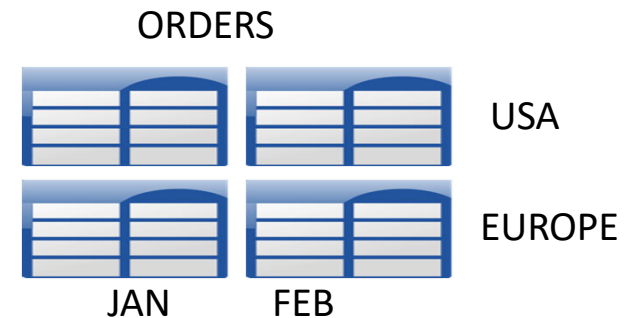


Partition

Divide and Conquer

관리 용이

성능 향상



Composite Partition

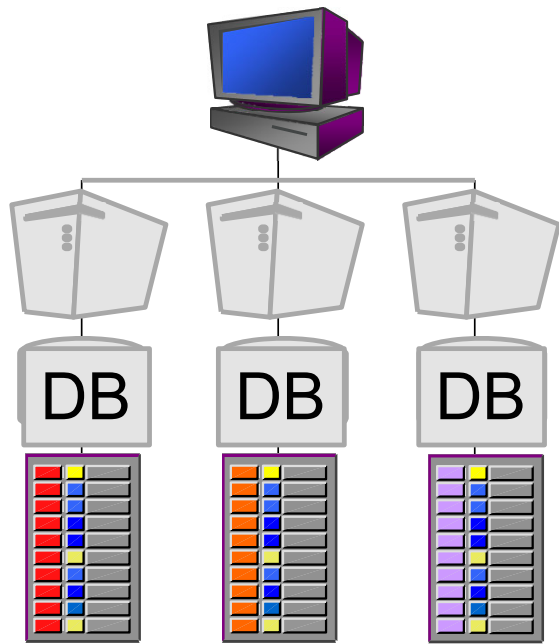
더 나은 성능

비즈니스 요구에 부합하는
더 나은 유연성

Transparent to applications

Physical vs Logical Partitioning

Shared Nothing Architecture

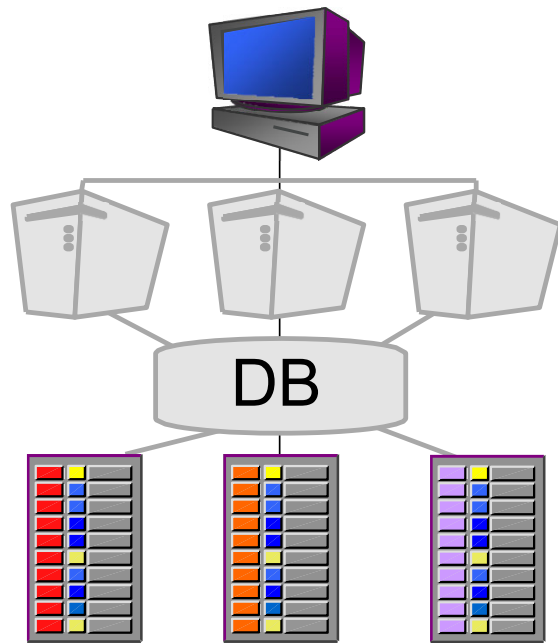


Physical Partitioning

- 기본적으로 이를 위한 시스템이 필요
- 각 노드는 DB의 부분
- Enables parallelism
- partition의 수는 최소 병렬도
- HASH distribution 필수
- 적절한 로드밸런싱을 위해 동일한 크기의 partition필요

Physical vs Logical Partitioning

Shared Everything Architecture



Logical Partitioning

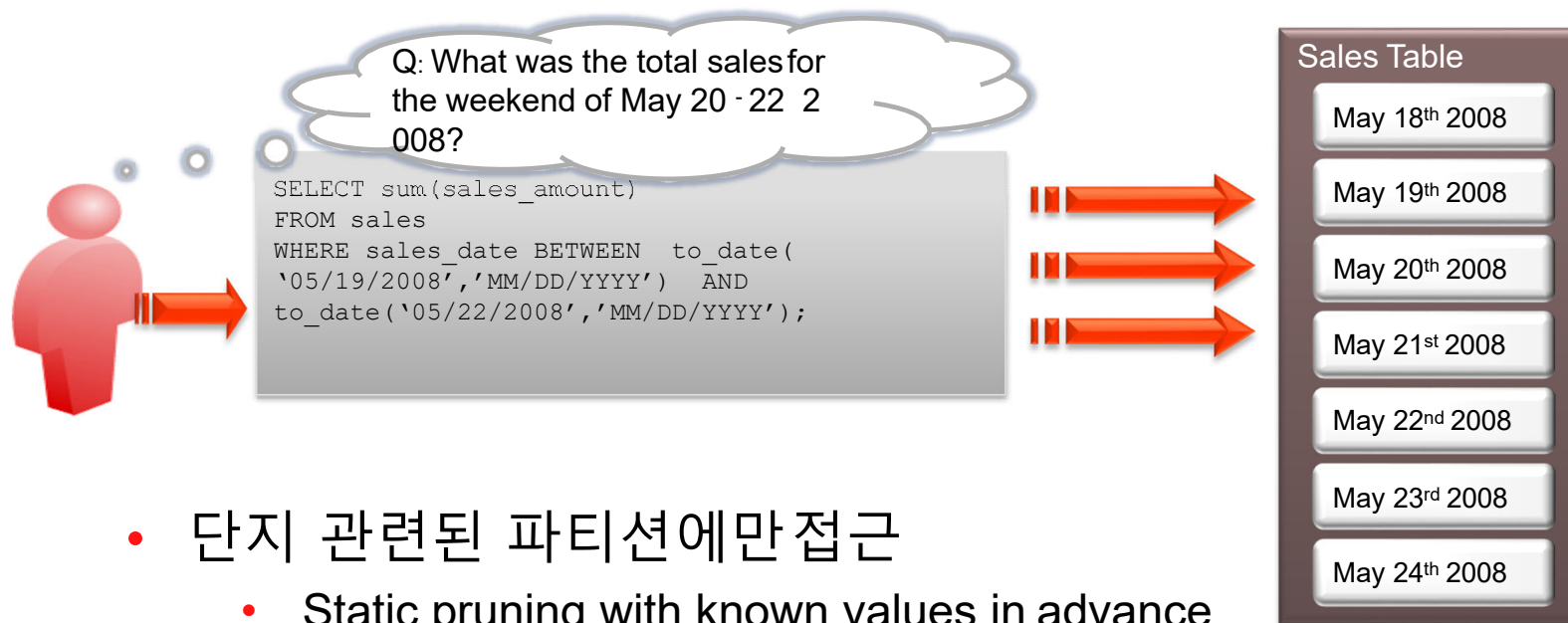
- 구성에 제약이 없음
 - SMP, MPP, Cluster, Grid 와 관계 없음
- 순수 business 요구사항에 근거
 - Availability, Manageability, Performance
- 모든 환경에서 이점 제공
 - 가장 포괄적인 기능

Partitioning의 목적

- 3가지 이점

- **Faster**: 관련된 데이터에서 조회/변경 등 작업
- **Flexible**: 필요한 데이터만 관리
- **Cheaper**: 적절한 저장장치에 구분하여 저장

Partitioning의 이점-Faster Partition Pruning

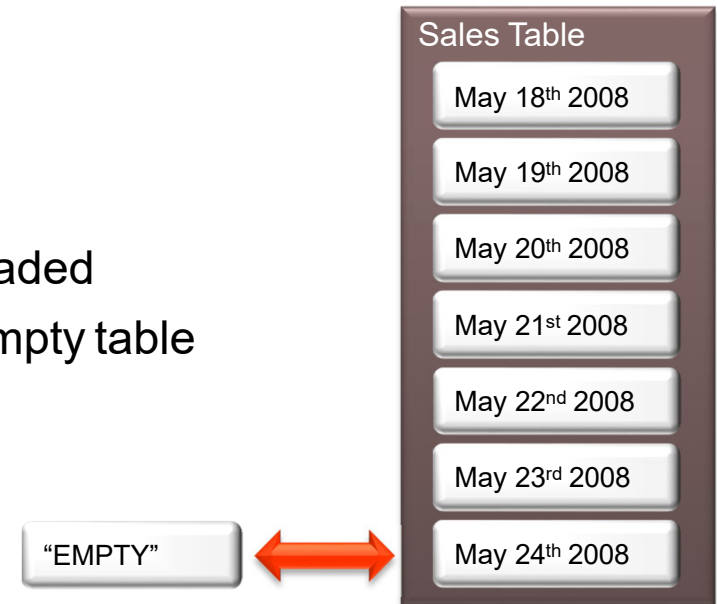


- 단지 관련된 파티션에만 접근
 - Static pruning with known values in advance
 - Dynamic pruning at runtime
- I/O operations 최소화
 - 큰 성능 향상

Partitioning의 이점-Flexible

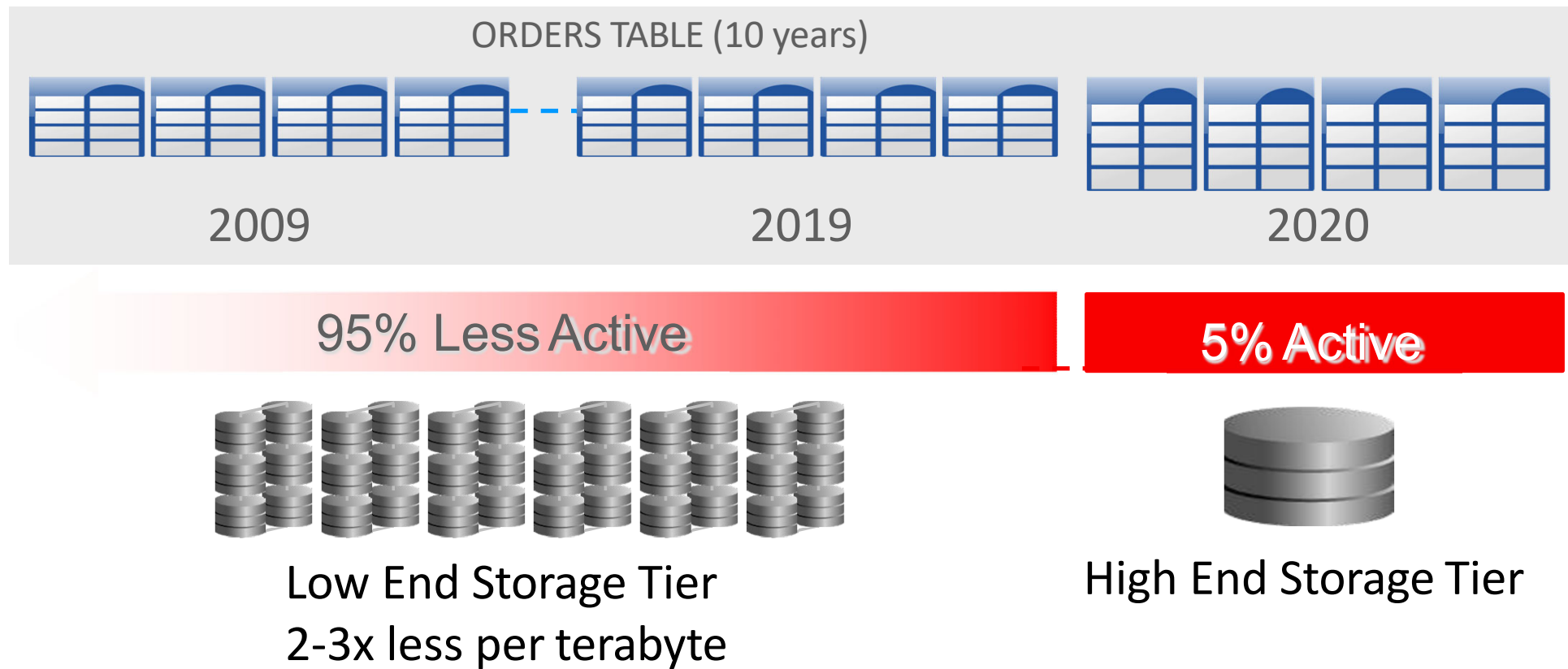
Partition Purging and Loading

- 메타데이터 작업으로 데이터를 제거/추가
 - Exchange the metadata of partitions
- Standalone table과 single partition exchange
 - Data load: standalone table contains new data to being loaded
 - Data purge: partition containing data is exchanged with empty table
- Purge대신 partition drop 가능
 - Data is gone forever



Partitioning의 이점-Cheaper

Partition for Storage and Compression Tiering

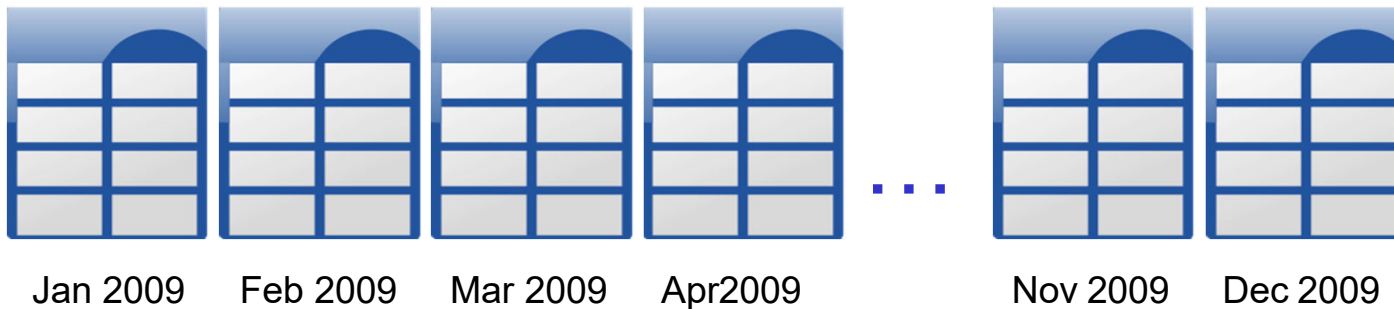


Partition의 종류

Range Partition

- 범위로 Data분배

- Lower boundary derived by upper boundary of preceding partition
- No gaps



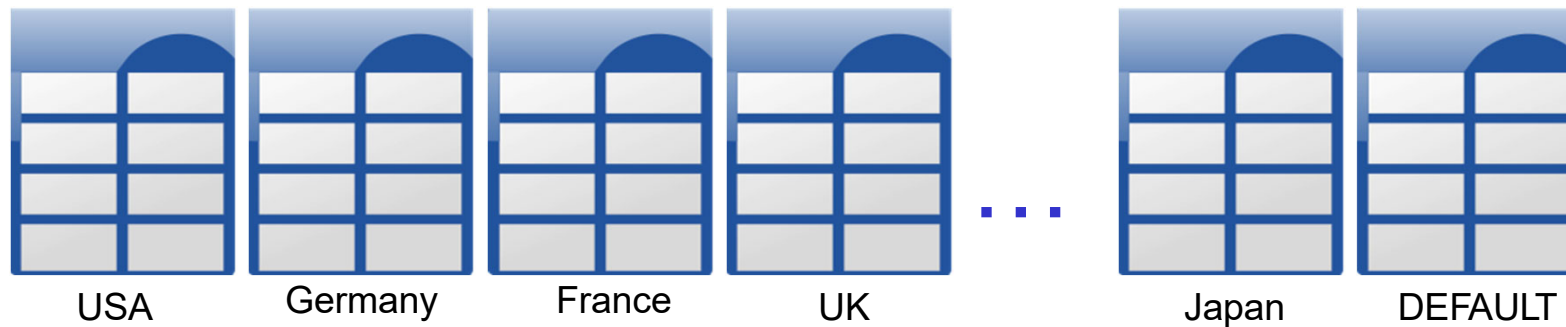
- 시간순 데이터에 이상적

Partition의 종류

List Partition

- 특정 값으로 Data분배

- One or more unordered distinct values per list
- Functionality of DEFAULT partition (Catch-it-all for all unspecified values)



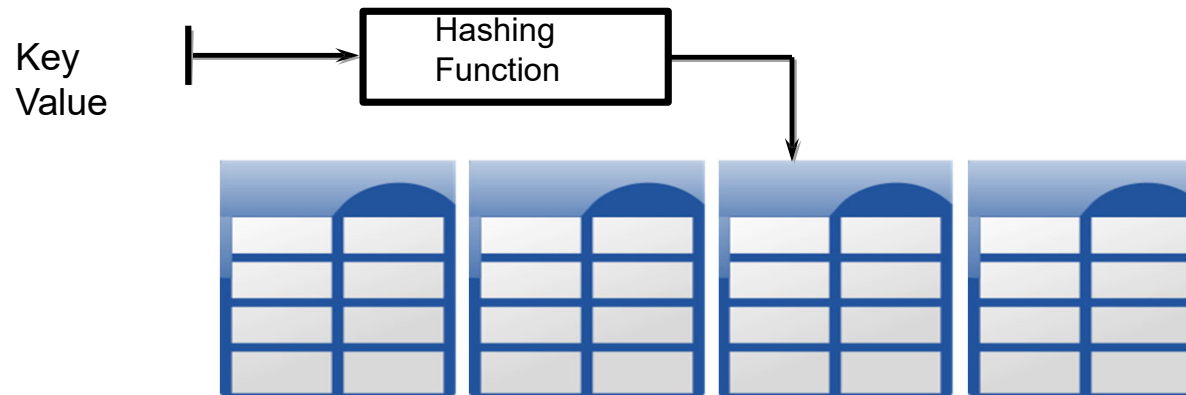
- 구분되는 고유한 값들에 이상적, 예) region

Partition의 종류

Hash Partition

- Hash값으로 Data분배

- Number of hash buckets equals number of partitions

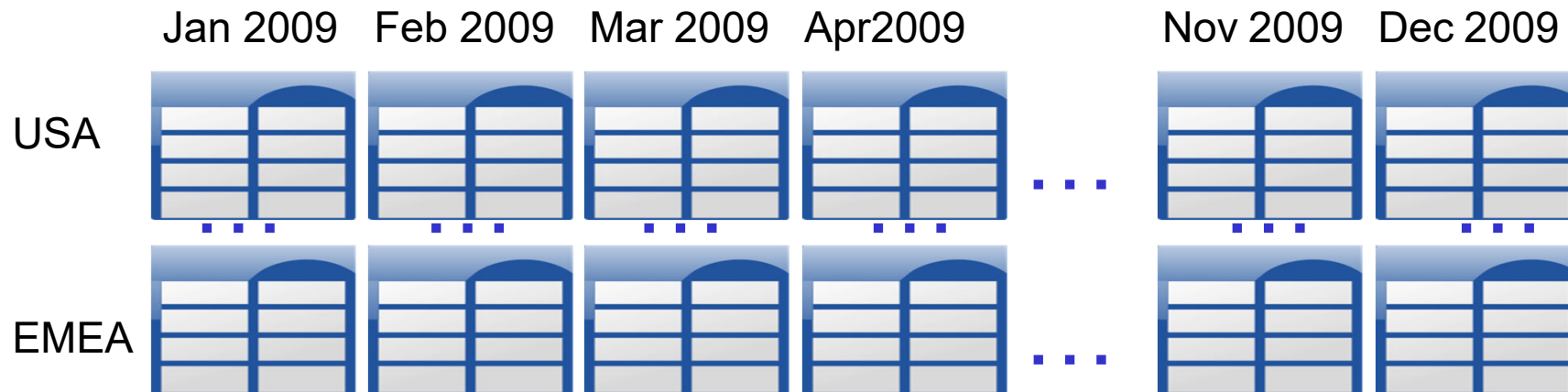


- 데이터 균등 분배에 이상적

Partition의 종류

Composite Partition

- 2차원으로 Data 분배
 - Record placement is deterministically identified by dimensions
 - Example RANGE-LIST



Program Agenda

- 1 들어가며
- 2 ILM(Information Lifecycle Management)
- 3 Partitioning
- 4 Compression**
- 5 ADO(Automatic Data Optimization)

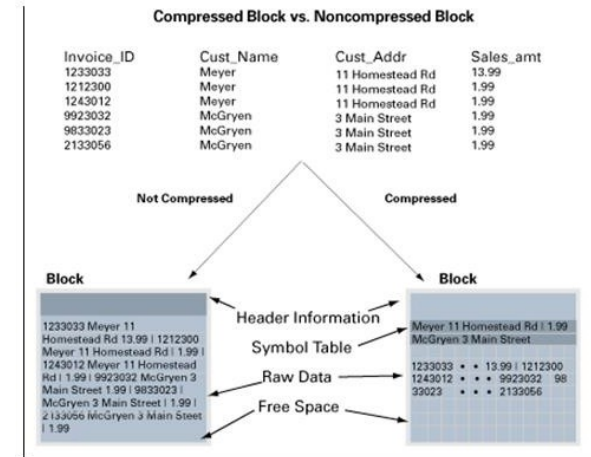
Advanced Row Compression

- “**ACTIVE**” data에 적합하며, cold or inactive data에도 사용 가능

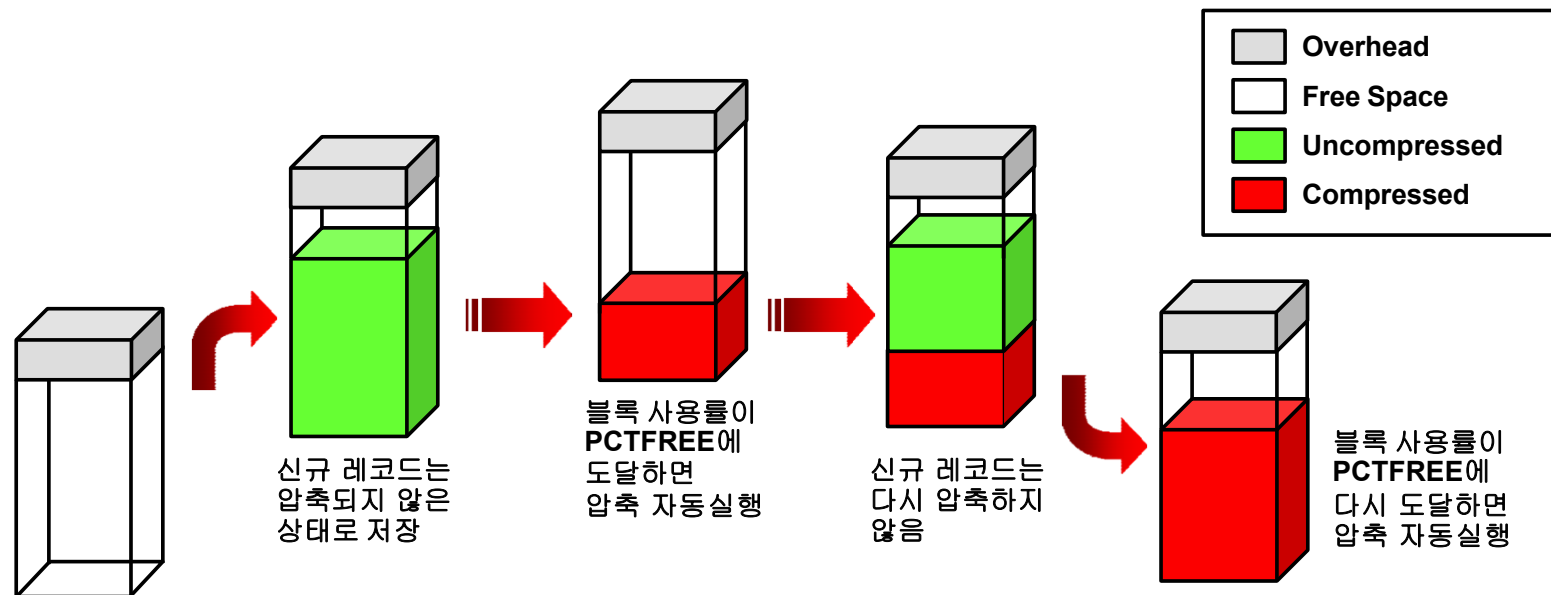
- OLTP, DW 에 모두 사용
- 2x ~ 4x 압축율

- 압축율과 성능의 균형

- DML과 Bulk load 두 경우 모두 compression 됨
- Buffer cache에 압축된 상태로 Data존재
- Query는 Buffer cache에서 적은 I/O로 많은 rows를 읽음으로 성능이 향상됨



Advanced Row Compression 동작방식



- 일반 OLTP INSERT에 대해 평상시엔 압축하지 않음
 - 일반 OLTP INSERT 성능 유지
- 블록 사용률이 PCTFREE에 도달하면 자동으로 압축
 - 공간 절약