



## ④ 크로스 집계

○ 크로스 테이블 - 행과 열이 교차하는 부분에 숫자 데이터가 들어간 것

↳ DB에선 다루기 힘들...

✱ 행 방향으로는 Data가 증가하고, 열 방향으로는 증가해서는 안됨.

↳ transaction table

트랜잭션 table  $\Rightarrow$  크로스 테이블

행 방향 = 크로스 집계!!  $\hat{=}$  pivot table의 기능  
↳ pivot graph 기능 포함.

크로스 테이블 = Uloop 사용 하여 table 만들 수 있음.

✱ 자료 데이터는 살펴볼 때는 엑셀은 BI.

✱ 데이터 양이 너무 많이 있으면, 그 모든 것을 매번 cross 집계 하는 것이 아니라,  
먼저 SQL로 집계하여 매번.

SQL도 피벗, 인회법이 가능함

Pandas에서 pivot, melt를 사용하여 집계 가능.

데이터마트 - 불특정 다수 입출력 처리.

② 맷시량 소스치에 의한 고스트

메모리 다 올라가지 않을 정도의 대량의 데이터를 선속하게 집계하려면,  
비리 데이터를 집계기 직접 할당 메모리 분할.

③ 데이터 캐시, 해킹 방법

① 모든 데이터를 메모리에 올리는 것.

② 압축과 분산에 의해 적당 줄이기

데이터를 가능할 한 적게 압축, 여러디스크에 분산.

↳ 데이터 로드와 다른 저장을 줄인 디스크 I/O를 병행 처리하는 것이 효과적

↳ 맷시량을 줄임, 분산 데이터 읽기/쓰기

↳ Mpp (Massive Parallel Processing)

ex) Amazon Redshift, Google BigQuery 등

Mpp는 데이터 집계에 최적화 되어있다.

열 지향  
column-oriented

Amazon Redshift

- 필요 column만 집계. 행과는 무관
- 필요없는 data I/O 필요 X
- 데이터 집계는 빠르나, 저장하는데 시간이 걸림.
- 압축 효율이 우수함.

US

행 지향  
row-oriented database

Oracle DB 등 RDB...  
MySQL

- 데이터 크거나 다지낼 형에 저장.
- 데이터 검색은 고속화 하기 위해 index 생성.

★ index에 의지하지 않는 고속 기술이 필요.

① MPP 데이터 베이스 접근 방식 - 병렬화에 의해 멀티 쿼리 실행하기

- 대량의 data를 읽기 때문에
- 1번의 쿼리 실행 시간이 길어짐.
- 양쪽 data의 전체 등으로, CPU 리소스를 필요로 하므로, 많은 process를 실행한  
가속화.

행 지향 DB는 쿼리, 하나의 쿼리를 하나의

- 스레드에서 실행됨.

- 일정 리미트의 쿼리를 분산 처리는 상용은 가용 X

Mpp에서는 하나의 쿼리를 다수의 작은 task로 분해하고,  
이를 병렬로 실행

select sum("금액") from '판매내역'

예를 들어 10만개의 Record로 구성된 1000개의 테스트로 할

① 데이터 분할  $\rightarrow \frac{100000}{1000}$

② 분산 처리  $\frac{100000}{1000}$   
 $\swarrow \searrow$  sum

③ 합계.

Mpp는, CPU 코어 수가 늘어남에 고속 실행.

( Load 병목 현상 방지 하기 위해 고르게 분산되어야 함 )

Mpp 데이터베이스

열지렁 ( 라스웨어 인저렁 )

대량 쿼리 실행

열지렁 ( 분산 스토리지 보관 )

리얼 data 의 경우

내부 data 저장소

Superset

Redash

Kibana - 로그스택

마이크로소프 BI 솔루션

쿼리서버

수집된 데이터

① Data Mart를 먼저  
만들어야함

Druid

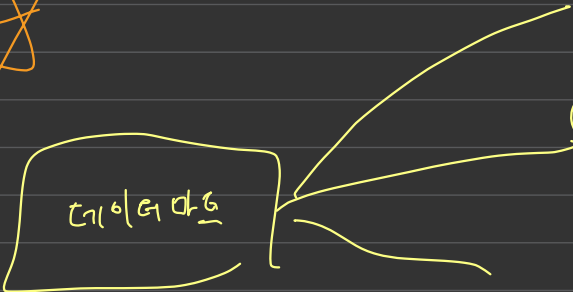
실시간 data 저장

실시간 쿼리

② BI 도구

이것까지 데이터의 장기적인 데이터를 시각화하거나, 정제의 큰 데이터를 세부적으로  
바꿀 수 있는 대시보드

① 리얼 data 리포트

② 중요 시도를 위한 높은 민감성이 있는  
대시보드. 

③ 주기적인 데이터 집계

★ 데이터 마스의 기원부터.

시각화에 필요한 정보만을 담은 데이터 마스가 필수적!!!

OLAP - 시각화에 적합한 데이터 마스 만들기

OLAP 쿼리 - 데이터 분석을 위해 만들어진 다차원 데이터

↳ 코호스 집계하는 쿼리가 OLAP

BI 도구와 생각은 대로의 그래프를 만들기 위해서, 이미 존재하는

데이터를 그대로 시각화 X, 그래프에 맞춘 뒤 "다차원 데이터/쿼리"

★ 시각화에 적합한 데이터 마스를 만드는 것은, BI 도구를 위한

★ 최적화 데이터 쿼리 만드는 process.

## ★ 테이블 지정기법

### 마스터 테이블

- 트랜잭션에서 참조되는 각종 정보.

(상항에 따라 다수 존재함)

### 트랜잭션 테이블

- 시간과 함께 생성되는 데이터 기록 단위

고객 마스터  
고객 ID  
성명  
주소

점포 마스터  
점포 ID  
점포명  
지역

상품 마스터  
상품 ID  
상품명  
상품 카테고리

판매 이력  
판매 ID  
상품 ID  
점포 ID  
고객 ID  
금액

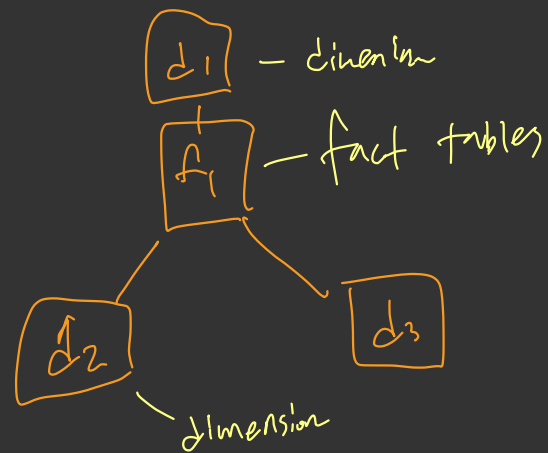
// DWI //

다양한 테이블

팩스 테이블



데이터 마트를 만들 때는 팩트 테이블을 중심으로 여러 디멘션 테이블  
구조를 짜는 것이 좋다. (스타스키마)



디멘션 테이블 작성하기

정규화에 의해 분해된 테이블을  
원래의 구조로 다시 짜서 테이블로  
만들.

(정규화를 반其道 적용)

데이터 마트에서

스타스키마 이용

① 단점

② 데이터 분석에 불리