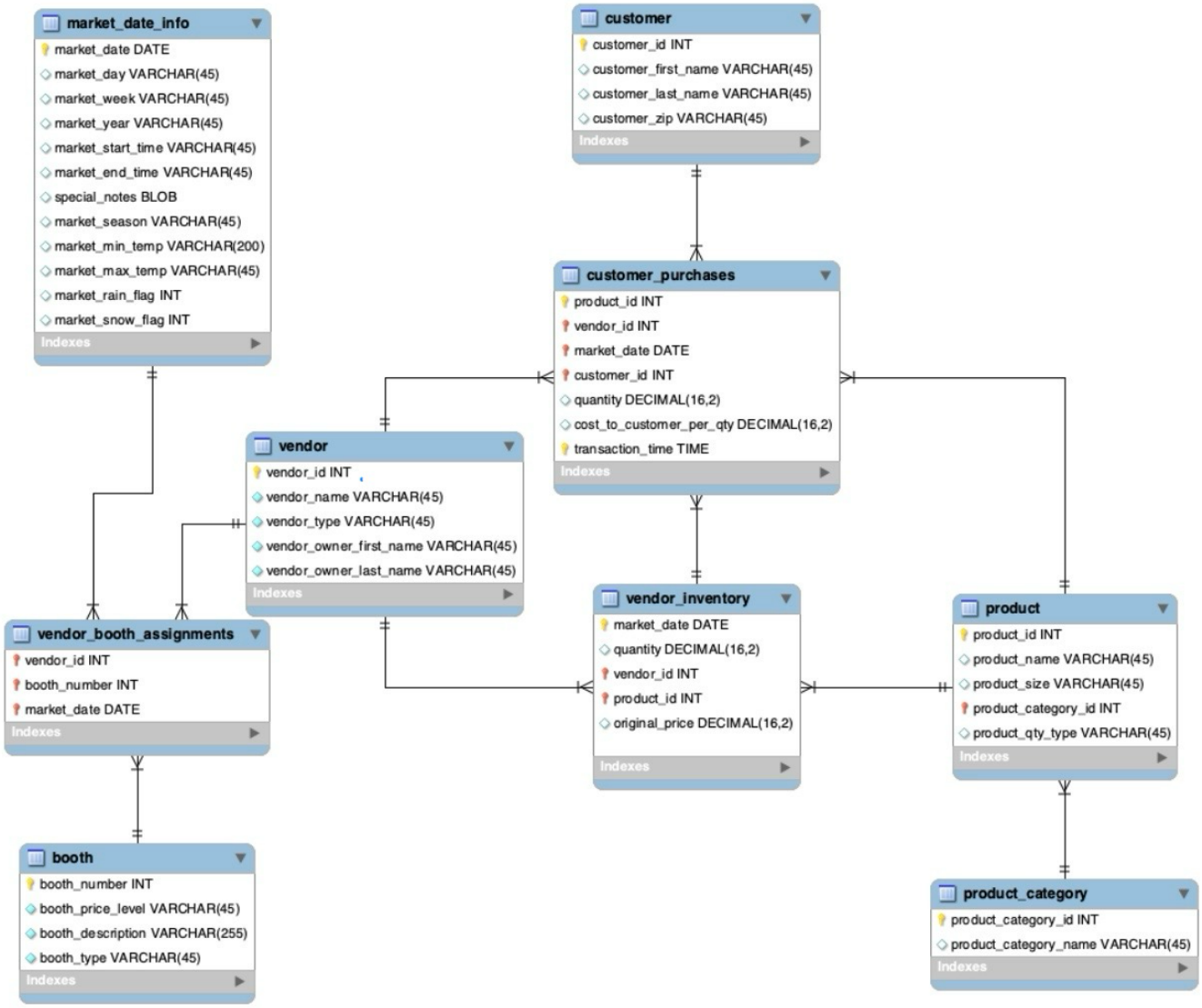
Agenda

1. CTE and Views
2. Busting a common myth about the COUNT()
3. Only select the columns that you really need
4. LIMIT is a trap
5. Use EXISTS() instead of COUNT()
6. Use APPROX_COUNT_DISTINCT instead of COUNT(DISTINCT)
7. Replace Self-Join with Windows Function
8. Trim your data early and often
9. Use MAX() instead of RANK()
10. Order your JOINs from larger tables to smaller tables
11. Does WHERE sequence matters?
12. Should we push ORDER BY to the end of the query?


market_date_info
market_date DATE
market_day VARCHAR(45)
market_week VARCHAR(45)
market_year VARCHAR(45)
market_start_time VARCHAR(45)
market_end_time VARCHAR(45)
special_notes BLOB
market_season VARCHAR(45)
market_min_temp VARCHAR(200)
market_max_temp VARCHAR(45)
market_rain_flag INT
market_snow_flag INT
Indexes
customer
customer_id INT
customer_first_name VARCHAR(45)
customer_last_name VARCHAR(45)
customer_zip VARCHAR(45)
Indexes
customer_purchases
product_id INT
vendor_id INT
market_date DATE
customer_id INT
quantity DECIMAL(16,2)
cost_to_customer_per_qty DECIMAL(16,2)
transaction_time TIME
Indexes
vendor
vendor_id INT
vendor_name VARCHAR(45)
vendor_type VARCHAR(45)
vendor_owner_first_name VARCHAR(45)
vendor_owner_last_name VARCHAR(45)
Indexes
vendor_booth_assignments
vendor_id INT
booth_number INT
market_date DATE
Indexes
vendor_inventory
market_date DATE
quantity DECIMAL(16,2)
vendor_id INT
product_id INT
original_price DECIMAL(16,2)
Indexes
product
product_id INT
product_name VARCHAR(45)
product_size VARCHAR(45)
product_category_id INT
product_qty_type VARCHAR(45)
Indexes
booth
booth_number INT
booth_price_level VARCHAR(45)
booth_description VARCHAR(255)
booth_type VARCHAR(45)
Indexes
product_category
product_category_id INT
product_category_name VARCHAR(45)
Indexes

Ad hoc Reporting

↳ DA/DE/DS

Business Question

↓

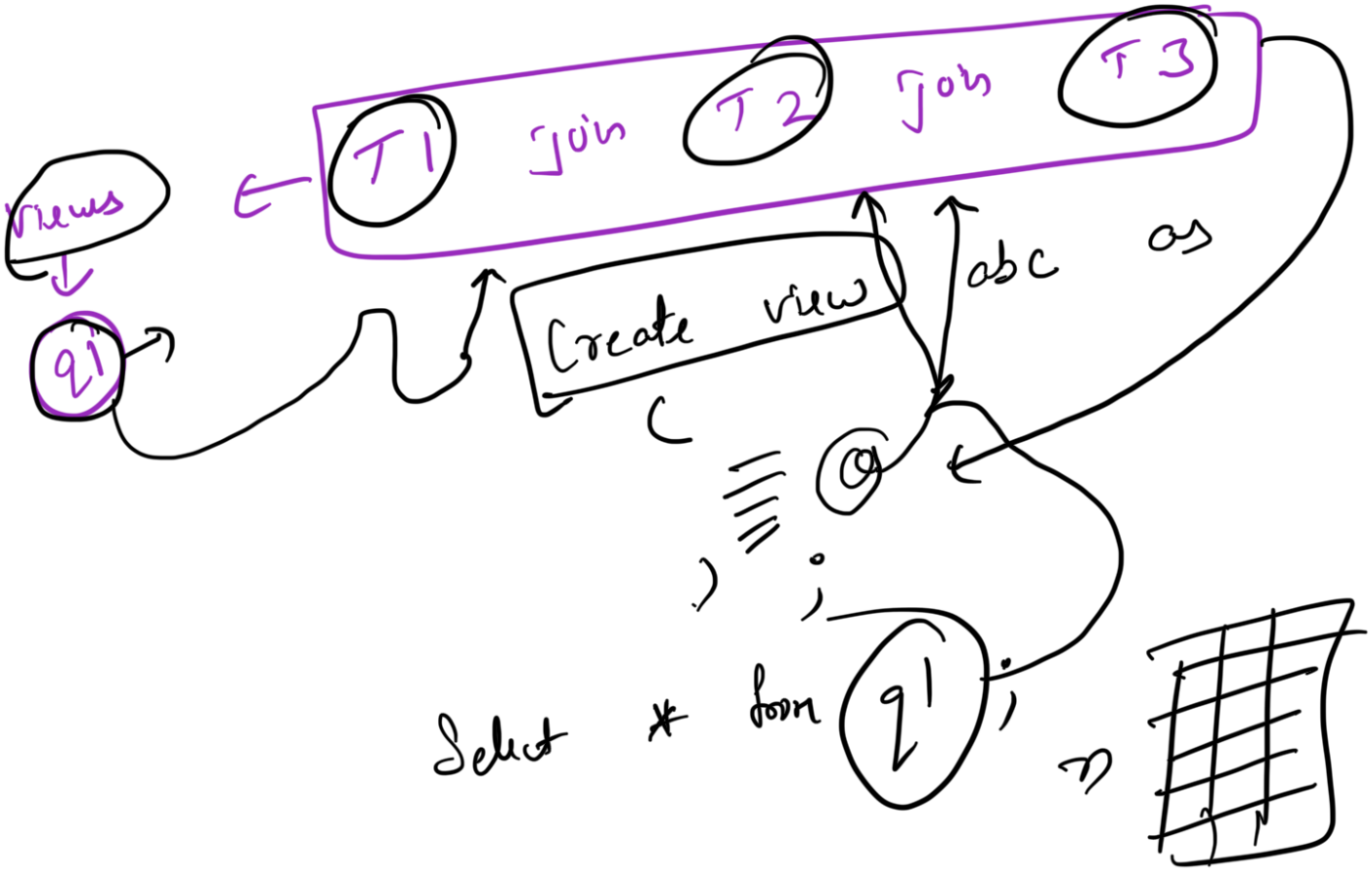
Data Question

↓

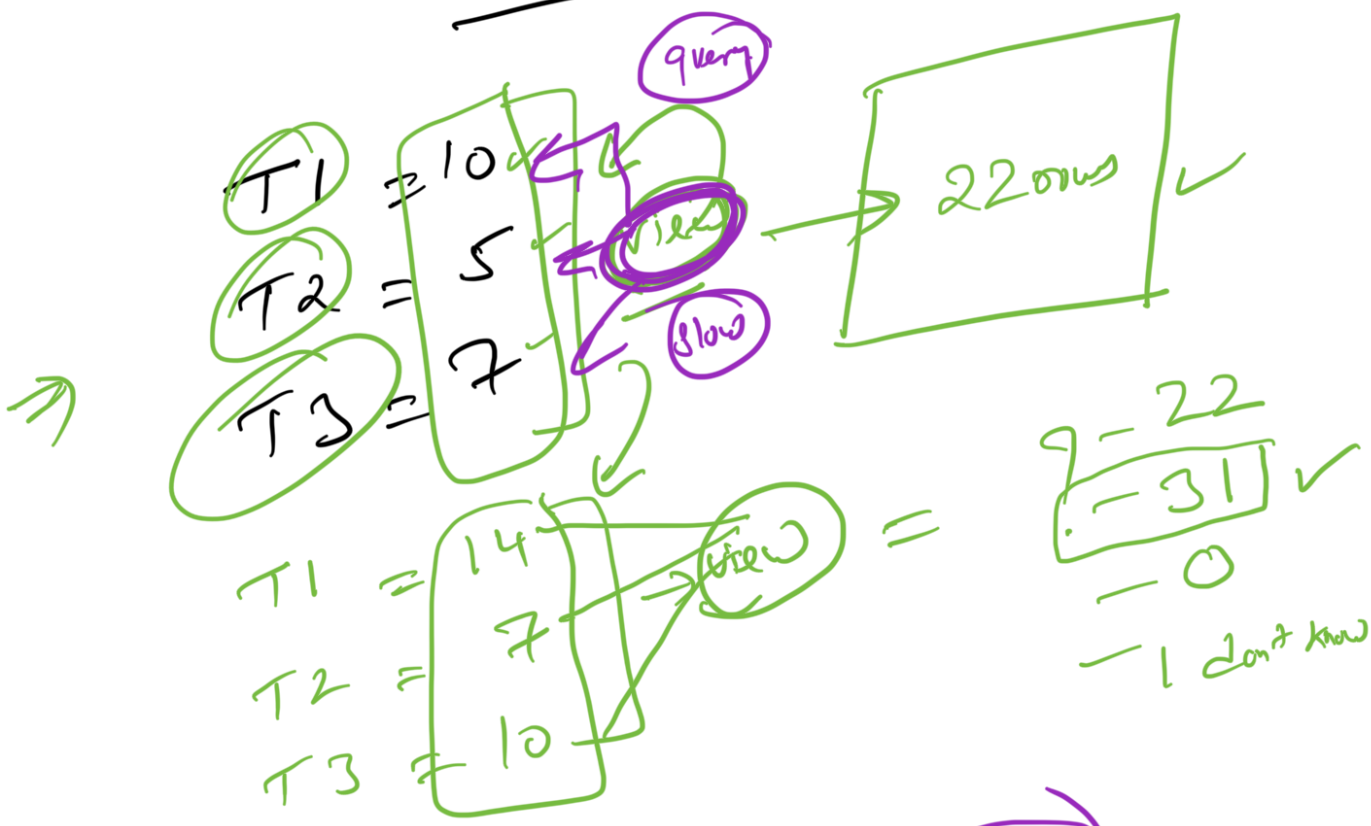
Data Answer

↓

Business Answer



Views ⇒ Storing the query



⇒ Materialized views = query + Data

Stale ↳ Rebuilding

② Common table expressions (CTE)

Syntax:-

with [query_alias] AS

([query])

⇒ [query 2 _ alias] as
(
query 2)
)
Select * from query_alias;

View	CTE
→ Static way of storing queries	→ Dynamic way of writing queries.
→ In view, DBA can give more fine grained access	→ There is no such access management.
→ Query to be used multiple times	→ Ad hoc queries (occasionally or just one) use CTE

Datasets ② ethereum Dataset
↳ 55 facts

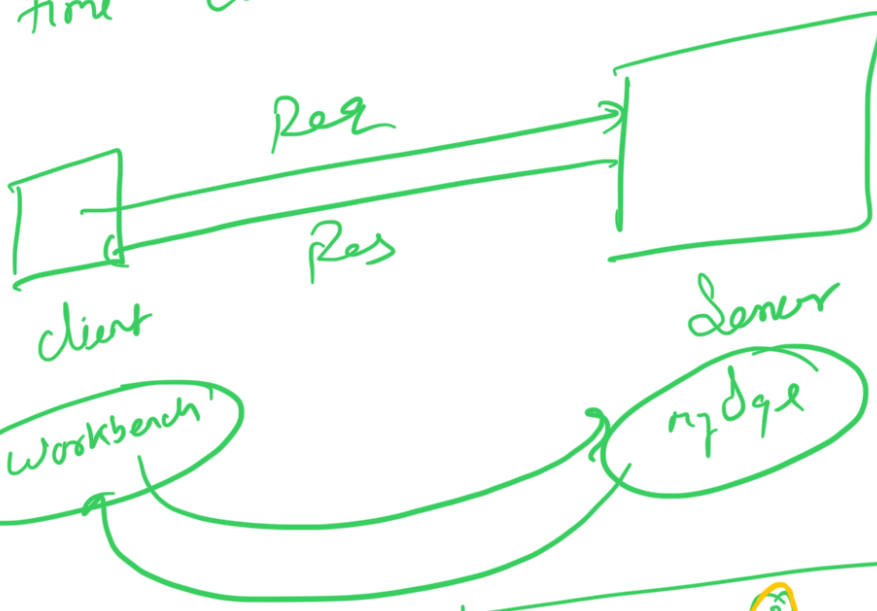
1 Bitcoin =

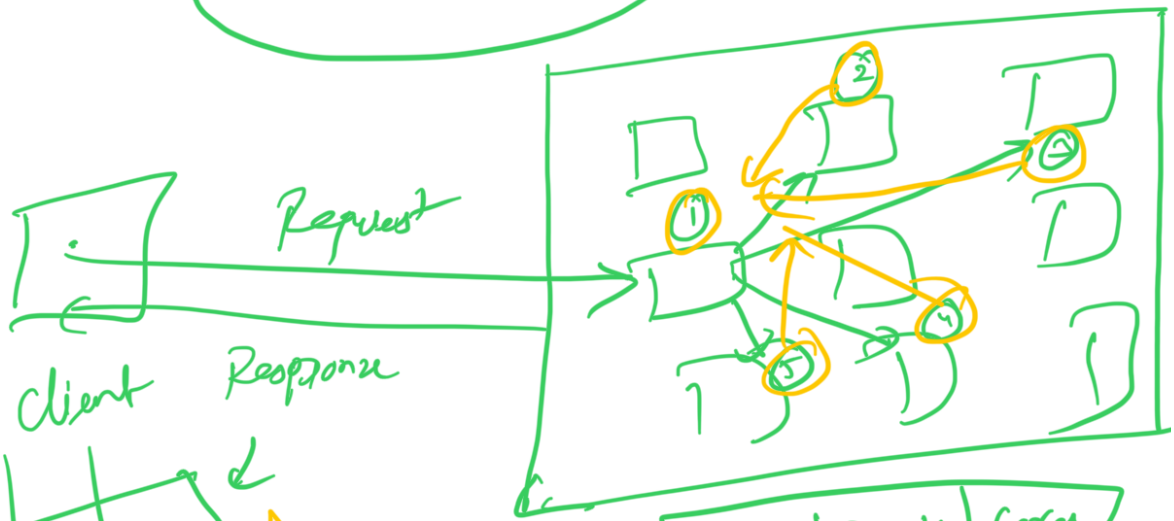
(1) Elapsed time : It is also query execution time, is the total time by a query to

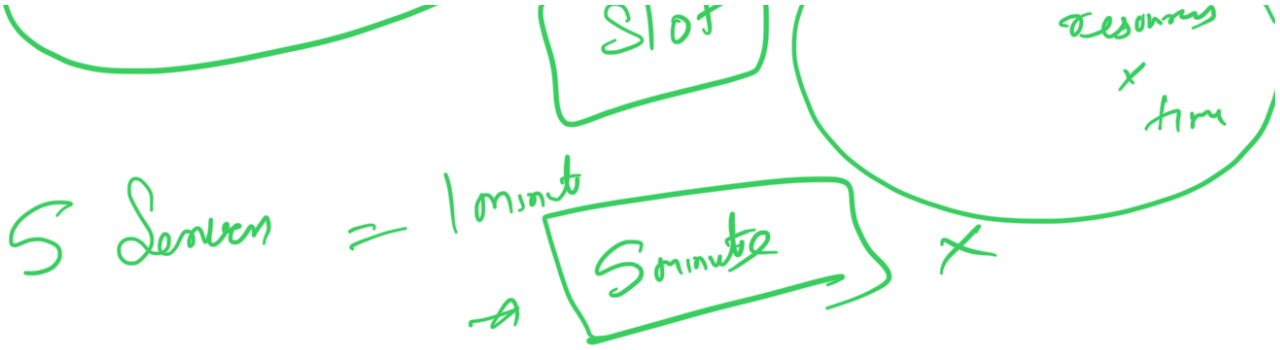
taken
ones from the moment
it is submitted until
results are returned.

→ less elapsed time ⇒ More faster query

(2) Slot time Consumed :-







5 Servers = 1 minute

→ 5 minutes

×

→ Slot time Consumed refers to the amount of Computational resources (R+C) that a query Consumes during its execution.

Slot

→ More slots Consumed, faster will be query, More Cost you need to Scan.

③ Bytes shuffled :- redistribution of data across different slots for processing, this is known as shuffling.

→ it represents the volume of data that is moved between different parts of distributed system.

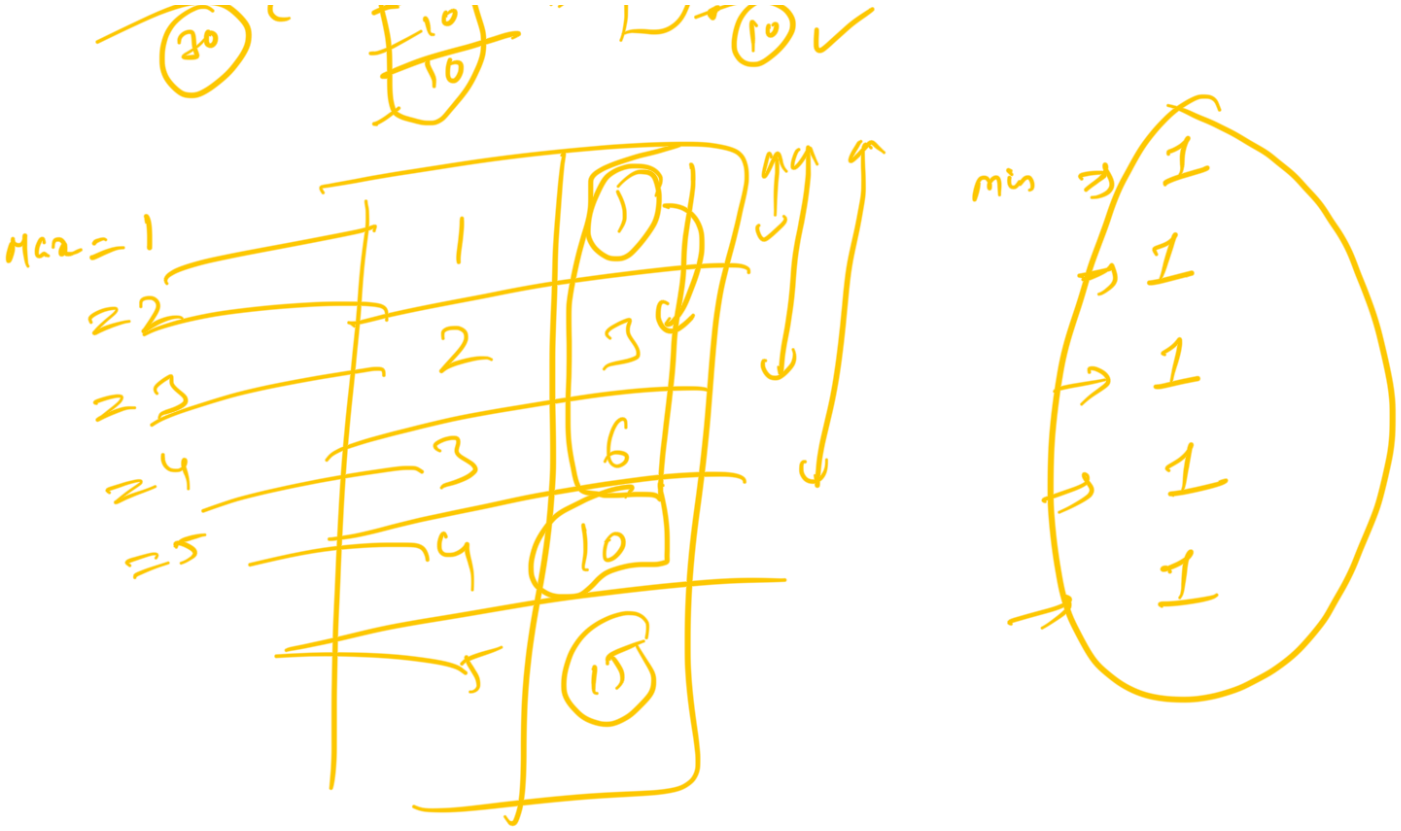
Less shuffling = More performance

⑨ Bytes Spilled to Disk :-

when a query intermediates results or data exceeds the memory capacity of the allocate slots, some of that data may be written to disk temporarily for storages

less Bytes shuffled = more performant query

lower elapsed times, efficient slot usage, minimize bytes shuffled and reduce bytes split to disk.


20
10
10
10
Hash = 1
2
3
4
5
1
2
3
4
5
1
3
6
10
15
Miss
1
1
1
1
1