DRAWINGS

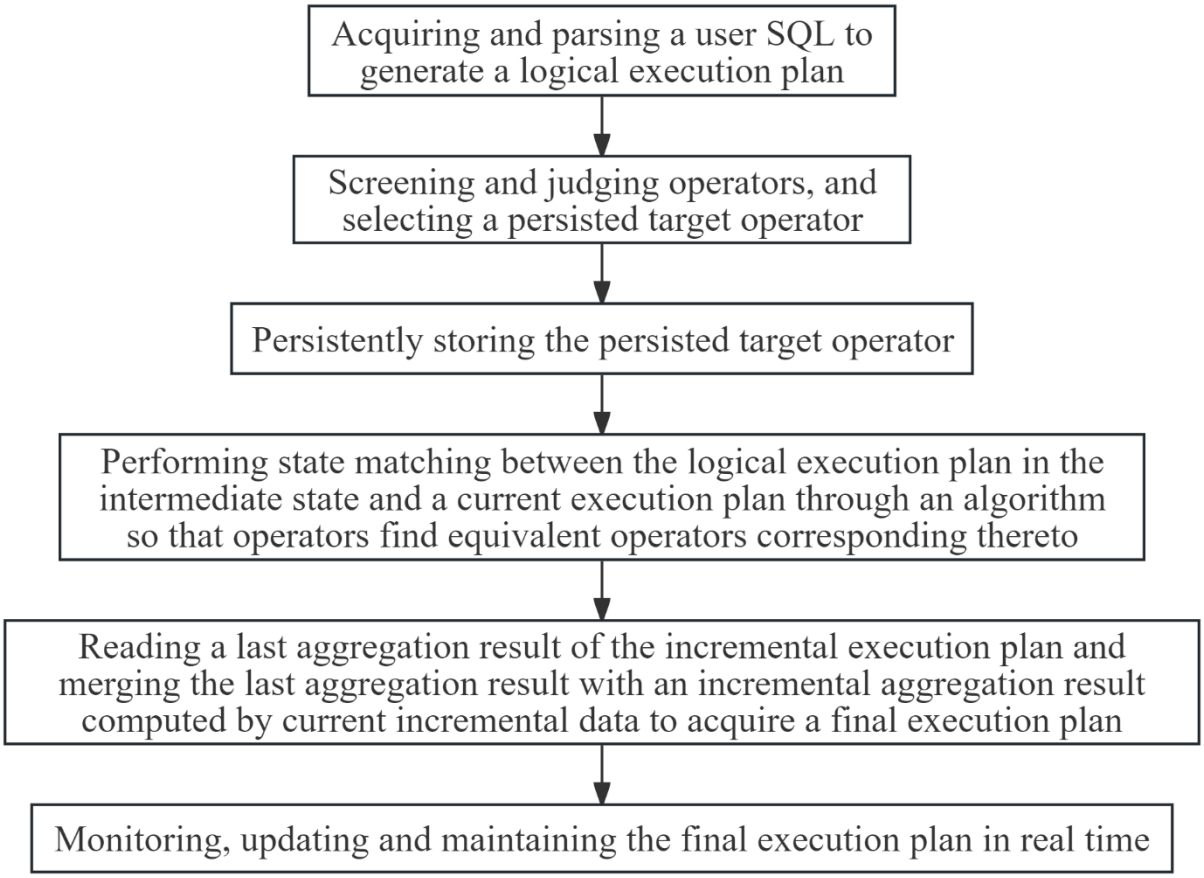


FIG. 1

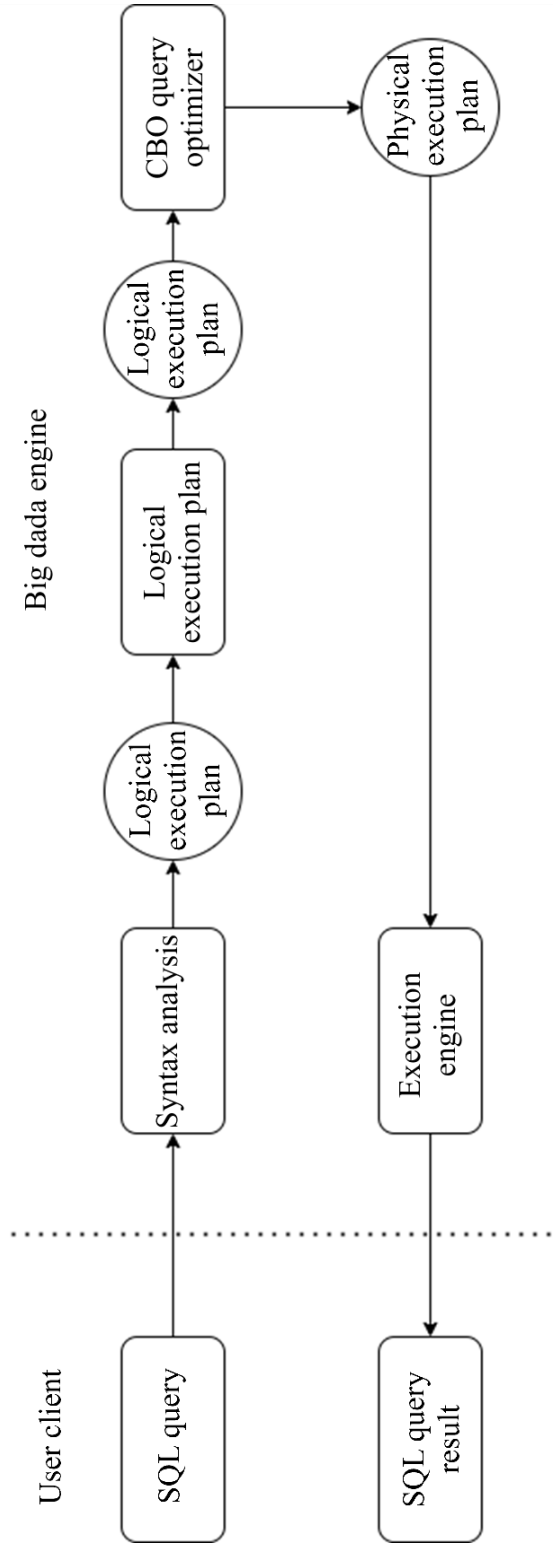

User client
Big dada engine
SQL query
Syntax analysis
Logical execution plan
Logical execution plan
Logical execution plan
CBO query optimizer
Physical execution plan
Execution engine
SQL query result

FIG. 2

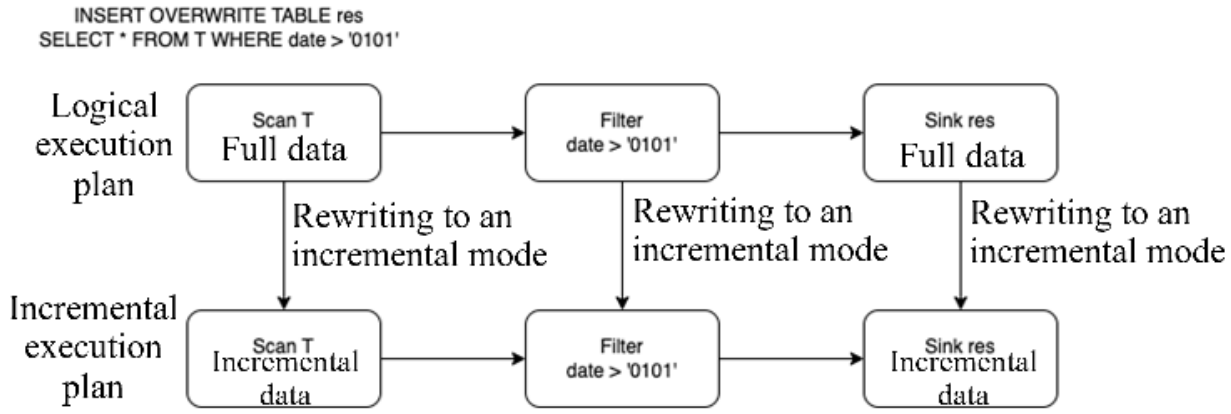

INSERT OVERWRITE TABLE res
SELECT * FROM T WHERE date > '0101'
Logical execution plan
Scan T
Full data
Filter
date > '0101'
Sink res
Full data
Rewriting to an incremental mode
Rewriting to an incremental mode
Rewriting to an incremental mode
Incremental execution plan
Scan T
Incremental data
Filter
date > '0101'
Sink res
Incremental data

FIG. 3

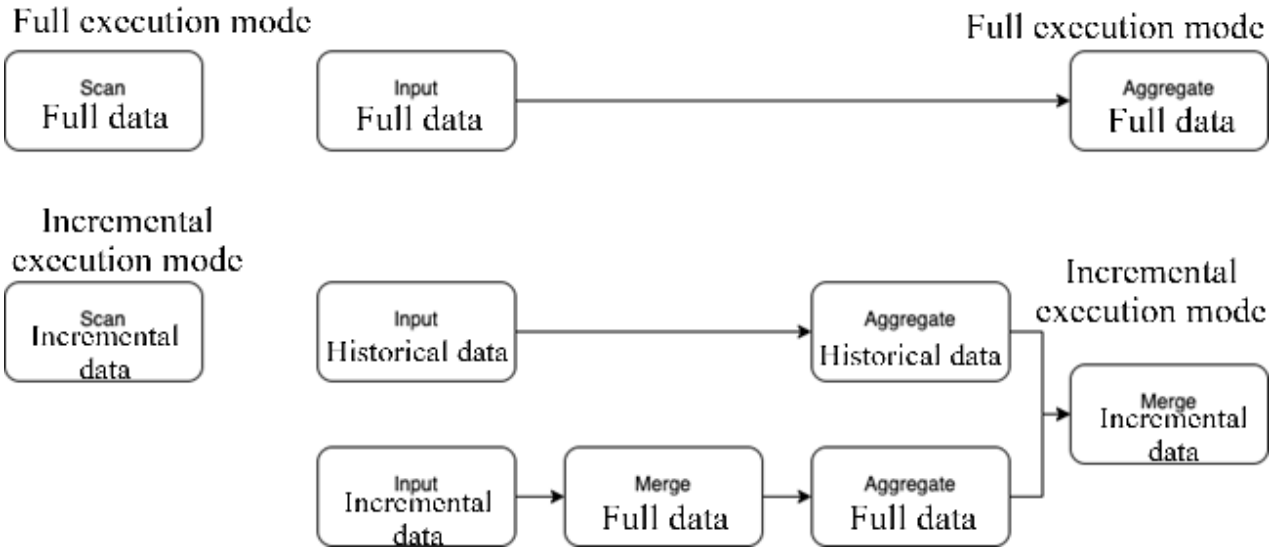

Full execution mode
Scan
Full data
Input
Full data
Full execution mode
Aggregate
Full data
Incremental execution mode
Scan
Incremental data
Input
Historical data
Aggregate
Historical data
Incremental execution mode
Merge
Incremental data
Input
Incremental data
Merge
Full data
Aggregate
Full data

FIG. 4

```
INSERT OVERWRITE TABLE res
SELECT * FROM
   (SELECT *, AVG(c1) as _avg FROM T2)  t2
INNER JOIN
   (SELECT * FROM T1) t1
on t1.join_key = t2.join_key
```

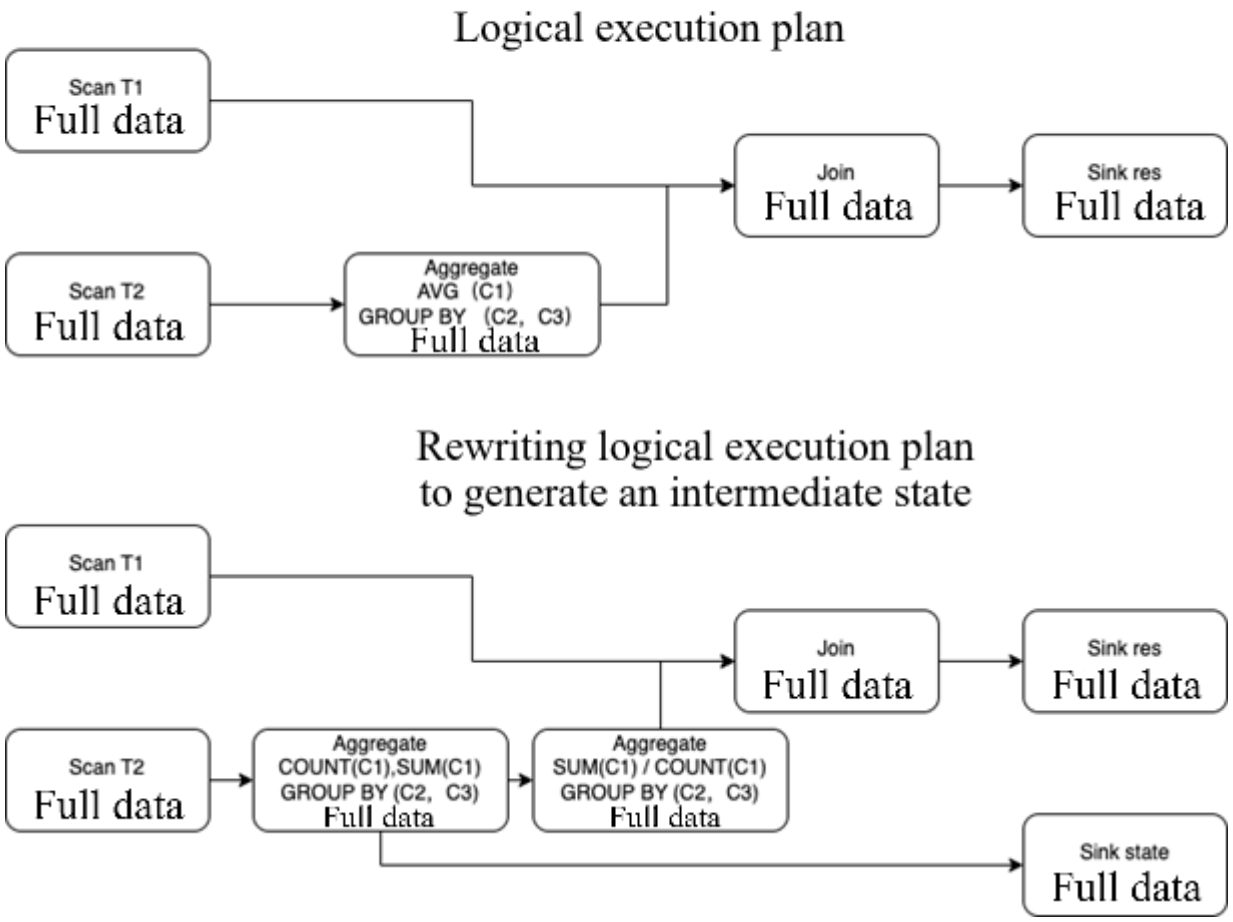


FIG. 5

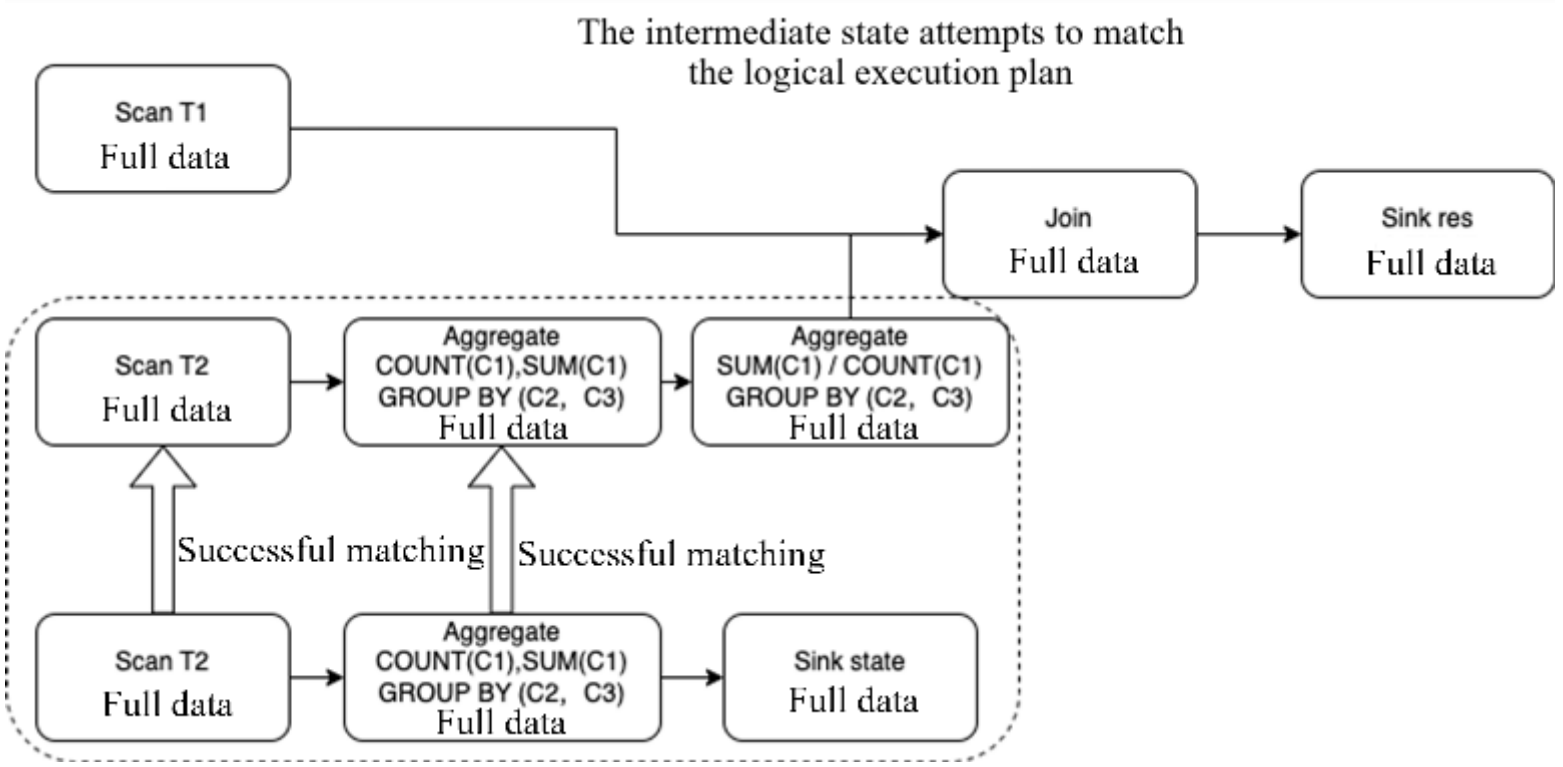

The intermediate state attempts to match the logical execution plan
Scan T1
Full data
Join
Full data
Sink res
Full data
Scan T2
Full data
Aggregate
COUNT(C1),SUM(C1)
GROUP BY (C2, C3)
Full data
Aggregate
SUM(C1) / COUNT(C1)
GROUP BY (C2, C3)
Full data
Successful matching
Successful matching
Scan T2
Full data
Aggregate
COUNT(C1),SUM(C1)
GROUP BY (C2, C3)
Full data
Sink state
Full data

FIG.6

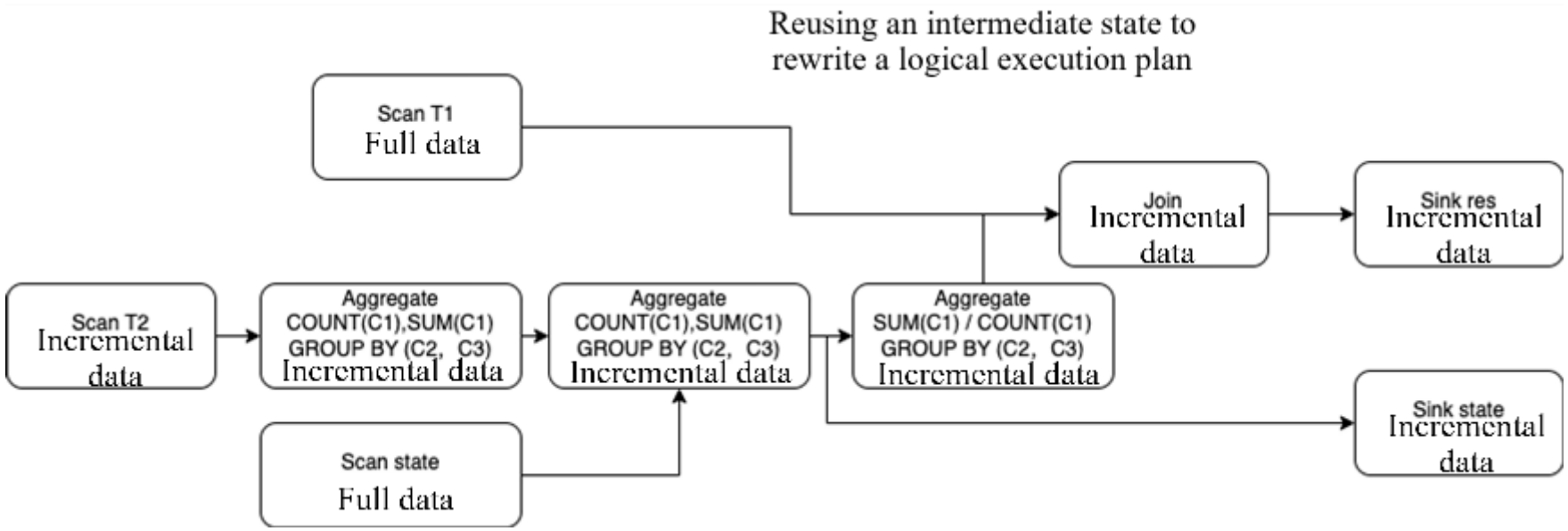

Reusing an intermediate state to rewrite a logical execution plan
Scan T1
Full data
Join
Incremental data
Sink res
Incremental data
Scan T2
Incremental data
Aggregate
COUNT(C1),SUM(C1)
GROUP BY (C2, C3)
Incremental data
Aggregate
COUNT(C1),SUM(C1)
GROUP BY (C2, C3)
Incremental data
Aggregate
SUM(C1) / COUNT(C1)
GROUP BY (C2, C3)
Incremental data
Sink state
Incremental data
Scan state
Full data

FIG. 7

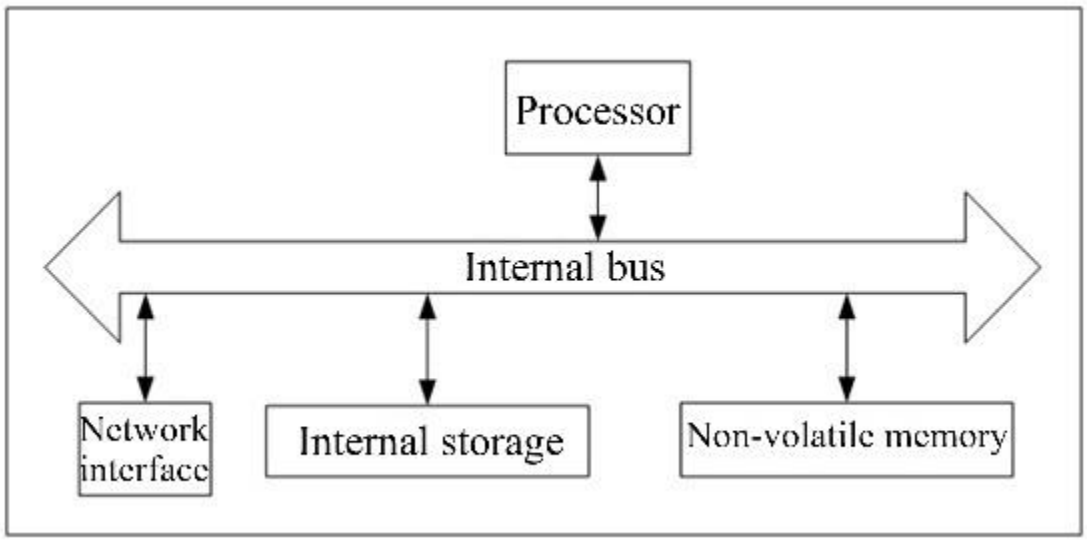

Processor
Internal bus
Network interface
Internal storage
Non-volatile memory

FIG. 8

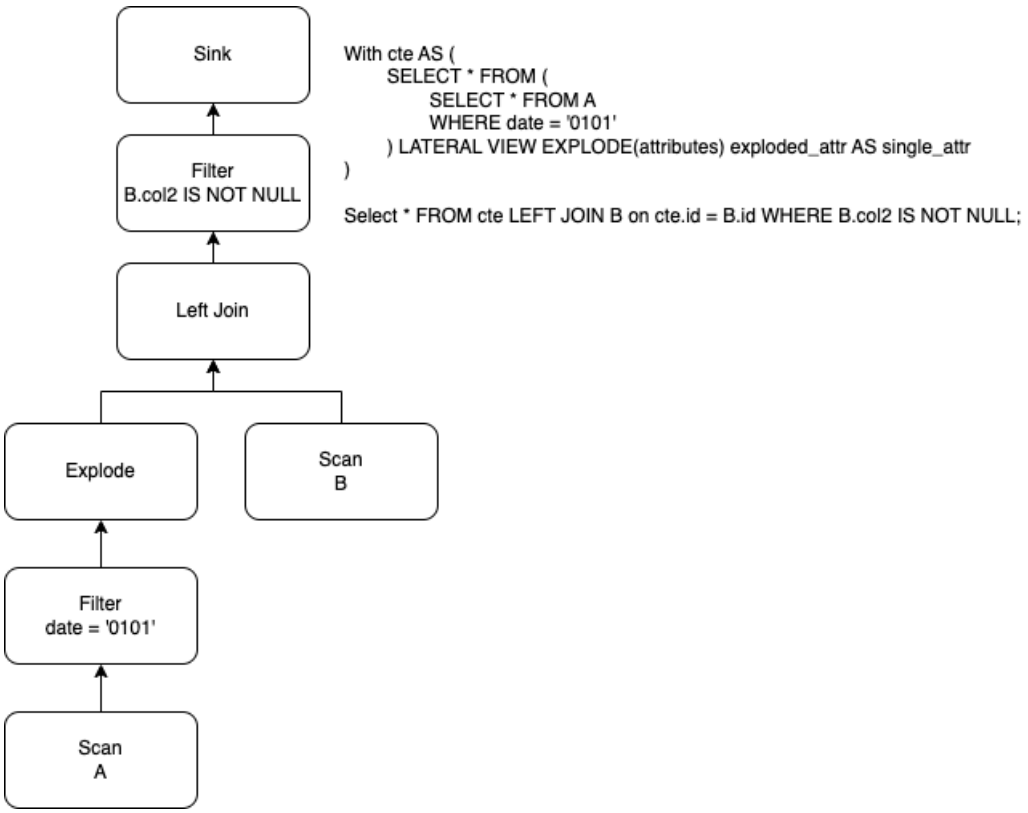

Sink
Filter
B.col2 IS NOT NULL
Left Join
Explode
Scan
B
Filter
date = '0101'
Scan
A
With cte AS (
SELECT * FROM (
SELECT * FROM A
WHERE date = '0101'
) LATERAL VIEW EXPLODE(attributes) exploded_attr AS single_attr
)
Select * FROM cte LEFT JOIN B on cte.id = B.id WHERE B.col2 IS NOT NULL;

FIG. 9