Other Recovery Related Structures

Transaction Table – has an entry for each active transaction. Contains ID, status, last LSN.

committed
inprogress
aborted.

handled by transaction Manager

Dirty page table – ~~[illegible]~~ has one entry for each dirty page. ~~[illegible]~~ Entry has a field recLSN w/c is 1st LSN when page became dirty.

Checkpts – ~~like~~ a snapshot of DBMS state, done to reduce the amount of work ~~Write Ahead Logging~~ during a restart.

In Aries has 3 steps, ① a begin checkpt record is written ② construct an end checkpt w/c consists of transaction table dirty page table, ~~[illegible]~~ end ckpt written to stable storage. ③ master While end ckpt is being made DBMS continues executing other transactions

record of LSN begin ckpt written to known place on stable storage

Database Tuning

We have designed and deployed a DB now what to improve its performance.

Need to find out typical workload must support

To find out can monitor system using UTLBSTAT.SQ (must be SYS using SVRMGRL) It is in $ORACLE_HOME/rdbms/admin

A workload description consists of

① a list of queries & frequencies & tables involved & attributes retained & joined

② a list of updates & frequencies & ~~types~~ & attributes

③ Performance goals for each type of query & update

To improve performance want to consider
① choice of indexes ② does conceptual schema need to be modified ③ can we improve queries?

Index Selection

1. index only if will benefit some query or update prefer indexes that speed up more than one query
2. Attributes mentioned in where clauses are candidates for indexing. If = then hash if range then B+ or ISAM.
3. Consider indexes w/ multi attributes ① if where clause includes conditions on more than attributes ② if allows index only plans
4. Clusters range queries don't cluster if index only strategy exists!
5. Use Hash index only if will use index nested loop join a lot or if have very important equality query
6. consider cost of updating index. (Note sometimes might improve)

Choice of Conceptual Schema

Choice of conceptual schema should be guided by consideration of the queries and updates in the workload.

- Might want to settle for 3NF rather than BCNF — b/c BCNF not dependency preserving
- If two ways to decompose into 3NF or BCNF choose the one easier on the workload
- might want to split a table already in BCNF called vertical partitioning.
  Ex) May be have frequently occurring query and want smaller table size
- horizontontal partitioning make two tables w/ identical schemas. Ex) Car brand within cartable.