Notes 3

Page 22 – Describe and explain questions

## Partitioning

* Don’t need all of it or are more interested in some of it.

**When will it be useful to break a table into parts?**

* Partition which will be based on certain values which is horizontal fragmentation into groups of rows that make sense. Company field, city, geographic. All the cork all the Waterford.

**Would you take a large table where you are more interested in some of the values than the others?**

* Certain date of birth, more interested in younger ppl.

**How do I recombine the fragments to get original table? Recombination of horizontal fragmentation how? Same structure in each fragment, how to recombine?**

* **Union** operator in SQL. However it may not be good because splitting up the files requires more seek time to locate.

**Vertical Fragmentation. Why?**

* Some attributes may be used more frequently some may be larger and used less often (BLOBS)
* Recombined with a join so needs a common attr. Duplicate the key.
* Large set of columns some columns used by a lot and are small, some are large and costly to store/access

## Denormalisation

*Normalisation* breaks down big table into many smaller ones. Tendency to produce lots of small tables requiring joins afterwards which increases seek times. Generally used to solve insert/update anomaly but it increases costs to do Select statements. Completely focussed on writes and not focussed on reads.

*Denormalisation.* Improved read performance by reoccurrence of the anomalies solved by normalisation.

## Data Compression/Decompression

Char fields – Remove/trim the whitespace to compress.

IP addresses can be compressed by storing (if a known domain for all) the top level octet elsewhere enabling the blocks of ip addresses to be compressed.

Looking at whitespace or looking at common things at the start.

**Encoding**

A is more common than Z so larger bit sequences are allocated for Z and less for A.

“the” may be encoded as it is very common in English.

Downside is decompression is required so processing cost.

Contiguous is a request made when making a table to enable Clustering. Which is what you do when you decompress. Defragmentation. Don’t mind fragments nearby as it is only a disk rotation to find it. Take stuff that is close together

Synonyms

Hashing, binary search, tree search