

# Tuning Queries

Ch 20

Frequency of a search (retrieval) on an attribute (zipcode) is very high, build an index for that attribute.

Customer

Acct#	Name	Addr	zip	tel
1			<del>Z5</del>	Z2
2			Z3	
3			Z1	
4			Z3	

ZipcodeIndex

Zip	Record
Z1	3
Z2	1
Z3	2, 4
Z5	1

Sorted ↓

When the index attribute value gets modified (insert/delete/update) operations on the original table, the index entries need to be corrected <sup>↑ sorted</sup> accordingly.

? Frequency of changes to the attribute (zip) values.

Index tuning wizard: Creation/Deletion of an index.

Statistics on attributes: }  
no. of unique values  
no. of retrieval  
no. of update

## DB Objects

Tables  
Views  
Index  
= = =

one master data file .mdf  
one transaction log file .ldf

one master data file .mdf  
Several secondary data files .ndf  
one transaction log file .ldf

grouped into filegroups

By placing different secondary files (or filegroups)

in different disks,

disk access on multiple objects from different disks  
will <sup>improve</sup> query performance.

## De Normalization

{ Merging (Joining) several base tables  
to provide a join result of frequently used queries.

Denormalized table is temporary.

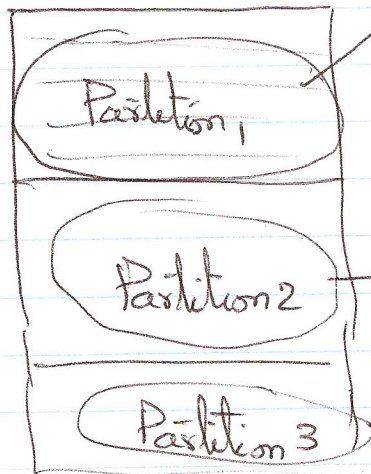
When a base table gets modified,

the denormalized table must be revised  
or recreated.

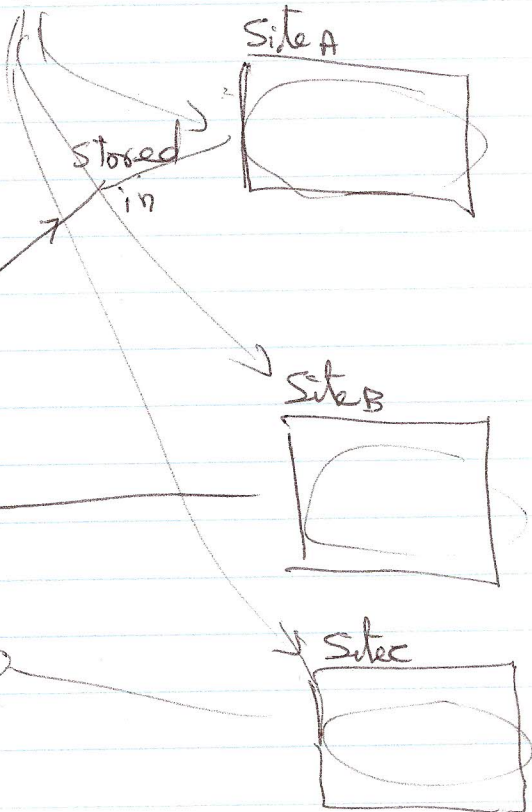
# Partitioning Tables

## Horizontal partitioning

Virtual table  
Customer

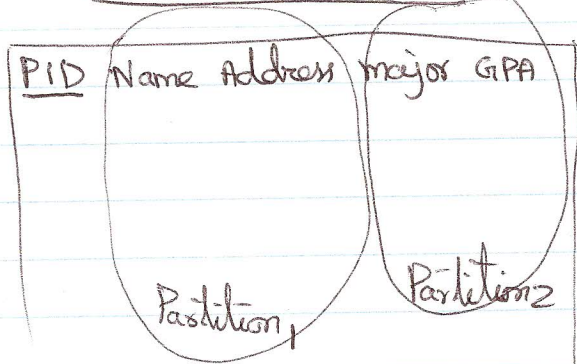


Queries



## Vertical Partitioning

Virtual Student table



Queries



Both must include  
Primary key PID