Indexes

—

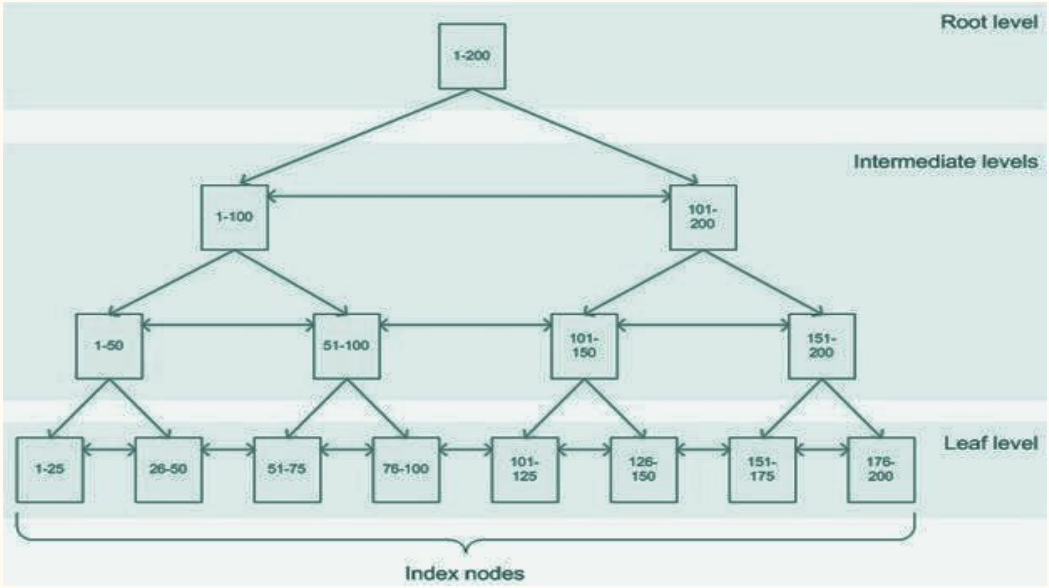
Theresa

Indexes are structures meant to improve the performance of SELECT queries.

- Indexes make SELECT queries go faster
- Indexes are independent of a database's tables. Creating/Dropping indexes won't change the data or affect the database.
- Indexes are maintained automagically for you by the Oracle Database after DML (INSERT, UPDATE, DELETE) statements are executed.

How?

B-Trees



- Indexes are organized by a B-Tree scheme. (There are other ways to organize indexes (e.g. bitmap, hash cluster, etc.), but B-Tree is the most commonly used scheme and thus the default.)
- B-Trees consist of *root* blocks, *branch* blocks, and *leaf* blocks.
- Root(s) have pointers to the branch blocks which hold pointers to the leaf block that contains the data and its rowid in the database.
- Branch blocks hold the minimum key prefix pointer needed to be able to choose the needed branch/leaf.
    - Ex: 101 branch block and the block next to it starts at 151, so we know 101 block goes up to 150.
- Leaf blocks are double linked, which means they contain pointers to the block before and after it, even if they don't know anything else about the leaf blocks.
    - Ex: the 151-175 leaf block knows that the block after it starts at 176 and that the block before it ends at 150.

Creating Indexes

```
CREATE INDEX indexname
ON table (column)
TABLESPACE tablespace
STORAGE (INITIAL 20K
NEXT 20K
PCTINCREASE 75)
COMPUTE STATISTICS;
```

- The syntax for creating an index in PL SQL
- The only required part for making a basic index is “CREATE INDEX indexname ON table (column);” which defines
  - the name you want to give your index
  - the table you want to index
  - And the column(s) you want to include in your index
- No other parameters are required, but there’s a lot of options for indexes. (These settings will simply go to their default values if not specified.)
- In this example syntax, “TABLESPACE tablespace” allows you to declare which tablespace you want the index to be created in.
- “STORAGE (INITIAL 20K NEXT 20K PCTINCREASE 75)” showcases some of the storage options you can set for your index.
  - INITIAL 20K specifies the size of the first “area” to be indexed.
  - NEXT 20K specifies the size of the second “area”
  - PCTINCREASE 75 will make all blocks after the first two allow a larger size that increases by 75% more of the block before it.
- Specifying memory size in storage is helpful when handling large databases and managing disk space becomes important.
- COMPUTE STATISTICS is a deprecated option. It tells the database to calculate statistics on the index every time it’s built/rebuilt. But Oracle Database now computes statistics automatically during index building.

- COMPUTE STATISTICS is kept in order to maintain backward compatibility.
- The options shown here aren't exhaustive, there's lot of options for setting up your index.
- You can create UNIQUE indexes by simply inserting the keyword into the CREATE INDEX statement. "CREATE UNIQUE INDEX"
    - Unique indexes don't allow duplicate values in the key column.
    - Can signify other kinds of indexes besides UNIQUE by adding the term into the CREATE INDEX statement

Some advanced stuff

- Invisible Indexes
- Key-Compressed Indexes
- Online Indexes
- Function-Based Indexes

- Invisible indexes are ignored by the optimizer but still automatically maintained whenever DML statements execute.
  - Exist as an alternative to dropping indexes or making them unusable
  - Useful for using temporary indexes without affecting application performance
- Key-compressed indexes have their index key broken into the prefixes and suffixes. Suffixes with the same prefix are all attached together.
  - Useful for saving space with unique mult-icolumn indexes
  - (same) prefix ---- suffix (unique)
    ---- suffix
    ---- suffix
- Online indexes are built online
  - Useful for allowing updating of tables while building indexes on said tables at the same time
- Function-based indexes store precomputed values from a function to allow queries to use indexes even when using a function on the indexed column.
  - How? You make an index of a function on the column and store the product of the function in the index instead of the original column.

References

ORACLE DATABASE ONLINE DOCUMENTATION ADMINISTRATOR GUIDE

http://docs.oracle.com/cd/B28359_01/server.111/b28310/indexes001.htm#ADMIN11709

http://docs.oracle.com/cd/B19306_01/server.102/b14200/clauses007.htm#g1058547

ASKTHEORACLE.NET

http://www.asktheoracle.net/how-do-indexes-work-internally-in-oracle.html