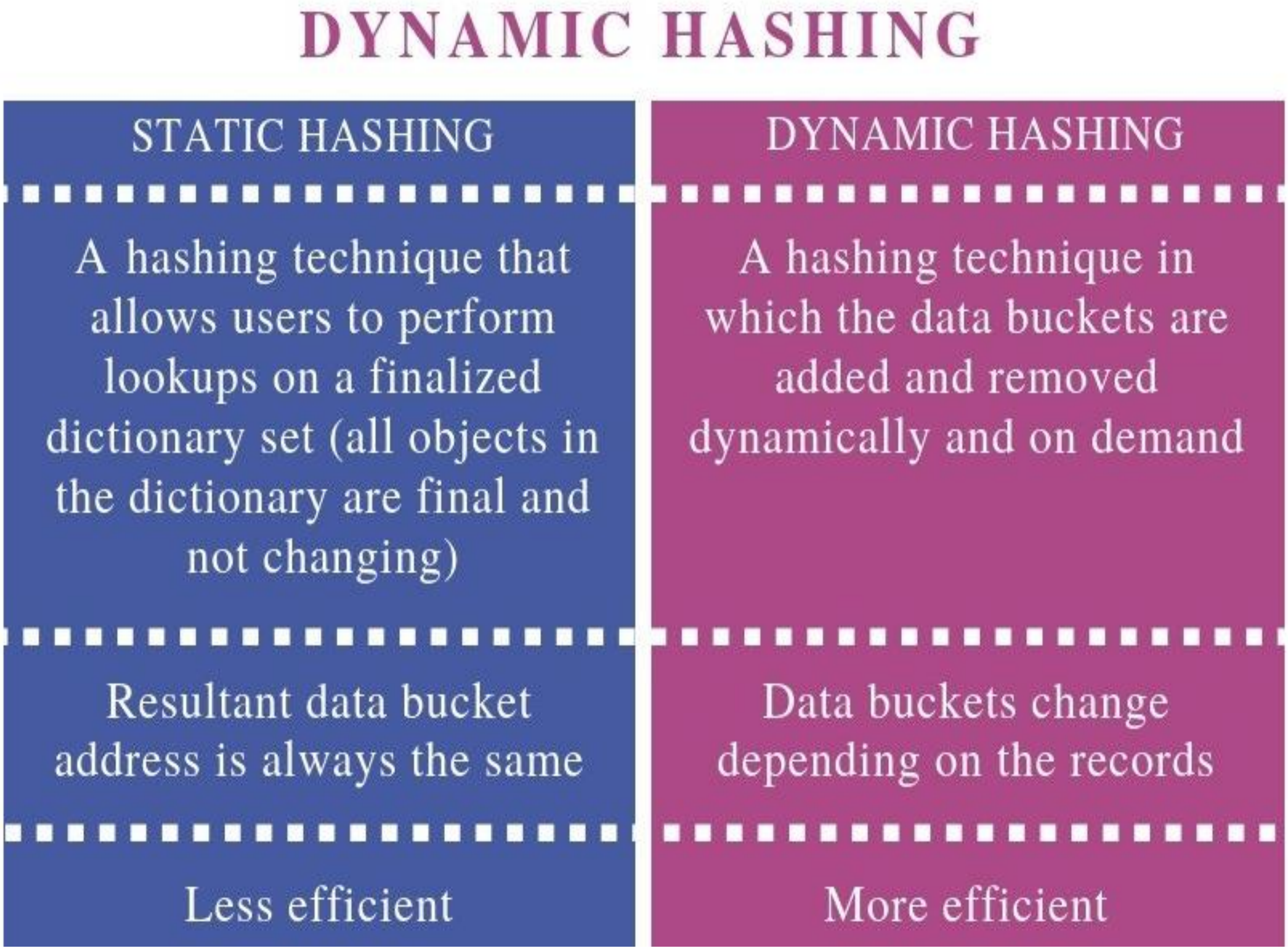
HASHING

Difference between Indexing and Hashing in DBMS :

Indexing	Hashing
It is a technique that allows to quickly retrieve records from database file.	It is a technique that allows to search location of desired data on disk without using index structure.
It is generally used to optimize or increase performance of database simply by minimizing number of disk accesses that are required when a query is processed.	It is generally used to index and retrieve items in database as it is faster to search that specific item using shorter hashed key rather than using its original value.
It offers faster search and retrieval of data to users, helps to reduce table space, makes it possible to quickly retrieve or fetch data, can be used for sorting, etc.	It is faster than searching arrays and lists, provides more flexible and reliable method of data retrieval rather than any other data structure, can be used for comparing two files for quality, etc.
Its main purpose is to provide basis for both rapid random lookups and efficient access of ordered records.	Its main purpose is to use math problem to organize data into easily searchable buckets.
It is not considered best for large databases and its good for small databases.	It is considered best for large databases.
Types of indexing includes ordered indexing, primary indexing, secondary indexing, clustered indexing.	Types of hashing includes static and dynamic hashing.
It uses data reference to hold address of disk block.	It uses mathematical functions known as hash function to calculate direct location of records on disk.
It is important because it protects file and documents of large size business organizations, and optimize performance of database.	It is important because it ensures data integrity of files and messages, takes variable length string or messages and compresses and converts it into fixed length value.



DYNAMIC HASHING

STATIC HASHING	DYNAMIC HASHING
A hashing technique that allows users to perform lookups on a finalized dictionary set (all objects in the dictionary are final and not changing)	A hashing technique in which the data buckets are added and removed dynamically and on demand
Resultant data bucket address is always the same	Data buckets change depending on the records
Less efficient	More efficient