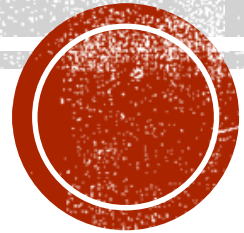
DISCOVER DATABASE



Bin Zhu

Pokémon GO

WeChat

淘宝网
Taobao.com

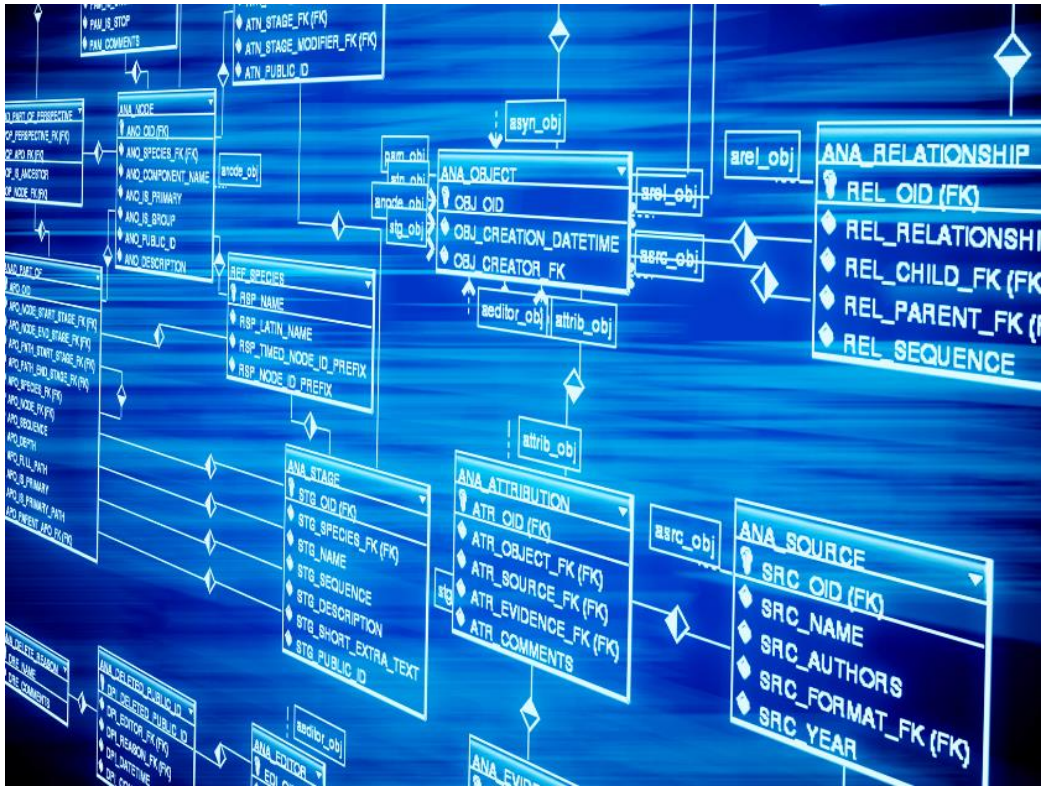
WHAT'S THE DATABASE

A **database** is an organized collection of data. It is the collection of schemas, tables, queries, reports, views, and other objects. The data are typically organized to model aspects of reality in a way that supports processes requiring information

(copied from Wikipedia)

THE OBJECTS IN DATABASE

- Tables
- View
- Materialized View
- Index
- Procedure
- Sequence
- Trigger
- DB Link



TABLE

A table is a collection of related data held in a structured format within a database. It consists of columns, and rows.

Columns stores a specific data type

Row → Or record

Emp No	Name	Age	Department	Salary
001	Alex S	26	Store	5000
002	Golith K	32	Marketing	5600
003	Rabin R	31	Marketing	5600
004	Jons	26	Security	5100

VIEW

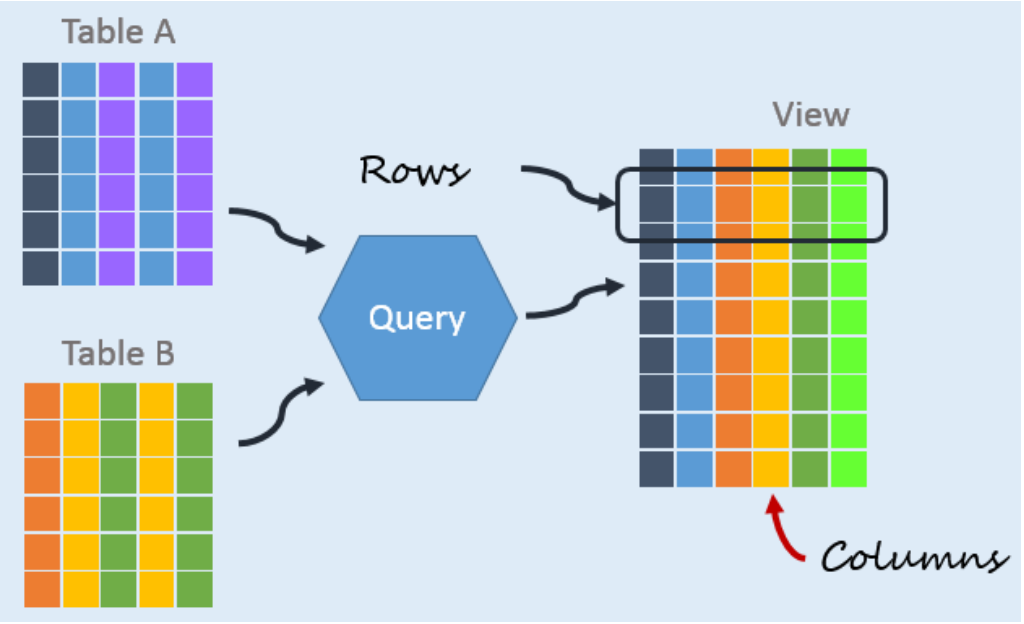
A **view** is a virtual table based on the result-set of an SQL statement.

A **view** contains rows and columns, just like a real table.

The fields in a **view** are fields from one or more real tables in the **database**.

Sample:

```
CREATE VIEW view_dept_201
AS
SELECT emp_id, name, hire_date
FROM gdb.employees
WHERE department = 201;
```



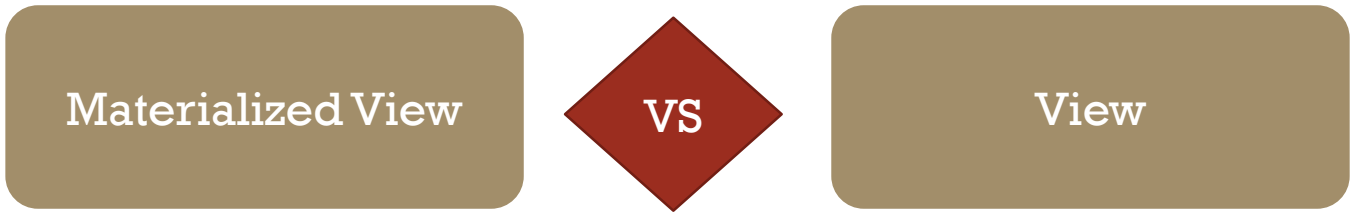
MATERIALIZED VIEW

A **materialized view** is a database object that contains the results of a query

Sample:

Create materialized view myView as select * from myTab;

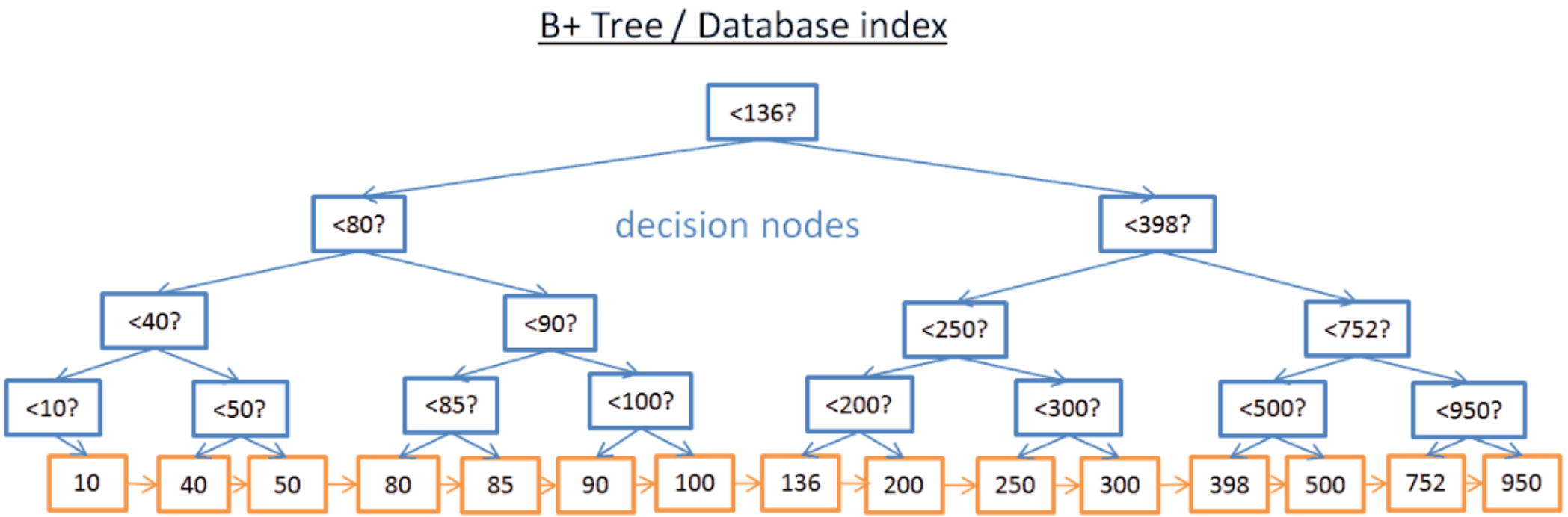
Refresh materialized view myView;



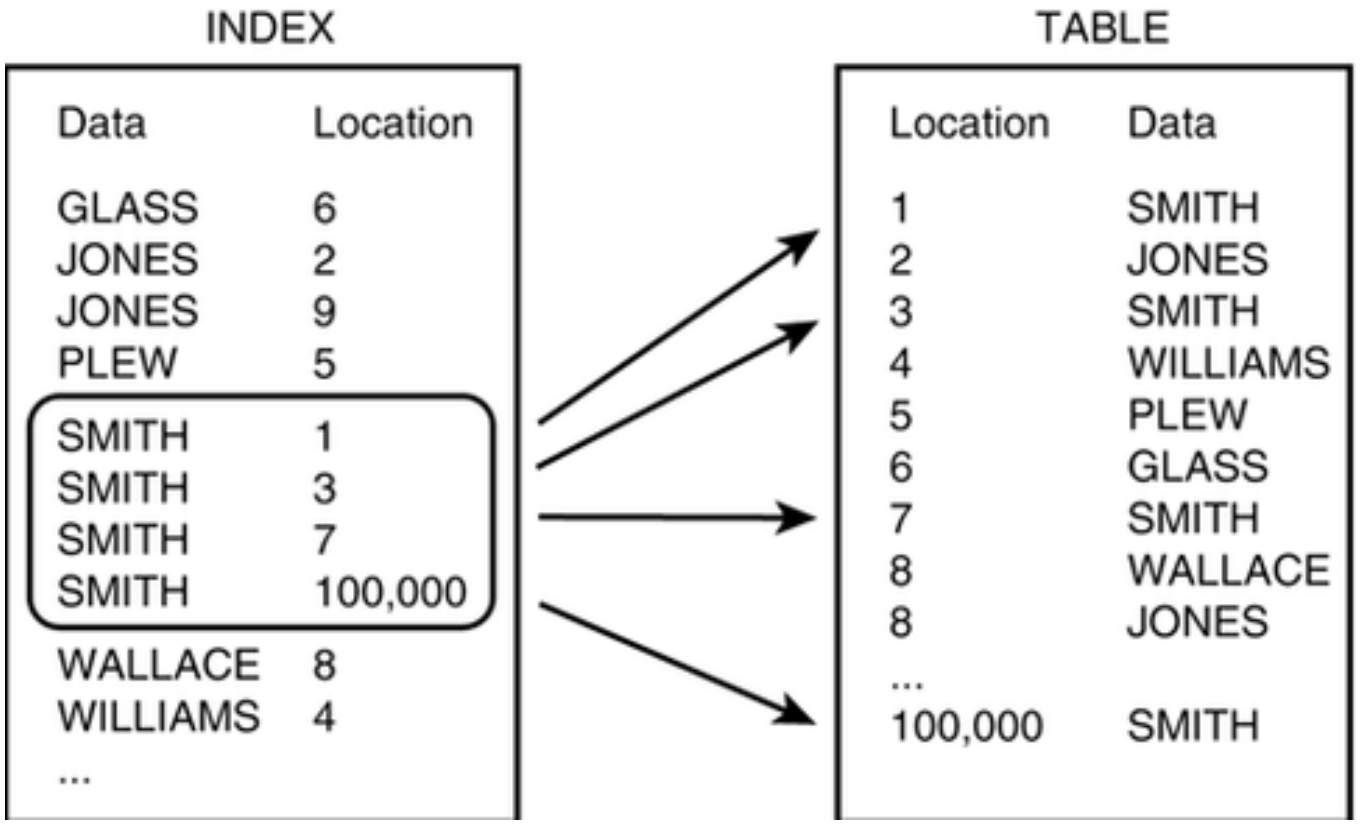
INDEX

- Indexes are **Database Object** that you can create to improve the performance of query.
- Indexes can be created **explicitly** or **automatically**.
- It is used to **retrieve data faster**.
- You can create indexes on one or more columns of a table.
- You can create more than one index on one table.
- User can't see indexes.

INDEX – HOW IT WORKS

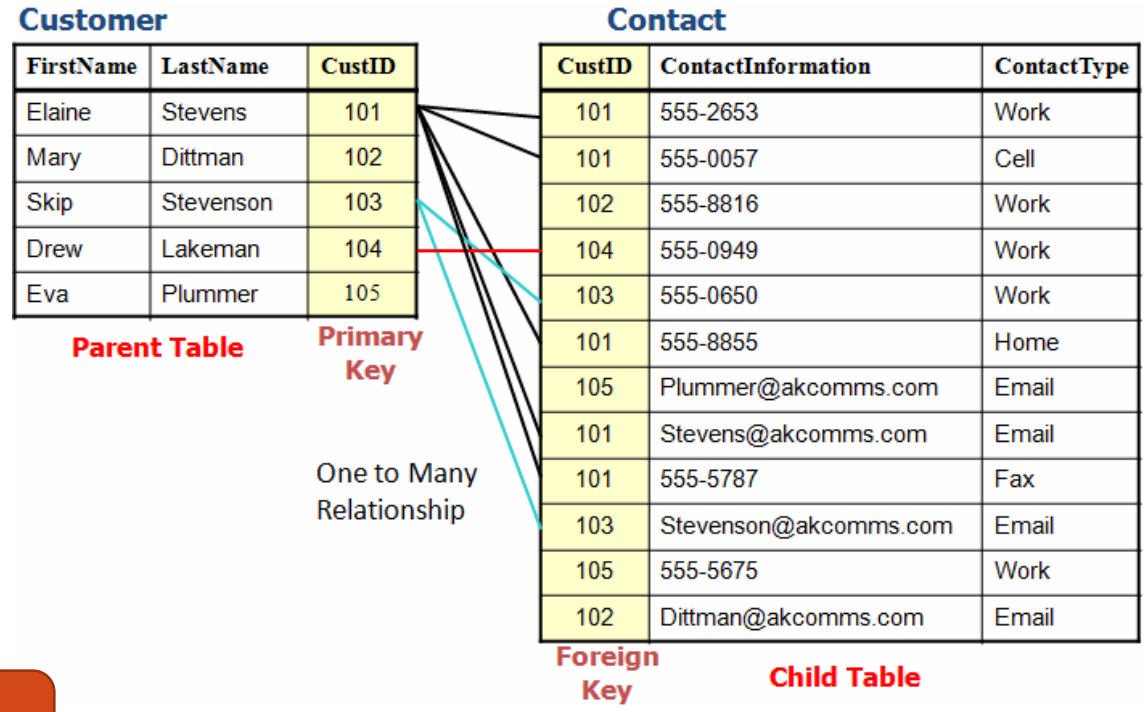


INDEX – HOW IT WORKS



INDEX - TYPE

- Primary Key
- Unique Index
- Foreign key
- Index



Primary Key ≠ Unique Index

PROCEDURE

A **stored procedure** is a set of Structured Query Language (SQL) statements with an assigned name that's **stored** in the **database** in compiled form so that it can be shared by a number of programs.

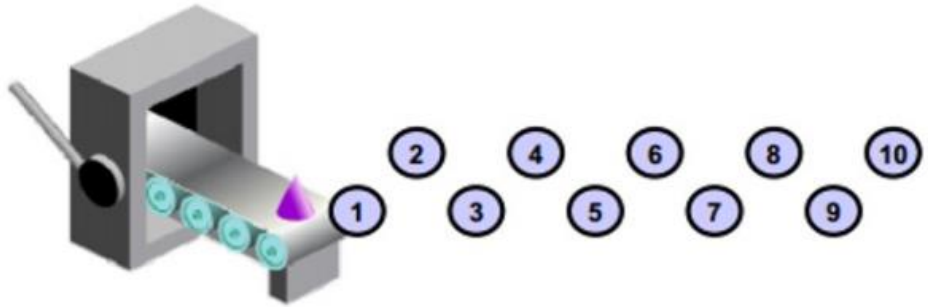
Sample

```
ALTER PROCEDURE <ProcedureName>
    -- Add the parameters for the stored procedure here
    <@Parameter1> <Datatype_For_Parameter1> = <Default_Value>,
    <@Parameter1> <Datatype_For_Parameter1> = <Default_Value>
AS
BEGIN
    -- Insert statements for procedure here
    SELECT * FROM TableName
END
```

SEQUENCE

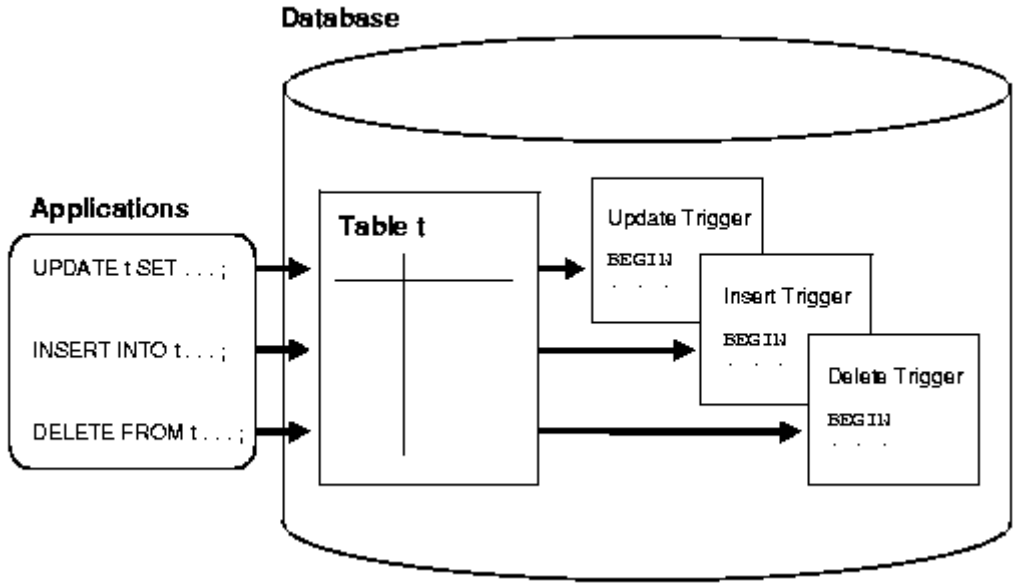
A sequence:

- Can automatically generate unique numbers
- Is a shareable object
- Can be used to create a primary key value
- Replaces application code
- Speeds up the efficiency of accessing sequence values when cached in memory



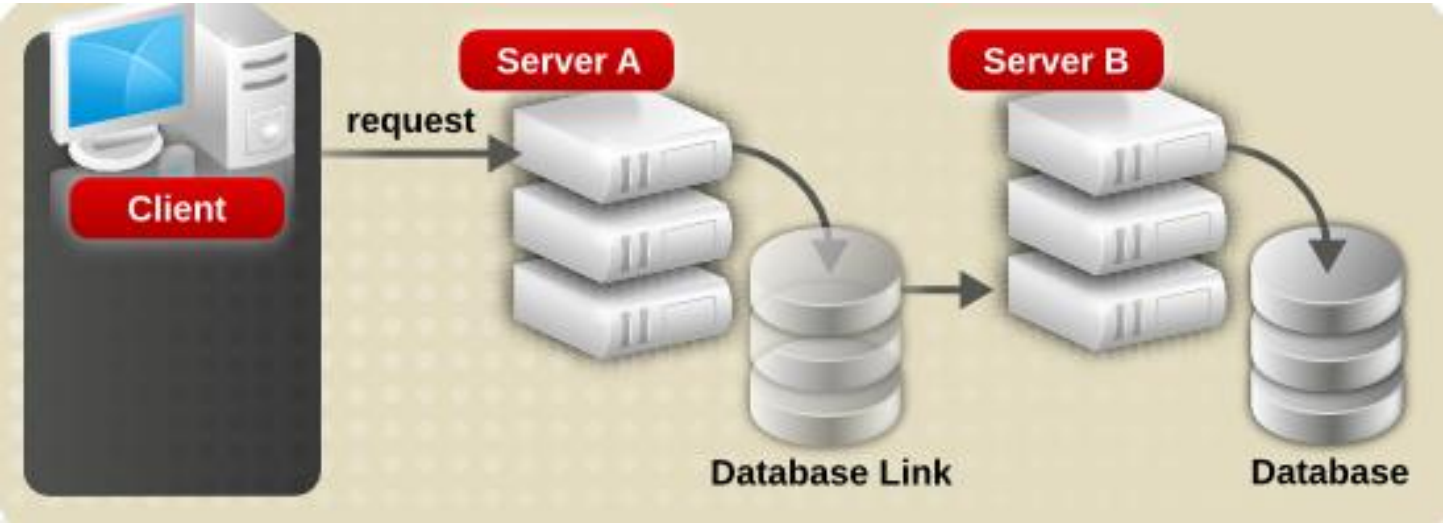
TRIGGER

A **trigger** is a special kind of stored procedure that automatically executes when an event occurs in the **database** server. DML **triggers** execute when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view



DB LINK

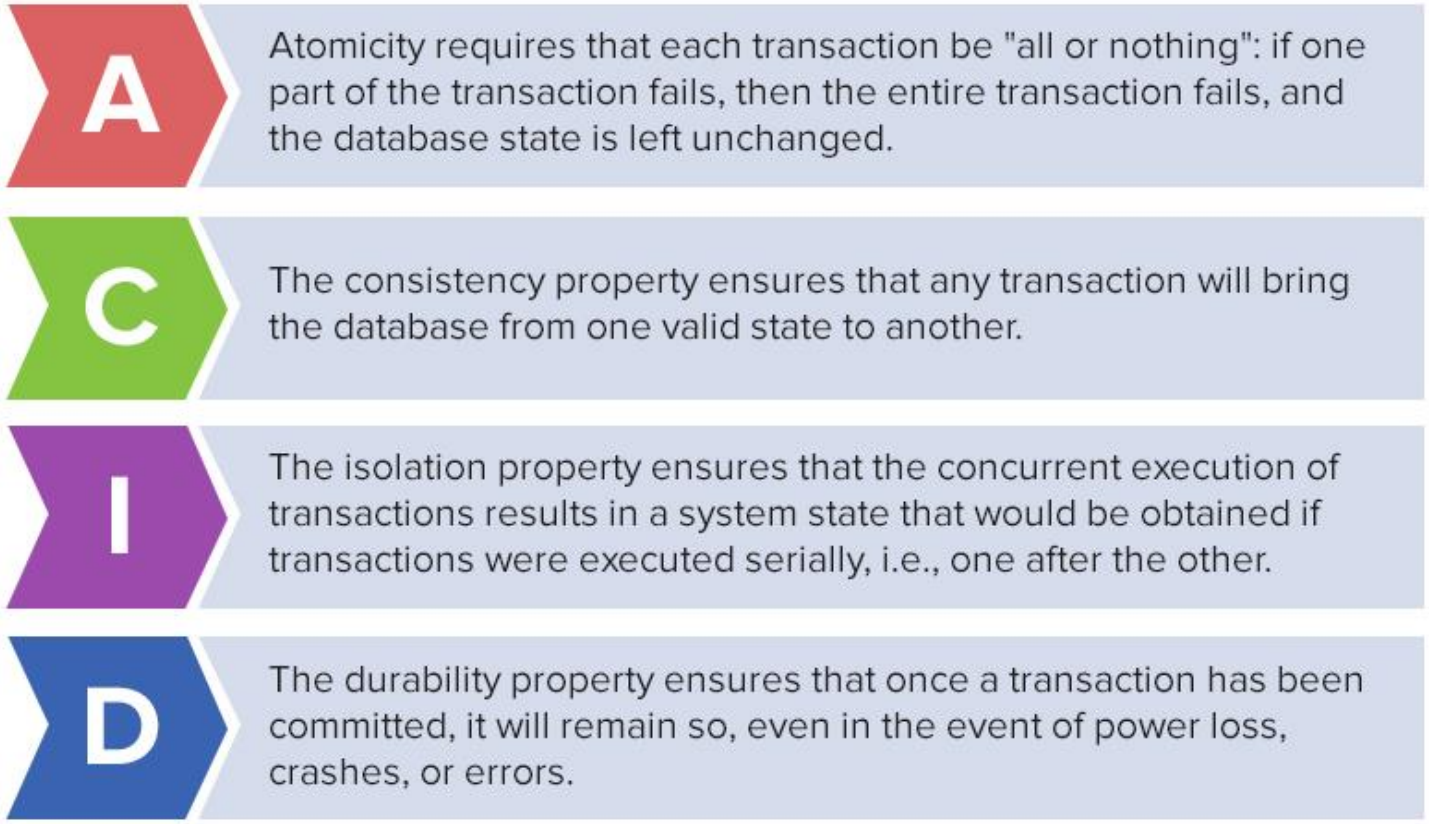
Dblink is a module which supports connections to other databases from within a database session



DATABASE TRANSACTION

- ACID
- Transaction
- Distributed Transaction
- Two Phase Commit
- Real Commit Diagram

DATABASE BASIC PROPERTIES



TRANSACTION

A transaction is a unit of work that you want to treat as "a whole". It has to either happen in full, or not at all.

Commit

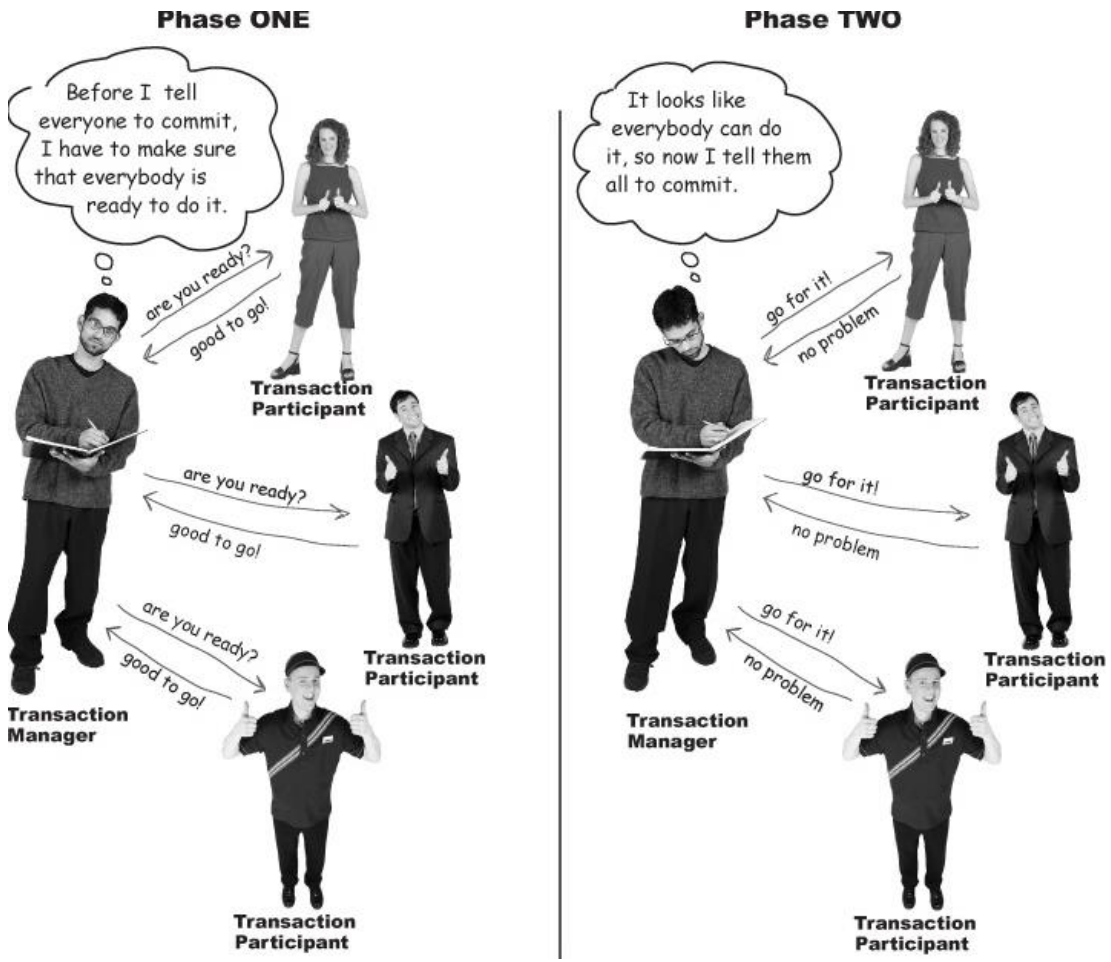
OR

Rollback

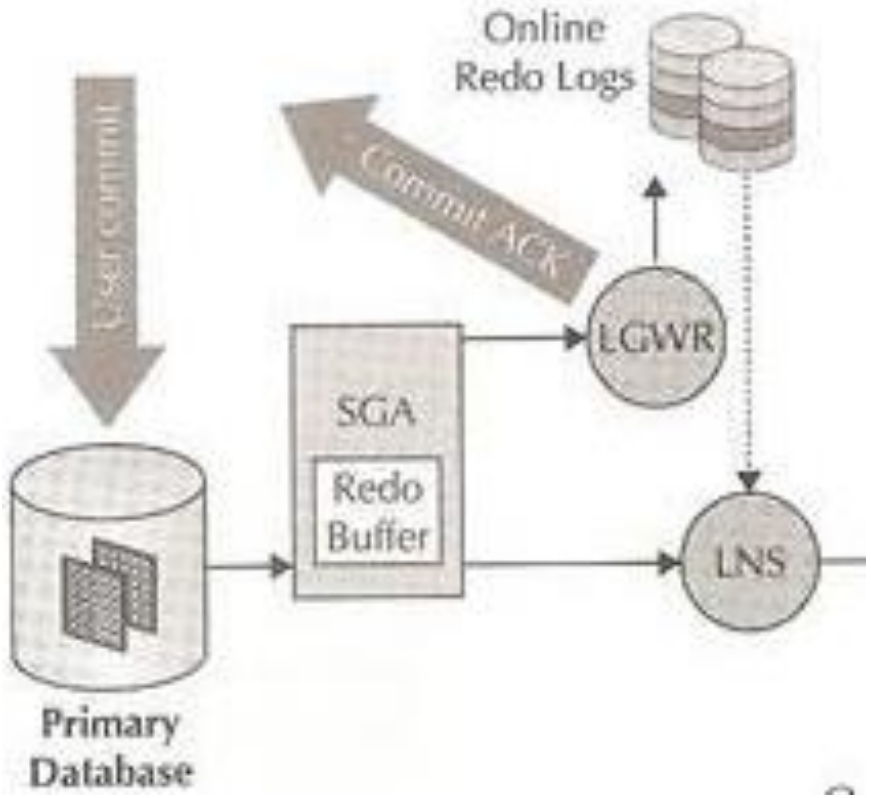
DISTRIBUTED TRANSACTION

A **distributed transaction** is a database **transaction** in which two or more network hosts are involved

TWO PHASE COMMIT

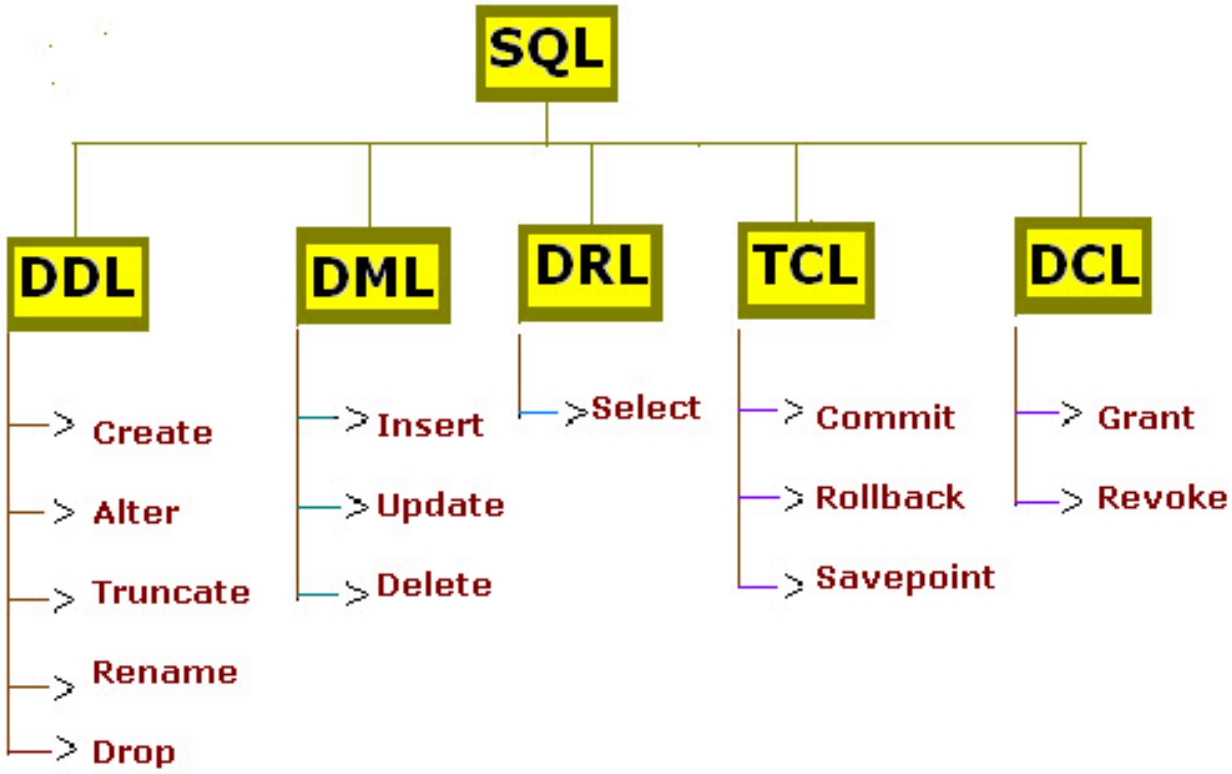


DATABASE REAL COMMIT DIAGRAM



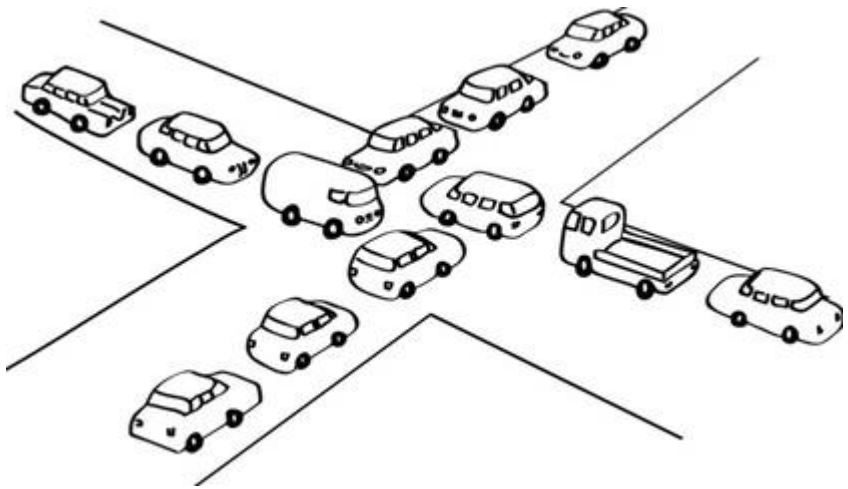
DATABASE SQL OPERATION

- DDL:
  Data Definition Language
- DML:
  Data Manipulation Language
- DRL:
  Data Retrieval Language
- TCL:
  Transaction Control Language
- DCL:
  Data Control Language



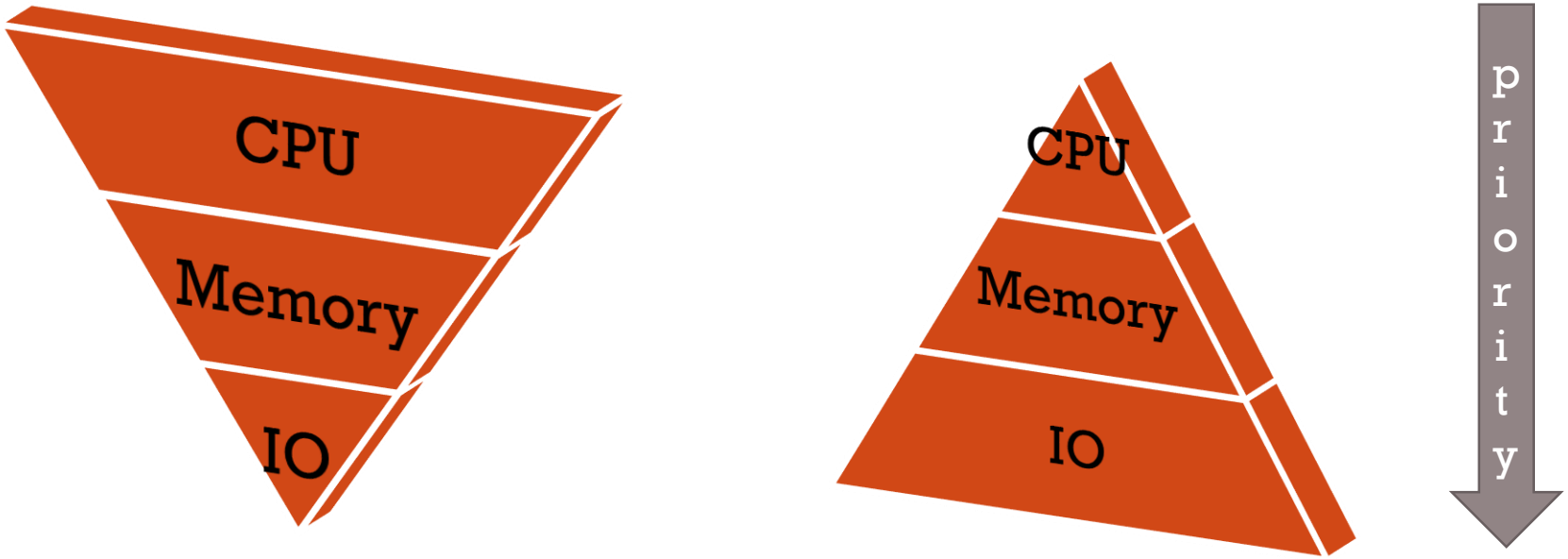
DATABASE DEADLOCK

A **deadlock** is a situation in which two or more transactions are waiting for one another to give up locks.



1, Select for Update
2, Insert
3, Update
4, Delete

DATABASE PERFORMANCE

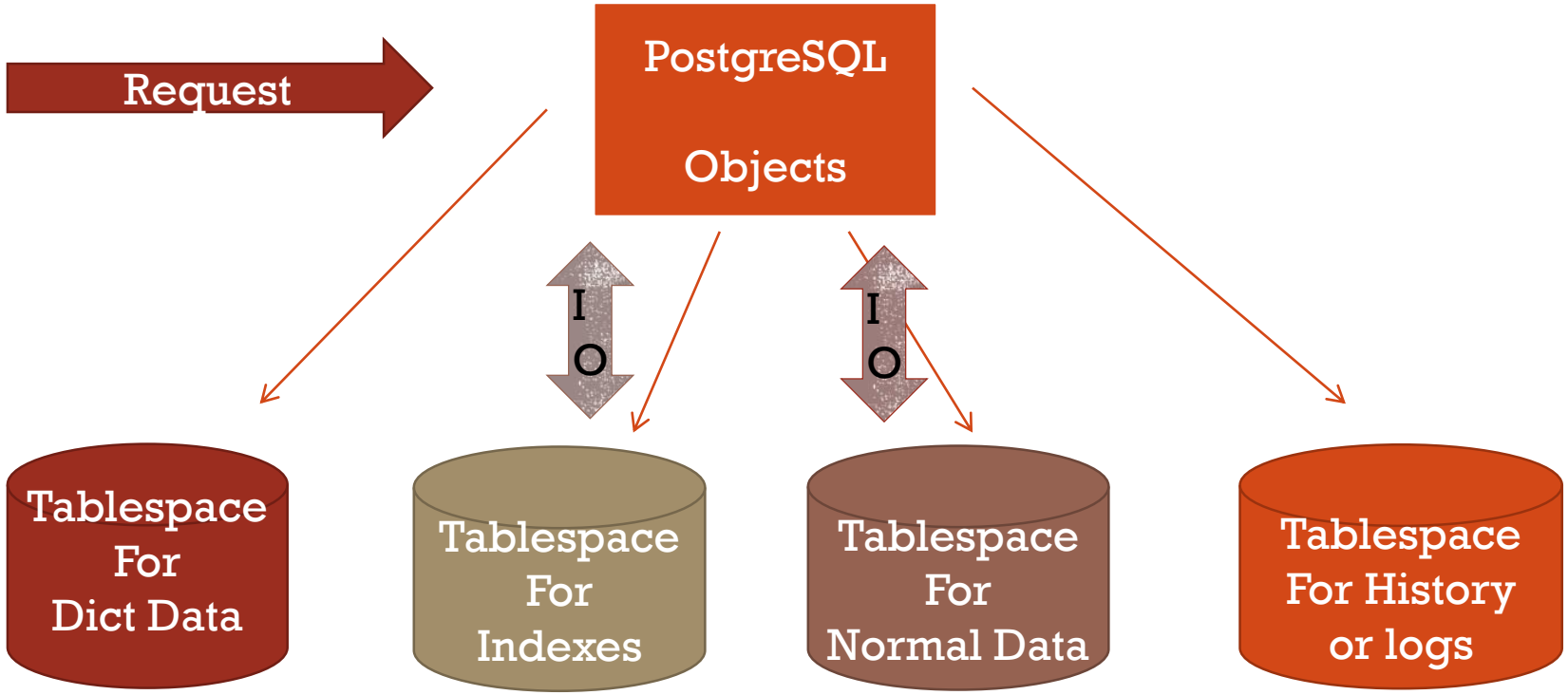


IO TUNING PERFORMANCE

- Tablespace
  A **tablespace** is a storage location where the actual data underlying database objects can be kept. It provides a layer of abstraction between physical and logical data, and serves to allocate storage for all DBMS managed segments.

- Fragmentation

TABLESPACE


Request
PostgreSQL
Objects
I
O
I
O
Tablespace
For
Dict Data
Tablespace
For
Indexes
Tablespace
For
Normal Data
Tablespace
For History
or logs

FRAGMENTATION

- Table Fragmentation
- Index Fragmentation

- How to Perform the fragmentation
  - Vacuum Database
  - Rebuild Index
  - Rebuild Table

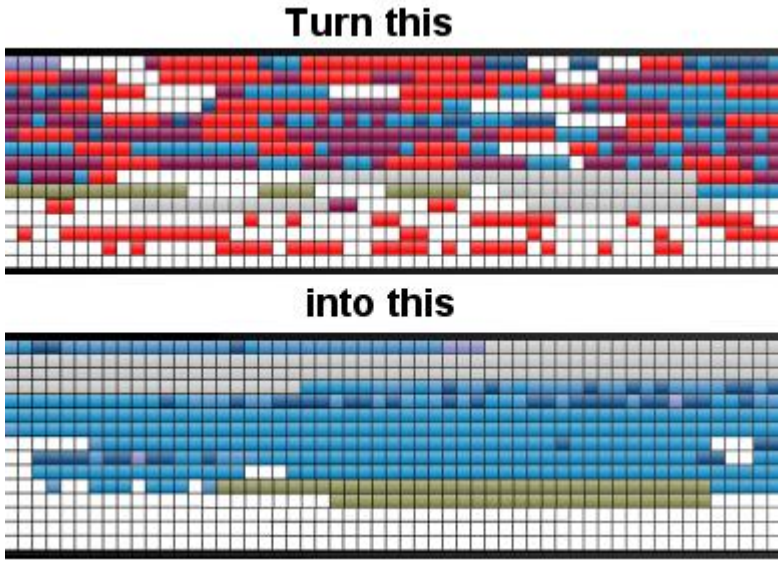


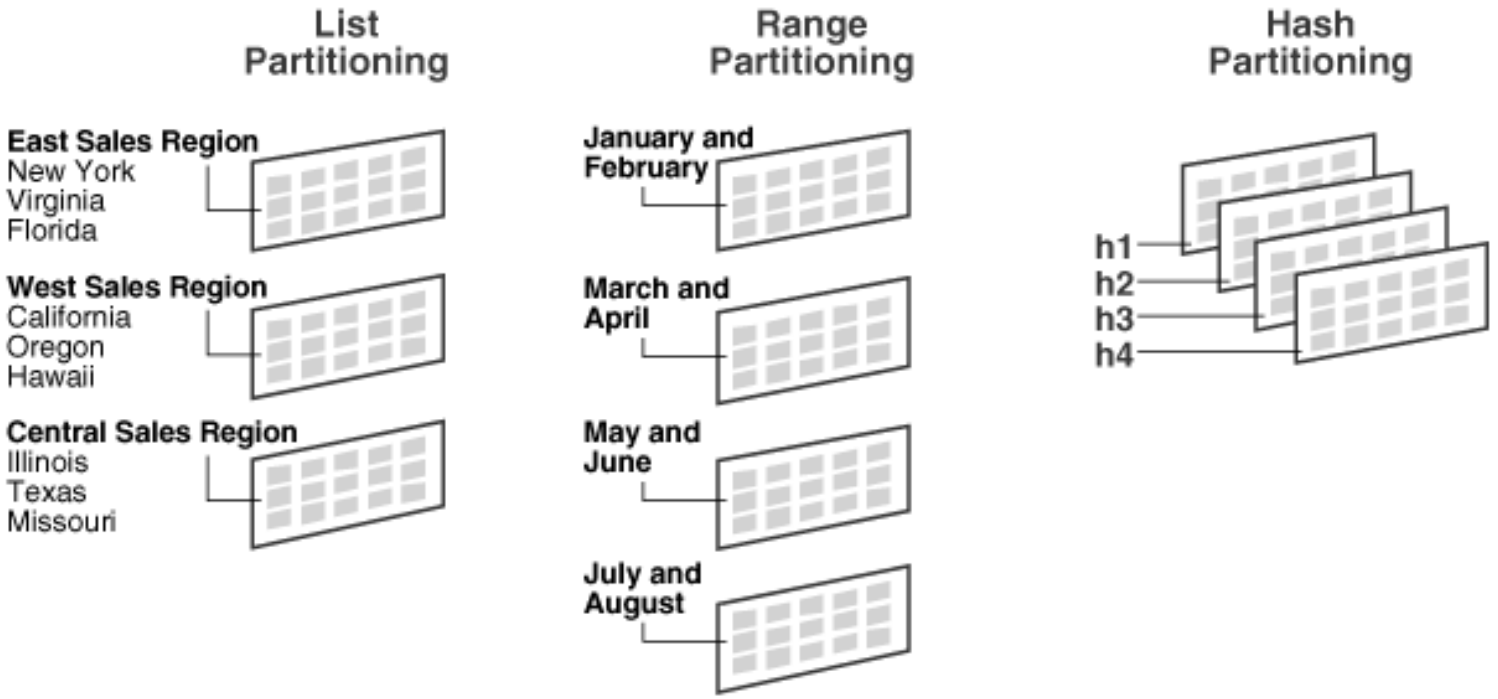
TABLE PARTITION

A **partition** is a division of a logical database or its constituent elements into distinct independent parts. Database partitioning is normally done for manageability, performance or availability reasons, as for load balancing.

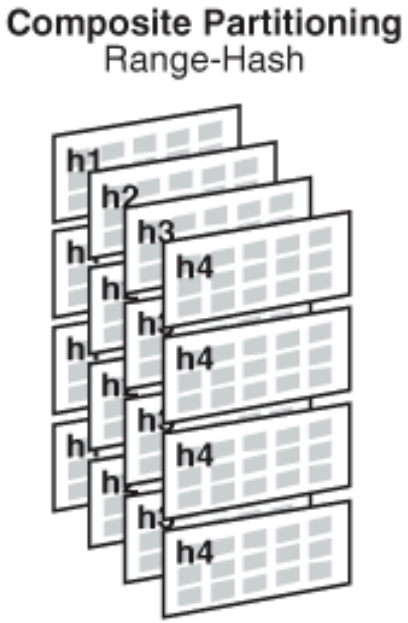
Partitioning criteria

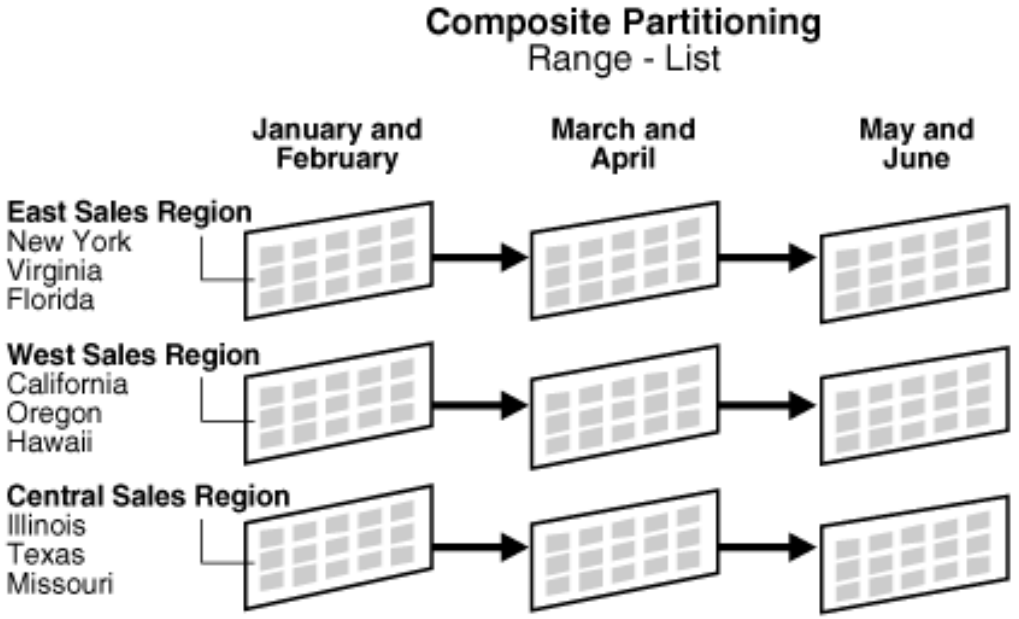
- Range partitioning
- List partitioning
- Hash partitioning
- Composite partitioning

RANGE, LIST, HASH PARTITIONING

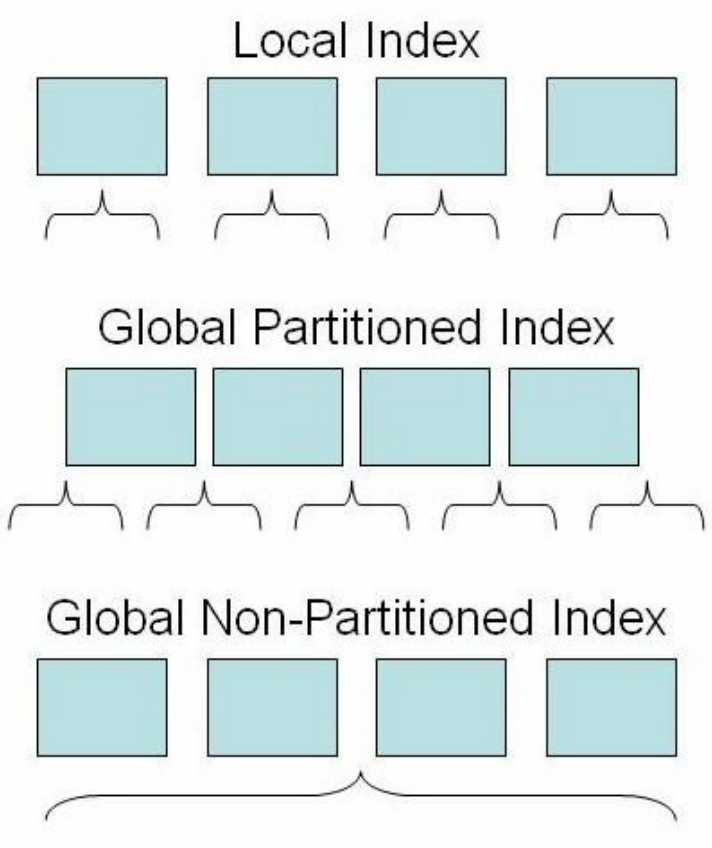


COMPOSITE PARTITIONING



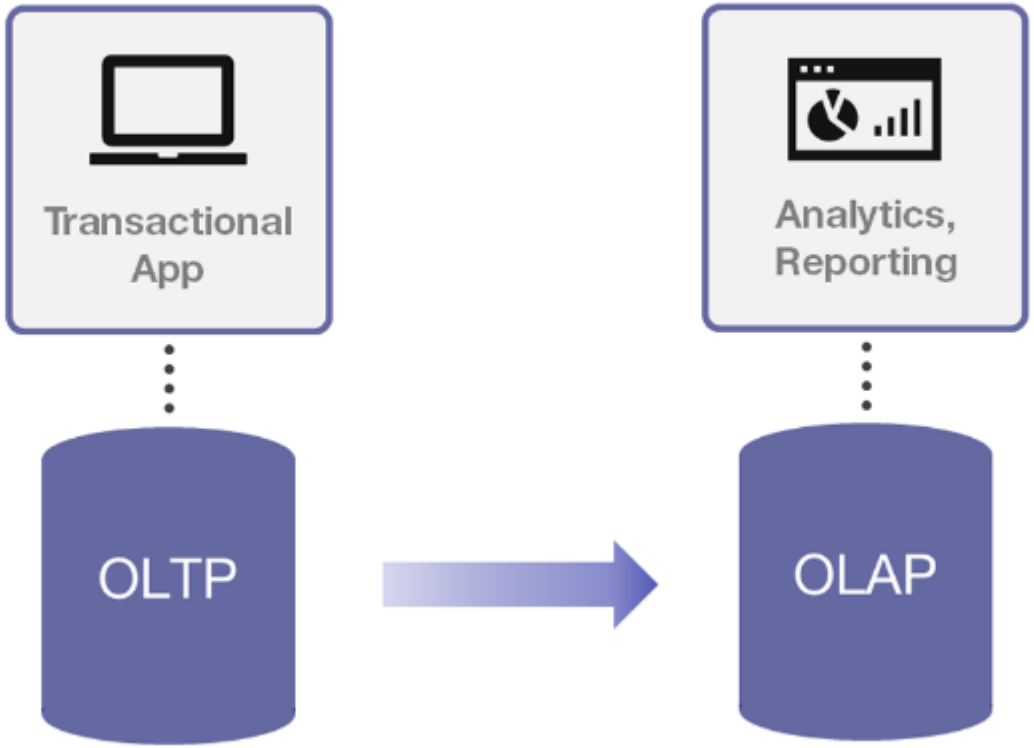


PARTITION INDEX



OLTP VS OLAP

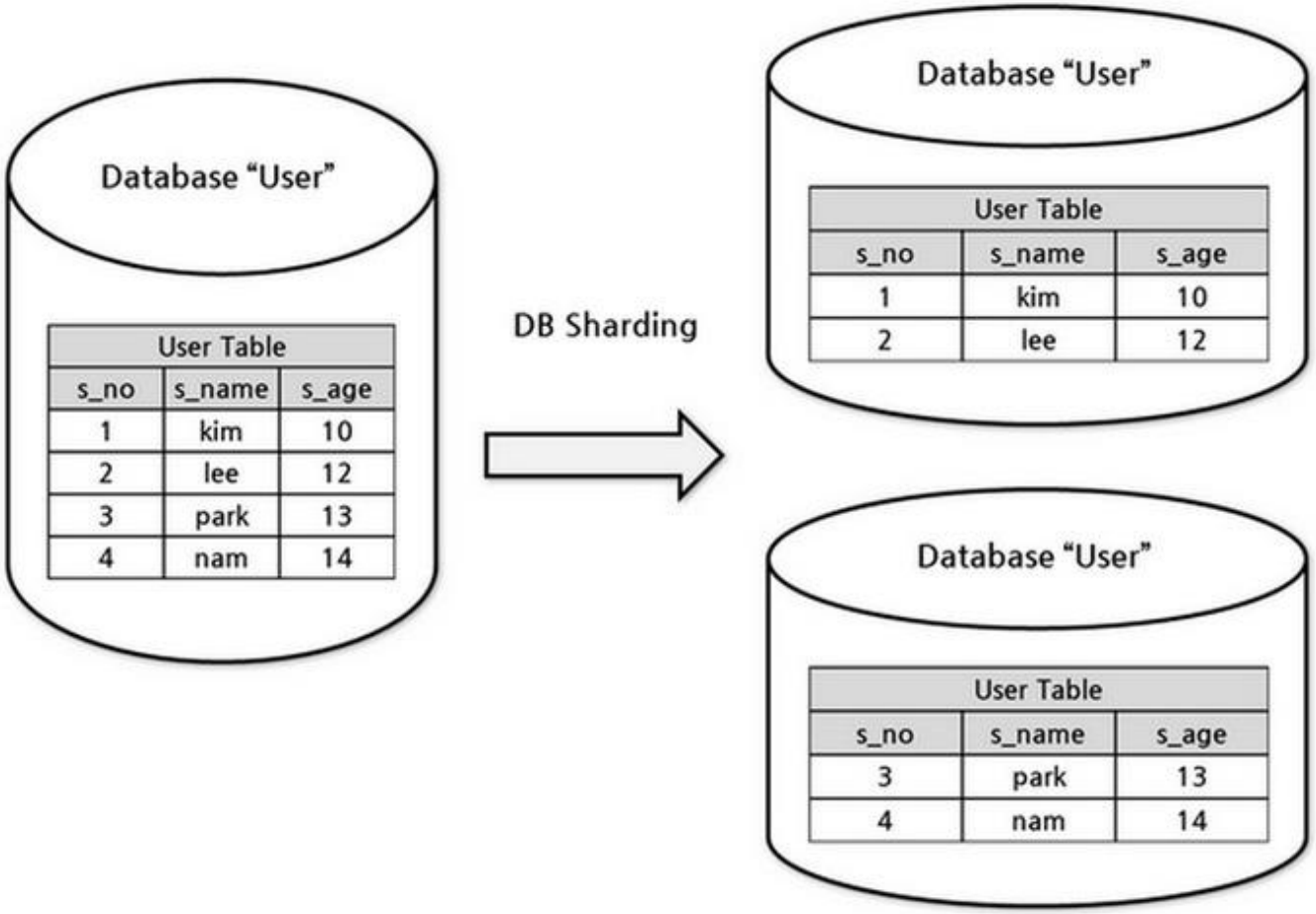
- High volume of transactions
- Fast processing
- Normalized data
- Many tables
- *"Who bought X?"*



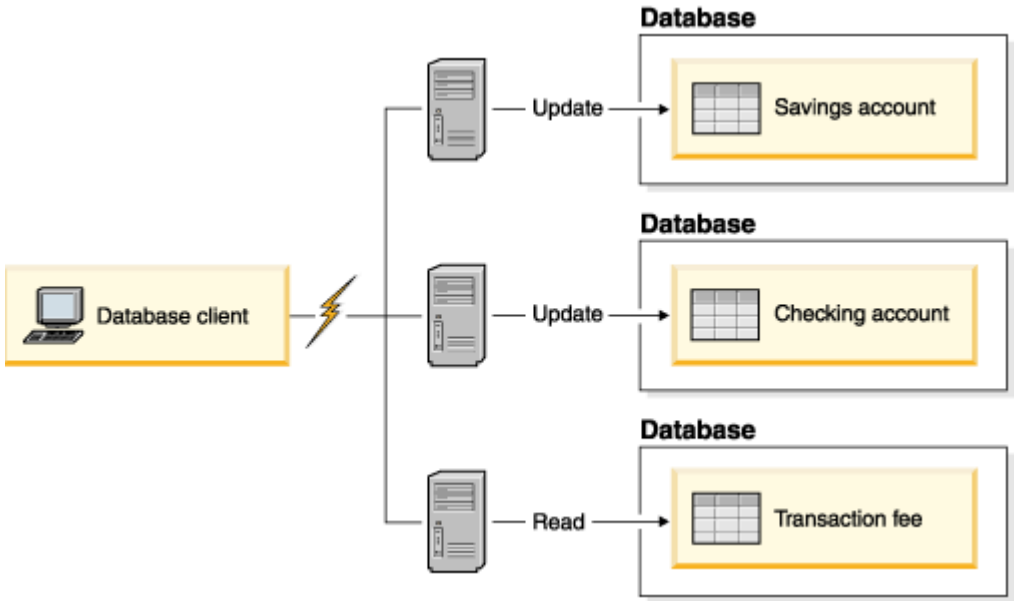
- High volume of data
- Slow queries
- Denormalized data
- Fewer tables
- *"How many people bought X?"*

DATABASE SHARDING

Database Sharding is a highly scalable approach for improving the throughput and overall performance of high-transaction, large database-centric business applications



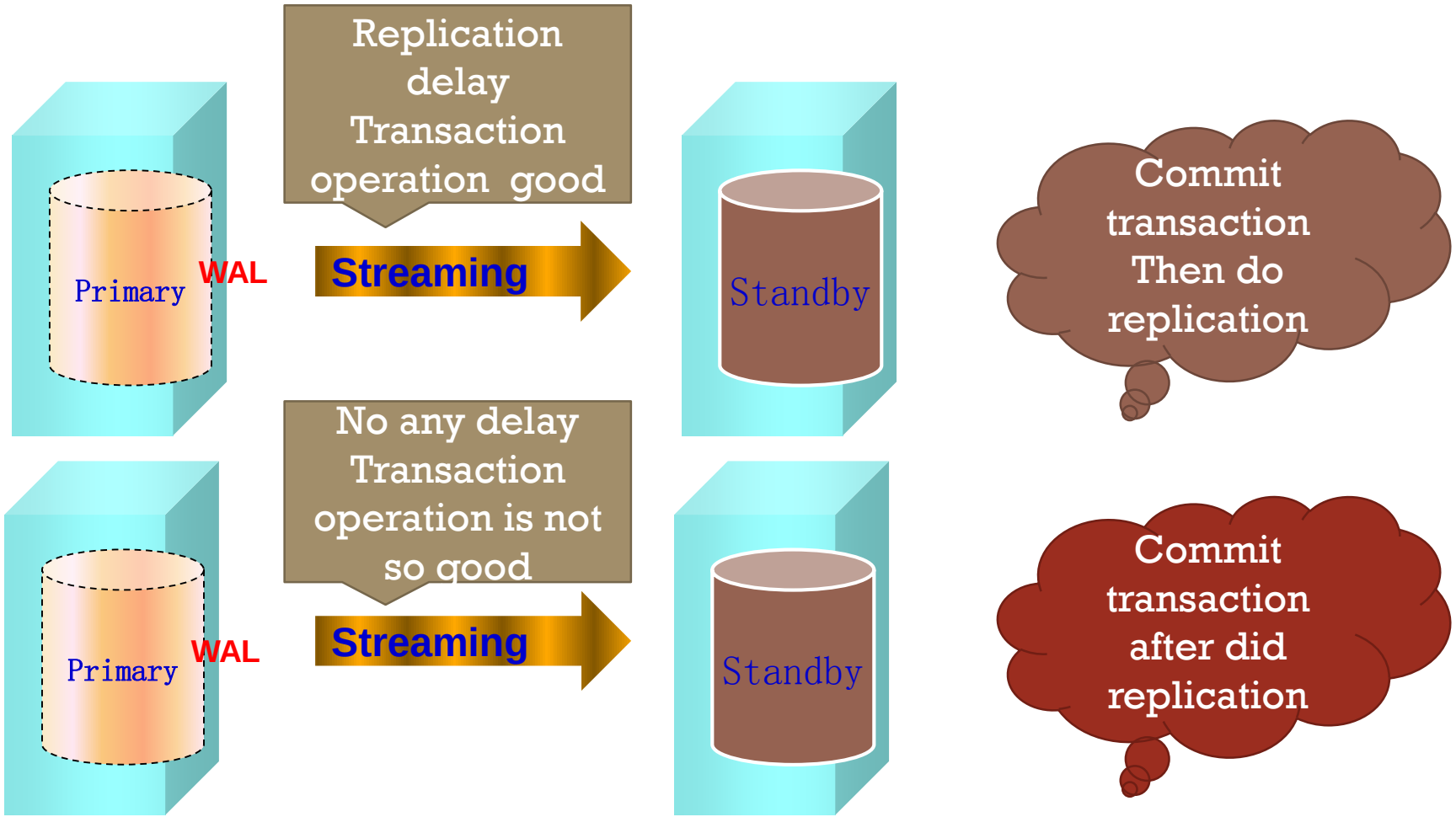
DATABASE SHARDING



DATABASE REPLICATION

Database replication is the frequent electronic copying data from a database in one computer or server to a database in another so that all users share the same level of information.

ASYNC REPLICATION VS SYNC



DATABASE CLUSTER

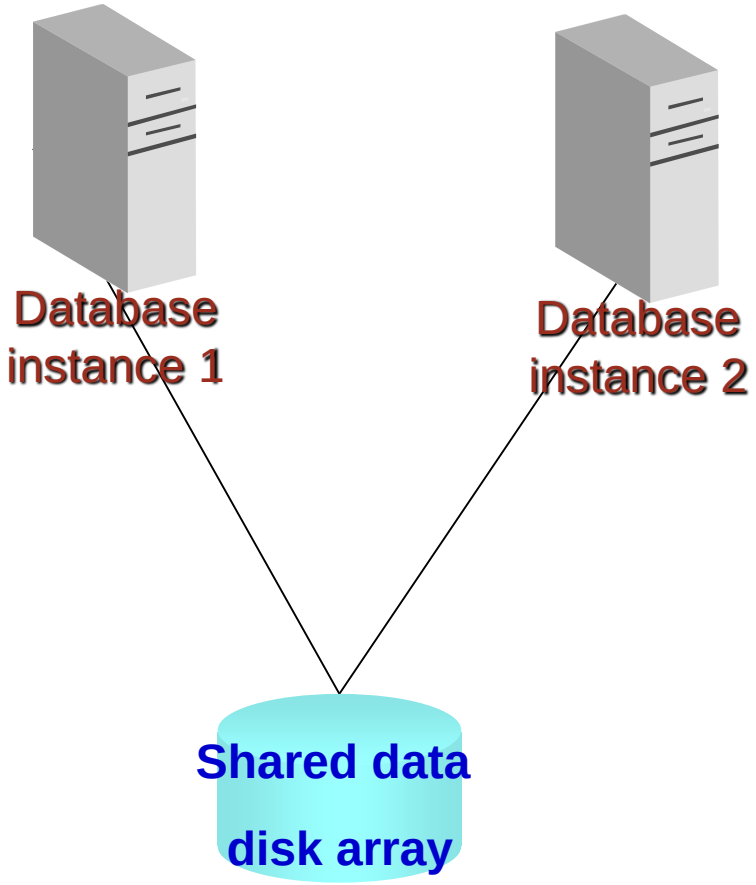
Clustering offers two major advantages, especially in high-volume database environments:

- Fault tolerance: Because there is more than one server or instance for users to connect to, clustering offers an alternative, in the event of individual server failure.
- Load balancing: The clustering feature is usually set up to allow users to be automatically allocated to the server with the least load.

DATABASE CLUSTER TYPE

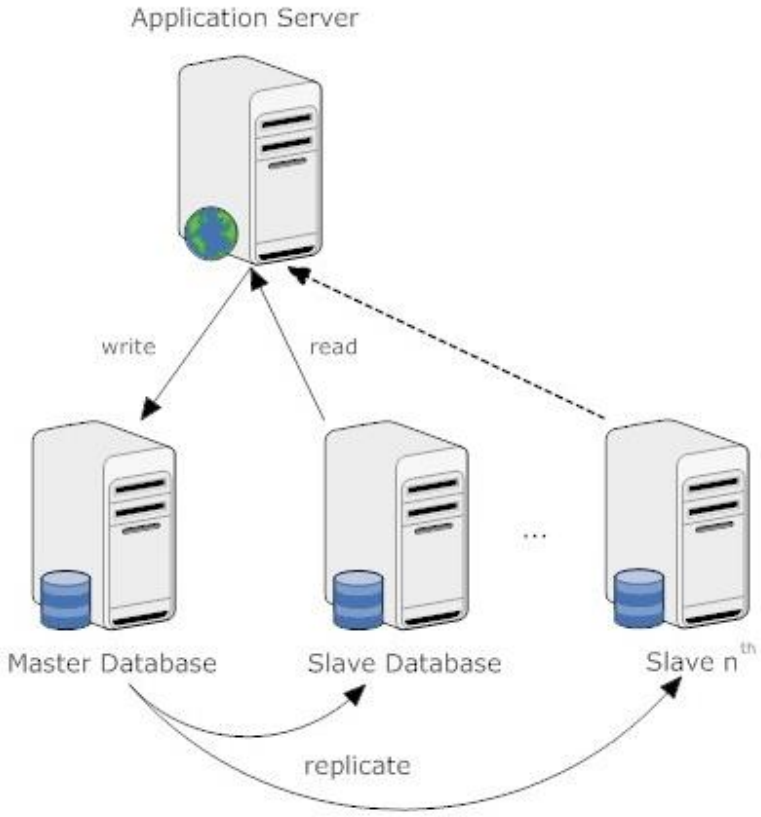
- Shared Data Array
- Master – Slave(s)
- Master - Master

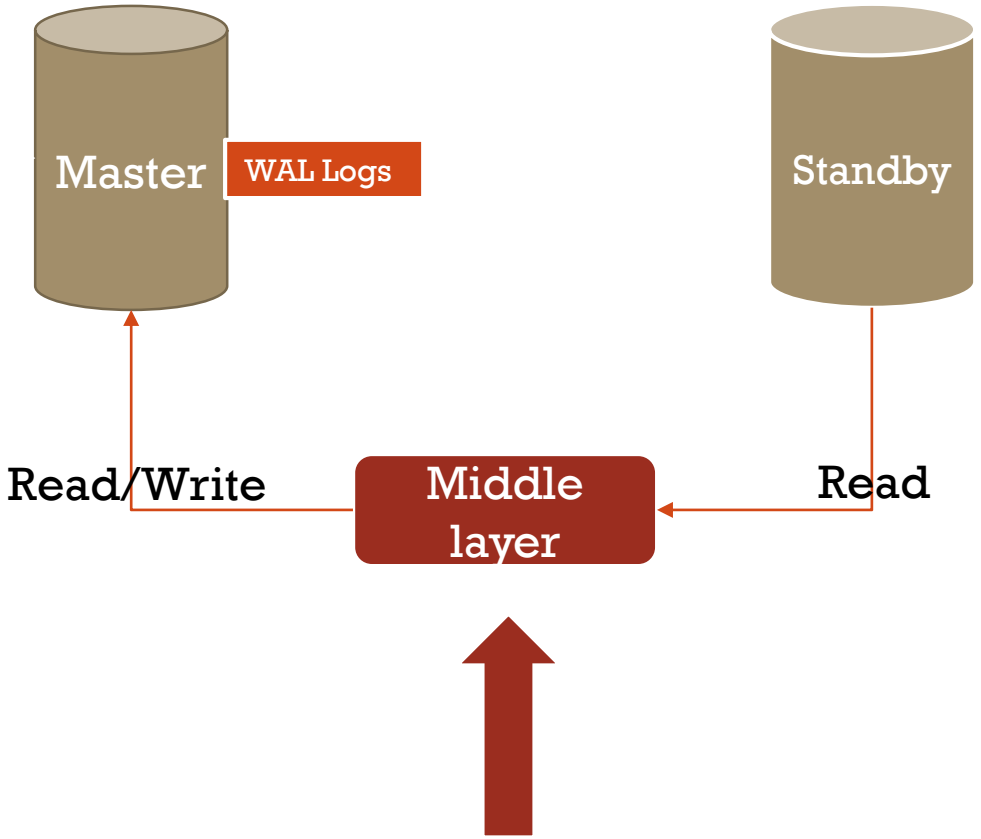
SHARED DATA ARRAY



- Positive
  Shared data disk array solution can avoid take long time to synchronized the data between the different database. When the postgres instance1 failed, you just need to assign all the clients to connect to instance2 and all the data will not lost.
- Negative
  Because all the data is stored in the same disks, if the disks are failed, the data will lost.

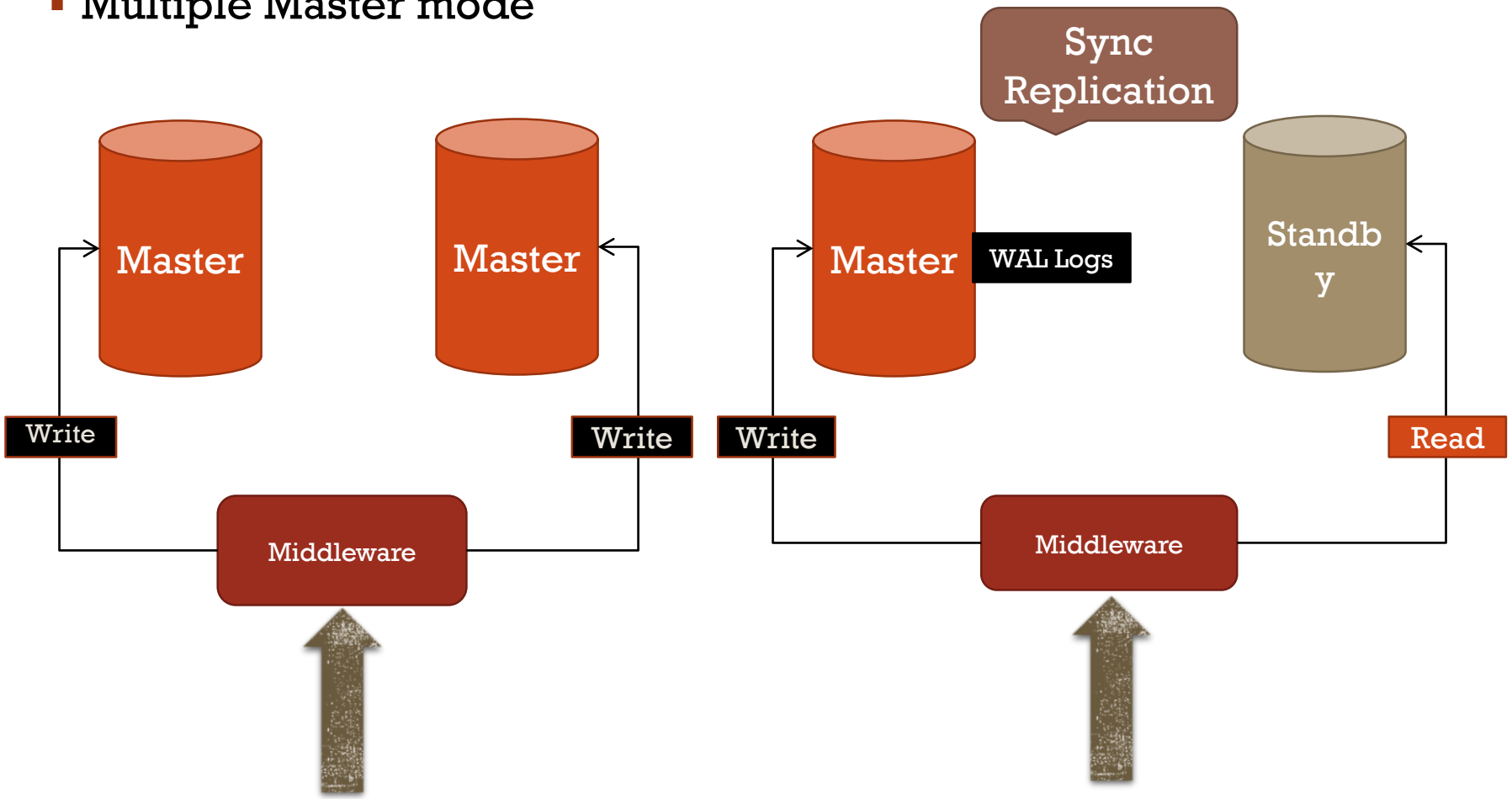
MASTER – SLAVE(S)





MASTER – MASTER

- Multiple Master mode

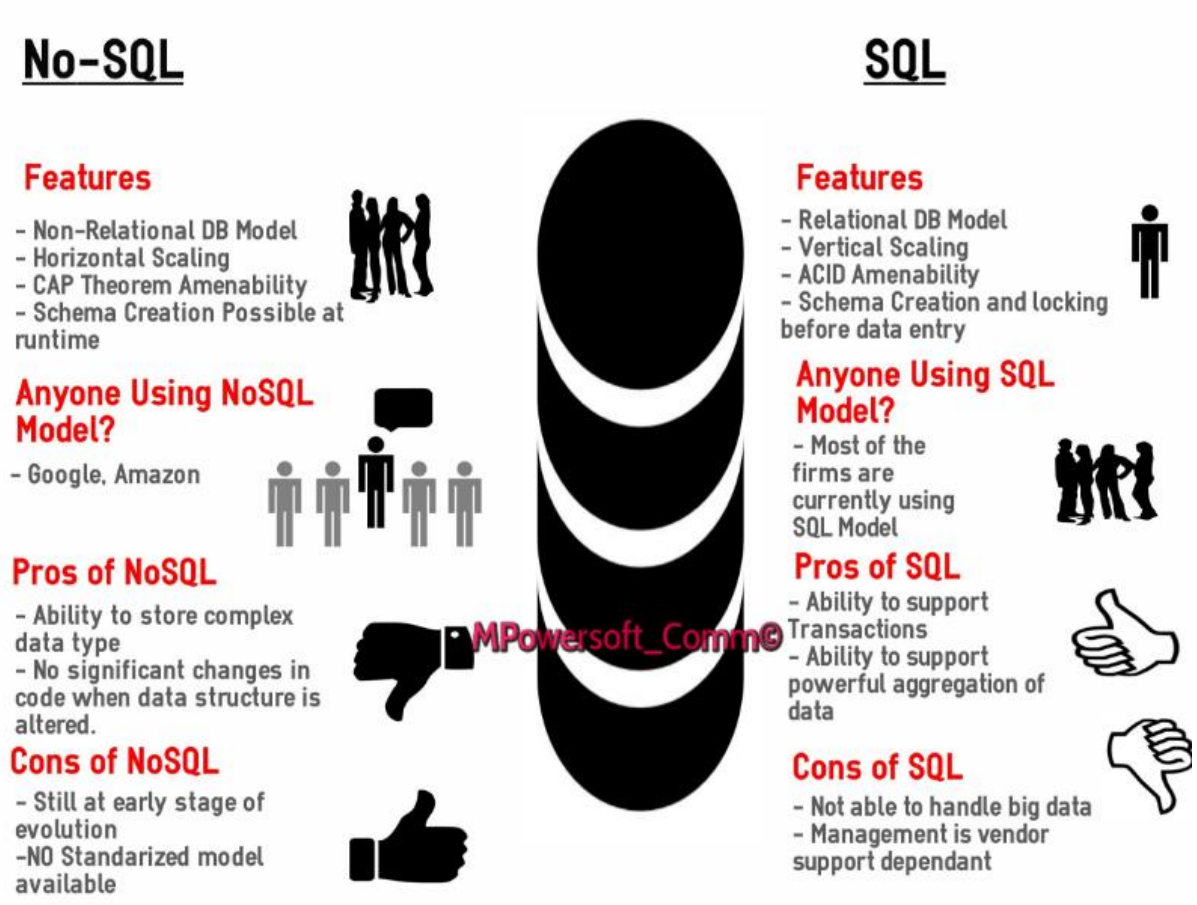


CLOUD DATABASE

Amazon RDS Database Engines

CLOUD DATABASE VIDEO (COPIED FROM AWS)

RDBM VS NOSQL



VIDEO (DYNAMO) (COPIED FROM AWS)

NEW SQL

NewSQL is a class of modern relational database management systems that seek to provide the same scalable performance of NoSQL systems for online transaction processing (OLTP) read-write workloads while still maintaining the ACID guarantees of a traditional database system

-- Wikipedia

VIDEO (GOOGLE SPANNER) (COPIED FROM GOOGLE)

CATALOGUE

- What's the database
- The objects in database
  - Tables
  - View
  - Materialized View
  - Index
  - Procedure
  - Sequence
  - Trigger
  - DB Link
- Database Transaction
  - ACID
  - Transaction
  - Distributed Transaction
  - Two phase commit
  - Database Real Commit Diagram
- Database operation
  - DDL
  - DML
  - DRL
  - TCL
  - DCL
- Database deadlock
- Database performance
  - Tuning performance
  - Tablespace
  - Fragmentation
- Table partition
  - Range
  - List
  - Hash
  - Composite
  - Partition index
    - Global
    - Local
- OLTP VS OLAP
- DB sharding
- DB replication
  - Async
  - Sync
- DB Cluster
  - Shared Disk Array
  - Master – Slave(s)
  - Multiple Master
- Cloud Database
- RDBM VS NoSQL
- NewSQL


Q&A
WHAT
WHERE
WHEN
WHO
WHICH
HOW