PostgreSQL Architecture

Postmaster:

* Postmaster is the first process which gets started in PostgreSQL
* Postmaster acts as supervisor process, whose job is to monitor, start, restart some processes if they die.
* Postmaster acts a listener and receive new connection request from the client.
* Postmaster is responsible for Authentication and Authorization of all incoming request.
* Postmaster spawns a new process call Postgres for each new connection.

Utility Process:

* Bgwriter\Writer: Periodically writes the dirty buffer to a data file.
* Wal Writer: Write the WAL buffer to the WAL file.
* Checkpointer: Checkpoint is invoked every 5 minute(default) or when max\_wal\_size value is exceeded. The check pointer syncs all the buffers from the shared buffer area to the data files.
* Auto vacuum: Responsible to carry vacuum operations on bloated tables. (If Enabled).
* Statscollector: Responsible for collection and reporting of information about server activity then update the information to optimizer dictionary((pg\_catalog).
* Logwriter\Logger: Write the error message to the log file.
* Archiver (Optional): When in Archive.log mode, copy the WAL file to the specified directory.

Memory Segments of PostgreSQL:

* Shared Buffers
* Wal Buffers
* Clog Buffers
* Work Memory
* Maintenance Work Memory
* Temp Buffers

Shared Buffer:

* User cannot access the datafile directly to read or write any data.
* Any select, insert, update or delete to the data is done via shared buffer area.
* The data that is written or modified in this location is called “Dirty data“.
* Dirty data is written to the data files located in physical disk through background writer process.
* Shared Buffers are controlled by parameter named: shared\_buffer located in postgresql.conf file.

Wal Buffer:

* Write ahead logs buffer is also called as “Transaction log Buffers”.
* WAL data is the metadata information about changes to the actual data, and is sufficient to reconstruct actual data during database recovery operations.
* WAL data is written to a set of physical files in persistent location called "WAL segments" or "checkpoint segments".
* Wal buffers are flushed from the buffer area to wal segments by wal writer.
* Wal buffers memory allocation is controlled by the wal\_buffers parameter.

Clog and other buffers:

* CLOG stands for "commit log", and the CLOG buffers is an area in operating system RAM dedicated to hold commit log pages.
* The commit logs have commit status of all transactions and indicate whether or not a transaction has been completed (committed).
* Work Memory is a memory reserved for either a single sort or hash table (Parameter: Work\_mem)
* Maintenance Work Memory is allocated for Maintenance work (Parameter: maintenance\_work\_mem).
* Temp Buffers are used for access to temporary tables in a user session during large sort and hash table. (Parameter: temp\_buffers).

Physical Files:

* Data Files: It is a file which is use to store data. It does not contain any instructions or code to be executed.
* Wal Files: Write ahead log file, where all transactions are written first before commit happens.
* Log Files: All server messages, including stderr, csvlog and syslog are logged in log files.
* Archive Logs(Optional): Data from wal segments are written on to archive log files to be used for recovery purpose.