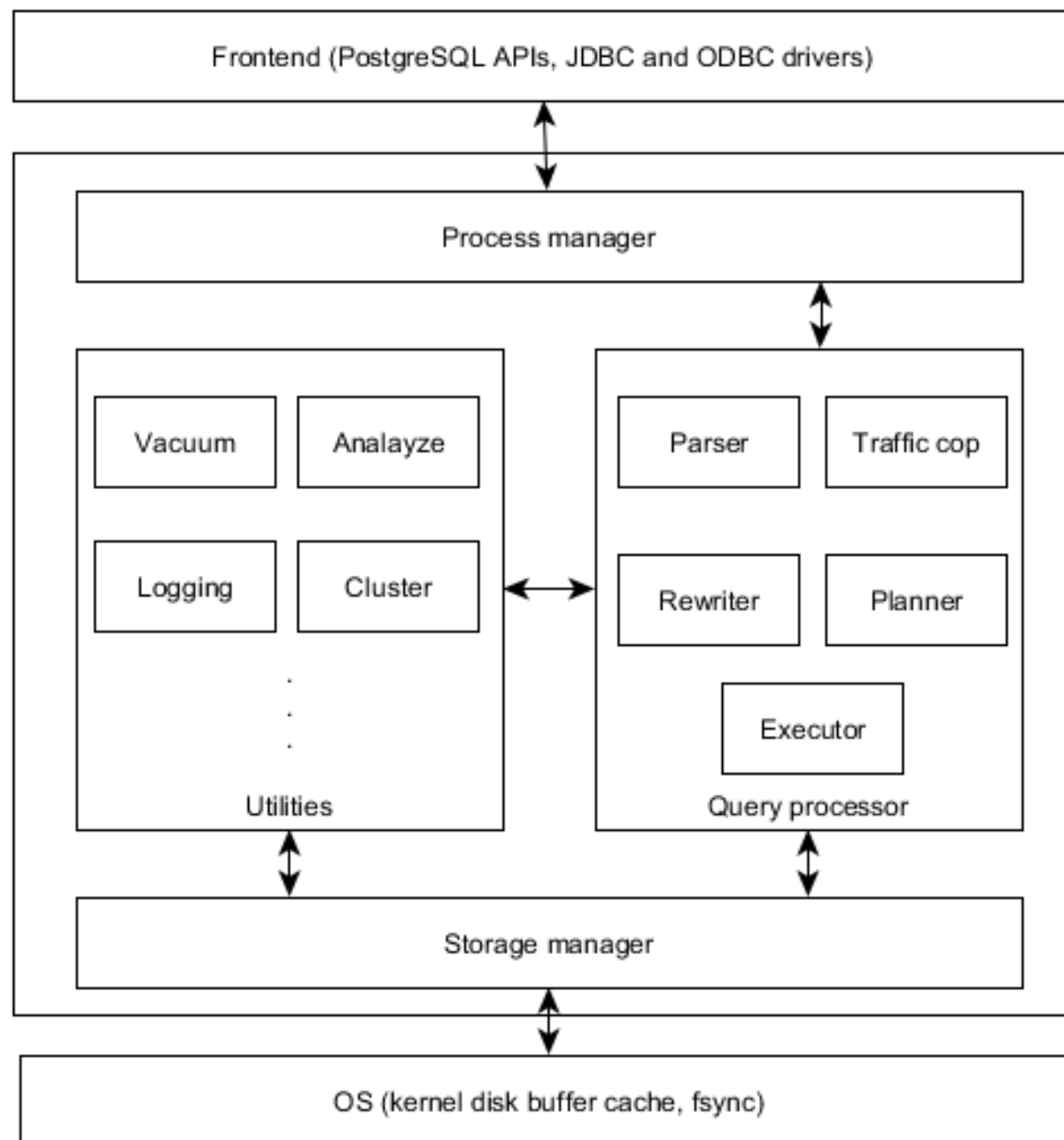


PostgreSQL Architecture

- PostgreSQL uses the client/server model, where the client and server programs can be on different hosts.
- The communication between the client and server is normally done via TCP/IP protocols or via Linux sockets.
- PostgreSQL can handle multiple connections from a client.

- A common PostgreSQL program consists of the following operating system processes:
 - **Client process or program (frontend):** The database frontend application performs a database action. The frontend can be a web server that wants to display a web page or a command-line tool to do maintenance tasks. PostgreSQL provides frontend tools such as `psql`, `createdb`, `dropdb`, and `createuser`.
 - **Server process (backend):** The server process manages database files, accepts connections from client applications, and performs actions on behalf of the client. The server process name is `postgres`. PostgreSQL's main server process forks a new process for each new connection; thus, client and server processes communicate with each other without the intervention of the main server process (`postgres`), and they have a certain lifetime, which is determined by accepting and terminating a client connection.

- The PostgreSQL server could be divided roughly into four subsystems.



- **Process manager:** The process manager client connections, such as forking and the terminating process. It also handles background services, such as logger, and WAL writer processes.
- **Query processor:** When a client sends a query to PostgreSQL, the query is parsed by the parser, and then the traffic cop subsystem determines the query type. A utility query is passed to the utilities subsystem. SELECT, INSERT, UPDATE, and DELETE queries are rewritten by the rewriter, following which an execution plan is generated by the planner. Finally, the query is executed and the result is returned to the client.
- **Utilities:** The utilities subsystem provides a means to maintain the database, such as claiming storage, updating statistics, and exporting and importing data with a certain format or logging.
- **Storage manager:** The storage handles the memory cache, disk buffers, and storage allocation.

- Almost all PostgreSQL components can be configured, including a logger, planner, statistical analyzer, and storage manager.
- PostgreSQL configuration is governed by the nature of the application, such as OLAP, OLTP, and HATP.