



PostgreSQL

Santiago Acosta



¿Qué es?

- PostgreSQL es un SGBD relacional orientado a objetos y libre, publicado bajo la licencia BSD.
- PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.



Historia Postgres

- 1982: se inicia con el proyecto **Ingres** en la Universidad de Berkeley (fue uno de los primeros intentos en implementar un motor de base de datos relacional)
- 1985: se comienza a trabajar en un nuevo proyecto sobre la experiencia de Ingres, dicho proyecto fue llamado **post-ingres** o simplemente **POSTGRES**.
- 1986: se publicaron varios papers que describían las bases del sistema.
- 1988: ya se contaba con una versión utilizable.
- 1989: el grupo publicaba la versión 1 para una pequeña comunidad de usuarios.
- 1990: se publicaba la versión 2 la cual tenía prácticamente reescrito el sistema de reglas.
- 1991: publicación de la versión 3, esta añadía la capacidad de múltiples motores de almacenamiento.
- 1993: crecimiento importante de la comunidad de usuarios, la cual demandaba más características.
- **1994: después de la publicación de la versión 4, el proyecto terminó y el grupo se disolvió.**



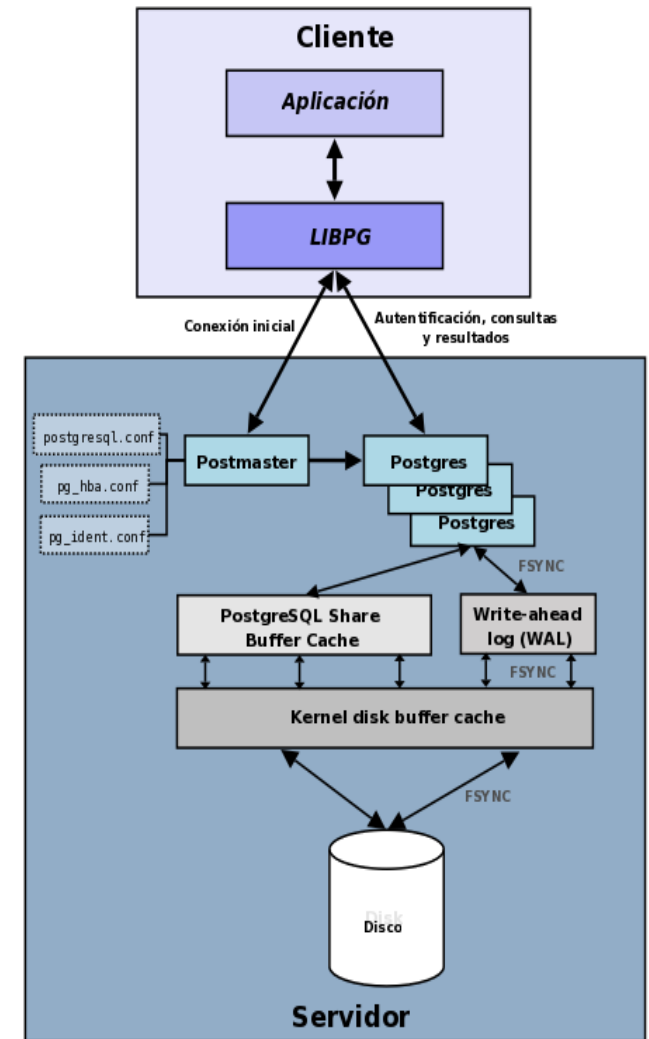
El proyecto PostgreSQL continúa haciendo lanzamientos principales anualmente y lanzamientos menores de reparación de bugs, todos disponibles bajo la licencia BSD, y basados en contribuciones de proveedores comerciales, empresas aportantes y programadores de código abierto mayormente.

- 1995 se añade soporte para lenguaje SQL (antes era lenguaje QUEL), **se crea Postgres95.**
- 1996 decidieron cambiar el nombre de Postgres95 de tal modo que refleje la característica del lenguaje SQL y lo terminaron llamando **PostgreSQL**
- 1997 La primera versión formal de PostgreSQL (6.0) fue liberada.
- 2000, ex inversionistas de Red Hat crearon la empresa Great Bridge para comercializar PostgreSQL y competir contra proveedores comerciales de bases de datos.
- 2001, Command Prompt, Inc. lanzó Mammoth PostgreSQL, la más antigua distribución comercial de PostgreSQL
- 2005, PostgreSQL recibió apoyo del proveedor de base de datos Pervasive Software anunció soporte comercial y participación comunitaria y logró algo de éxito, sin embargo, en julio de 2006 dejó el mercado de soporte de PostgreSQL.
- 2007 EnterpriseDB anunció el Postgres Resource Center y EnterpriseDB Postgres, diseñados para ser una distribución de PostgreSQL incluyendo muchos módulos contribuidos y agregados
- 2008 EnterpriseDB Postgres fue renombrado Postgres Plus en marzo de 2008.



Componentes más importantes en un sistema PostgreSQL.

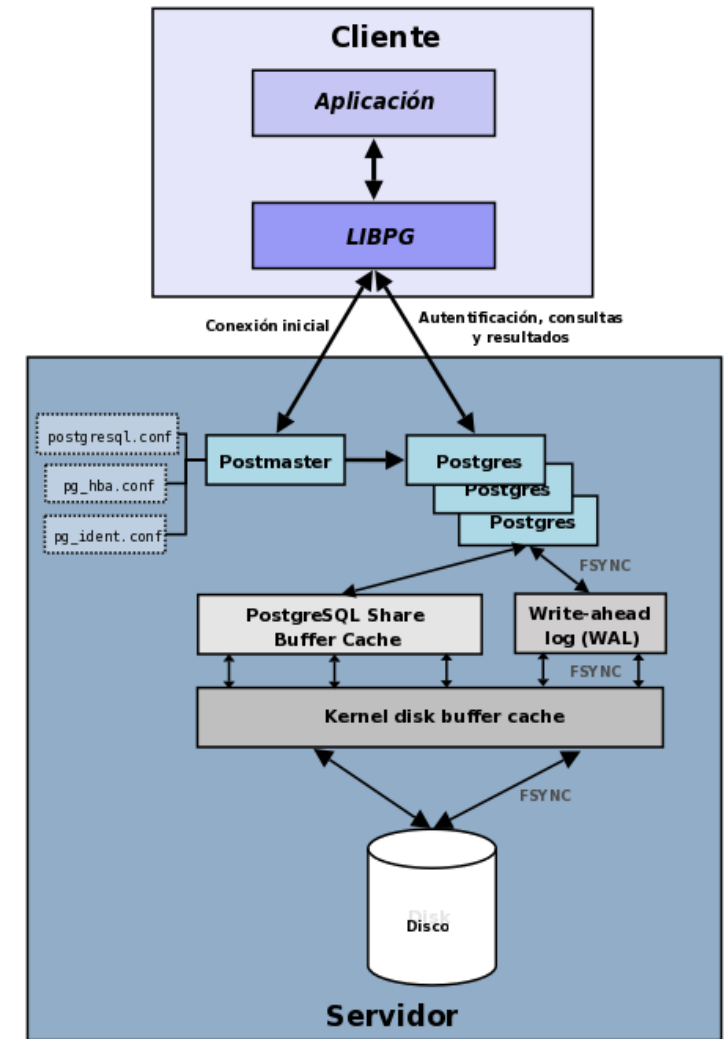
- **Aplicación cliente:** Esta es la aplicación cliente que utiliza PostgreSQL como administrador de bases de datos. La conexión puede ocurrir via TCP/IP ó sockets locales.
- **Demonio postmaster:** Este es el proceso principal de PostgreSQL. Es el encargado de escuchar por un puerto/socket por conexiones entrantes de clientes. También es el encargado de crear los procesos hijos que se encargaran de autentificar estas peticiones, gestionar las consultas y mandar los resultados a las aplicaciones clientes.
- **Ficheros de configuración:** Los 3 ficheros principales de configuración utilizados por PostgreSQL, postgresql.conf, pg_hba.conf y pg_ident.conf





Componentes más importantes en un sistema PostgreSQL.

- **Procesos hijos postgres:** Procesos hijos que se encargan de autenticar a los clientes, de gestionar las consultas y mandar los resultados a las aplicaciones clientes
- **PostgreSQL share buffer cache:** Memoria compartida usada por PostgreSQL para almacenar datos en caché.
- **Write-Ahead Log (WAL):** Componente del sistema encargado de asegurar la integridad de los datos (recuperación de tipo REDO)
- **Kernel disk buffer cache:** Caché de disco del Sistema operativo
- **Disco:** Disco físico donde se almacenan los datos y toda la información necesaria para que PostgreSQL funcione





Características Principales

PostgreSQL ofrece varias características avanzadas:

- 1.Código Abierto:** Es de código abierto y tiene una comunidad activa, lo que permite contribuciones constantes.
- 2.Soporte de ACID:** Asegura transacciones fiables con propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad).
- 3.Extensibilidad:** Permite crear funciones definidas por el usuario, tipos de datos personalizados y operadores.
- 4.Soporte para JSON:** Además de datos estructurados, maneja datos semiestructurados con JSON, siendo ideal para aplicaciones híbridas.
- 5.Alto Nivel de Concurrencia:** Usa el control de concurrencia multiversión (MVCC), lo que reduce el bloqueo de lectura y mejora el rendimiento.
- 6.Replicación y Escalabilidad:** Ofrece replicación sincrónica y asincrónica para mejorar la disponibilidad y la escalabilidad de datos.
- 7.Funciones avanzadas de indexación:** Compatible con varios tipos de índices (B-tree, Hash, GiST, GIN, entre otros) para consultas rápidas.



Comparativa con Otras Bases de Datos

- **MySQL:** PostgreSQL es más avanzado en términos de características y está mejor optimizado para consultas complejas y grandes cargas de trabajo. MySQL es comúnmente preferido para aplicaciones web simples y de menor escala.
- **Oracle:** PostgreSQL es una alternativa económica a Oracle, aunque Oracle es muy utilizado en grandes empresas por su soporte y funciones específicas. PostgreSQL es adecuado para quienes buscan una solución robusta sin un costo elevado.
- **SQL Server:** PostgreSQL es de código abierto, mientras que SQL Server es una solución comercial. PostgreSQL suele preferirse en entornos de Linux, mientras que SQL Server es popular en el ecosistema de Windows.



Ventajas

- **Confiabilidad:** Su integridad de datos es muy alta gracias al soporte ACID.
- **Escalabilidad Horizontal y Vertical:** Facilita la replicación y soporta arquitecturas distribuidas.
- **Adaptabilidad:** Compatible con datos relacionales y no relacionales.
- **Seguridad:** Incluye autenticación sólida, cifrado y control de acceso detallado.
- **Comunidad Activa:** La comunidad contribuye con mejoras constantes y soporte gratuito.

Desventajas

- **Curva de Aprendizaje:** Para usuarios nuevos puede resultar complejo en comparación con otras bases de datos más simples.
- **Rendimiento en Operaciones Simples:** En ciertos casos, operaciones simples pueden ser más lentas comparadas con MySQL.
- **Consumo de Recursos:** Puede requerir más recursos de hardware para grandes operaciones, especialmente con muchas conexiones simultáneas.
- **Soporte Empresarial Limitado:** Aunque cuenta con comunidad activa, algunas empresas prefieren bases de datos con soporte técnico dedicado como Oracle o SQL Server.



Casos de uso



Bloomberg



NETFLIX

