

Práctica 1 - Preguntas

1. ¿Qué otros SMBD existen actualmente en el mercado?

Además de Postgres tenemos otros SMBD que diferenciaremos por su manera de modelar los datos:

Bases de Datos Relacionales, se basan en el modelo relacional en tablas, como: MySQL, MariaDB, Oracle Database y Microsoft SQL server.

Bases de Datos no SQL, para datos no estructurados y grandes volúmenes de información, MongoDB, Cassandra, Redis y CouchDB.

2. ¿Cuáles son las principales diferencias con PostgreSQL?

Cada SMBD tiene sus diferencias particulares con PostgreSQL, pero en general se pueden destacar las siguientes:

- **Completamente open-source.** PostgreSQL es totalmente open-source por lo cual se puede usar en su totalidad de modo gratuito, de igual manera cuenta con un fuerte soporte por parte de la comunidad. Esto en comparación de algunos SMBD que requieren una licencia para ser usados como Oracle. O con un modelo de uso gratuito pero limitado como MySQL.
- **Gran escalabilidad.** Una de las principales características de PostgreSQL es su fácil escalabilidad vertical. A cambio, hay otros SMBD que están mejores optimizados para otras tareas como MariaDB, la cual destaca por sus diversas optimizaciones a la hora de manejar los datos.
- **Carencia de herramientas de revisión.** Una de los principales problemas de PostgreSQL es su falta de herramientas para revisar el estado actual de la base de datos. A comparación de otros SMBD que le permiten al usuario observar el estado actual de la base de datos y con eso la pronta detección de fallos.
- **Fuerte soporte de la comunidad.** Al ser totalmente open-source cuenta con una gran comunidad dispuesta a brindar soporte y nuevas herramientas para extender su funcionamiento. Esto causa que, a veces no cuente con la documentación más clara. A comparación de otros SMBD donde el soporte y las herramientas extras, en caso que haya, sean brindadas exclusivamente por la compañía.

3. ¿Por qué una empresa debería escoger una base de datos *open source*? Una empresa debería considerar el uso de una base de datos open source principalmente por dos razones clave: reducción de costos y flexibilidad para adaptarse a las necesidades del negocio.

- Reducción de costos A diferencia de las bases de datos propietarias, que requieren licencias costosas y pagos recurrentes, una base de datos open source puede utilizarse sin costos iniciales. Esto permite a la empresa destinar su presupuesto a otras áreas estratégicas, como mejorar la infraestructura, contratar talento o desarrollar nuevas funcionalidades.
- Flexibilidad y personalización Al tener acceso al código fuente, las empresas pueden modificar el software del sistema de gestión de bases de datos (SMBD) según sus necesidades específicas. Esto es especialmente útil para organizaciones con requerimientos particulares que no pueden ser satisfechos por soluciones cerradas. Además, pueden optimizar el rendimiento o agregar características propias sin depender de un proveedor externo.

En resumen, optar por una base de datos open source no solo reduce costos, sino que también permite a las empresas invertir en mejorar su software y adaptarlo a sus necesidades, logrando mayor independencia tecnológica y competitividad.

4. ¿Cuáles son las ventajas, para un DBA el trabajar con un SMBD, open source?

Las principales ventajas para un administrador de bases de datos al trabajar con Sistemas Manejadores de Bases de Datos de código abierto radican en múltiples factores, siendo uno de los más evidentes el costo. Muchos de estos sistemas, como PostgreSQL, MySQL o MariaDB, son completamente gratuitos, lo que puede representar un ahorro significativo en licencias y reducir el impacto en el presupuesto de una empresa u organización.

Sin embargo, la mayor ventaja es la flexibilidad y la capacidad de personalización que ofrecen estos sistemas. Un DBA con experiencia y conocimientos avanzados puede acceder al código fuente del SMBD y adaptarlo a las necesidades específicas de su entorno de trabajo, optimizando su rendimiento y ajustándolo a requerimientos particulares. Además, estos sistemas suelen ser altamente configurables, lo que permite ajustar parámetros de seguridad, almacenamiento, concurrencia y rendimiento de manera precisa.

Otra gran ventaja es la posibilidad de contribuir activamente al desarrollo del software. Al ser de código abierto, un DBA no solo puede modificar el sistema para su beneficio, sino que también puede aportar mejoras y optimizaciones a la comunidad, ayudando a que el SMBD evolucione constantemente. Esto fomenta un ecosistema de innovación y colaboración, donde las mejores prácticas y soluciones se comparten y refinan colectivamente.

5. ¿Qué son las bases de datos NoSQL? Menciona 3 ventajas y desventajas contra las bases relacionales.

Una base de datos NoSQL “es un enfoque para el diseño de bases de datos que permite el almacenamiento y la consulta de datos fuera de las estructuras tradicionales que se encuentran en las bases de datos relacionales” (IBM, 2022).

Ahora, en cuanto a ventajas tenemos que:

- **Flexibilidad** tenemos que las bases de datos NoSQL abordan grandes volúmenes de datos los cuales suelen cambiar rápidamente y como consecuencia estas bases son excelentes para un desarrollo ágil, las iteraciones rápidas y envíos frecuentes de código [1].
- **Velocidad** esto es porque NoSQL permite un almacenamiento de información, así como los procesamientos son más rápidos y ágiles para los usuarios [1].
- La **replicación** nos proporciona confiabilidad de los datos y como consecuencia nos asegura el acceso durante el tiempo de inactividad y si en caso de que los servidores se llegarán a desconectar se esta protegiendo contra la pérdida de datos [1].

Por otro lado, las desventajas pueden ser que

- Por la naturaleza de las bases de datos NoSQL los datos que se utilizan suelen ser no normalizados, es decir, no se modelan mediante referencias sino por registros y por ejemplo las empresas trabajan con datos normalizados para evitar anomalías y/o duplicaciones [2] .
- A pesar de que la complejidad de las consultas de las bases de datos NoSQL funcionan bastante bien al realizar consultas con una única tabla, al momento de que crece la complejidad de las consultas es mejor optar por una base de datos relacional [2].
- Tecnología menos madura así como difícil de gestionar [3].

Referencias

- [1] Ibm. *bases de datos nosql*. 12 de dic. de 2022. URL: <https://www.ibm.com/mx-es/think/topics/nosql-databases> (visitado 16-02-2025).
- [2] *Mejores prácticas de seguridad de bases de datos*. URL: <https://www.oracle.com/mx/database/nosql/what-is-nosql/>.
- [3] Coursera Staff. *NoSQL vs SQL: Las diferencias + cuándo usar cada una*. 25 de abr. de 2024. URL: <https://www.coursera.org/mx/articles/nosql-vs-sql>.