**ENSAYO**

Principalmente se puede iniciar hablando de lo importante que es elegir una buena arquitectura al iniciar cualquier tipo de sistema, además de tener un buen análisis entre lo bueno y lo malo que puede llegar a ofrecer una arquitectura en especifica.

Una vez realizado un análisis previo y saber que necesidades se tiene, es importante tener presente la necesidad principal del negocio en el cual se quiere implementar dicha solución.

Primeramente se identifica las principales falencias que tiene la arquitectura de **PostgreSQL** cuando es comparada con otro sistema como lo es **MySQL**, la representación de versiones en el sistema **PostgreSQL** resulta deficiente, ya que por casi 10 años esta no ha evolucionado de forma correcta.

En cuanto a las demás características **PostgreSQL** ofrece un método de almacenamiento de registros por tupla o fila, el cual mapea una dirección de memoria con un ID de registro por tupla, donde se crea una equivalencia para acceder a los datos. Esta estructura o diseño genera una desventaja para **PostgreSQL**, debido a que en el momento de escritura o actualización de pequeños bytes de información; esta acción se convierte en una transacción costosa en cuanto a recursos físicos. Por lo cual manipular altos volúmenes de información se vuelve traumático para este sistema de base de datos.

Este diseño de **PostgreSQL** se convierte en un problema en el momento de replicar datos debido a que ocupa potencialmente mucho ancho de banda en este proceso de copia o salvado de datos, esto conlleva a posibles errores en las réplicas de datos y desencadena hallazgos difíciles de identificar o rastrear.

Otra limitante que posee este sistema de base de datos es el siguiente, se dificulta la réplica de datos entre diferentes versiones, lo cual se convierte en algo complicado para los administradores de base de datos, ya que si la base de datos sufre una actualización y se necesita realizar una réplica, se debe tener en cuenta el riesgo que sufre la información y el peligro de perder datos en el momento de realizar una operación de estas.

Además de analizar el comportamiento y arquitectura de **PostgreSQL** entraremos a analizar **MySQL** como herramienta importante en grandes infraestructuras. Esta soportada por un motor de almacenamiento llamado InnoBD, la diferencia más importante entre estos dos sistemas de base de datos **MySQL** y **PostgreSQL** parte del motor de almacenamiento debido a que InnoBD mantiene una estructura secundaria. En lugar de mantener un puntero en disco como lo hace **PostgreSQL**, esto crea una ligera desventaja para InnoBD frente a **PostgreSQL** cuando se realizan búsquedas. Pero **MySQL** posee un diseño más compacto frente a la exploración de las tablas para así identificar rápidamente la fila que se necesita ya sea mediante eliminaciones, actualizaciones o consultas. En **MySQL** es permitido diferentes tipos de replicación lo cual significa que para una arquitectura de estas es poco probable que generar un error de corrupción de datos, finalmente las réplicas en **MySQL** se hacen triviales ya que producir copias de datos entre diferentes versiones no cambia la forma de realizar la réplica, o alguna alteración de datos o riesgo de producir daños en este proceso.

En cuanto a otras ventajas de **MySQL** encontramos un rendimiento mucho mejor al de **PostgreSQL,** el almacenamiento también genera un valor agregado para **MySQL** ya que este no utiliza directamente la memoria cache como lo hace **PostgreSQL,** en cambio, **MySQL** tiene su propio diseño personalizado de memoria que es muy similar al de la memoria cache, este diseño de almacenamiento brinda mejor rendimiento para **MySQL.** Hablando de conexión **MySQL** implementa conexiones concurrentes al generar un hilo de conexión, esto resulta una gran ventaja para realizar conexiones simultáneas y generar más velocidad en la base de datos.

Finalmente estos dos sistemas de bases de datos son ampliamente reconocidos, la usabilidad de cada uno depende mucho de la necesidad que se tenga en el negocio, puede llegar a ser muy importante antes de decidir utilizar un sistema u otro conocer a fondo la arquitectura del negocio y el volumen de datos que se quiere manejar en nuestro sistema.