Inicialmente Uber estaba diseñado para la atención de una área geográfica limitada y su programa era ajustado a las necesidades del momento, sin tener en cuenta la expansión del programa. Uber inicia labores en otras zonas y empieza a tener dificultades con el programa, ya que este estaba hecho a la medida y era difícil integrar nuevos componentes de código, lo cual se presento como un reto para los ingenieros de uber,la misión era encontrar la forma de integrar de nuevo código a su programa. Para esto Uber decidió seguir el ejemplo de empresas como: Amazon, Twitter, Netflix, SoundCloud y romper el esquema monolítico base para formar una arquitectura orientada a servicios (SOA) para el uso de micro servicios, dejando asi cada funcionalidad lo mas atómica posible, haciendo esto se podía replicar la información según las necesidades y ubicaciones , creando asi mayor redundancia y se hace posible la  conmutación. Asignar bases de datos de áreas geográficamente mas cercanas , pero cada ciudad se copian en un centro de datos diferente en otra ubicación.

Entre las limitaciones para este cambio surge el motor de la base de datos el cual era Postgres y como habian montado la arquitectura de su aplicación: ya que Usaban una tabla para almacenar datos globales, esto es una manera muy antigua de montar la arquitectura,

Ejemplo de como trabaja postgres, SI SE CREA UNA Tupla en una tabla esta es inmutable o sea que no se puede actualizar, para ello lo que hace postgress es crear una nueva tupla con un nuevo identificador, creando redundancia en mucha información y creando muchos procesos para tratar de tener esta información disponible, los índices son una herramienta para tal fin. Las dificultades con este motor se presentan con la cantidad de escrituras al disco, ya que cualquier modificación en la base de datos, significaba un gran costo en el motor de base de datos para llevar a persistencia esta información, creando gran trafico en la red, por consecuencia se podrían ver afectados los servicios productivos, teniendo en cuenta que postgres realiza replicaciones de la información, el ancho de banda para la aplicación web se veía afectada por los proceso de la base de datos.

Otra dificultad que surge con este motor es la corrupción de la información, al realizar una actualzicion de versión, alterando la información de la base de datos, dejando la información inconsistente, dejando la lógica de la aplicación un poco afectada, esto es a groso modo el factor por el cual se hace el cambio hacia mysql; La diferencia arquitectónica más importante es que, mientras Postgres correlaciona directamente los registros de índice con las ubicaciones en disco, InnoDB mantiene una estructura secundaria. MySQL implementa conexiones concurrentes al generar un hilo por conexión. Esto es una sobrecarga relativamente baja, lo cual representa una baja obstensible en el uso de la red para los procesos, facilitando y dejando libre o mejor no saturando la red apra los procesos propios de la base de datos,teniendo asi mayor disponibilidad la red para el aplicativo web

Lo bueno de postgres

Posee una gran escalabilidad.

Implementa el uso de rollback’s, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz,

Tiene la capacidad de comprobar la integridad referencial,

Lo malo de postgres

Consume gran cantidad de recursos.

Tiene un límite de 8K por fila, aunque se puede aumentar a 32K, con una disminución considerable del rendimiento.

1. Lo bueno de mysql

Sin lugar a duda, lo mejor de MySQL es su velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento.

Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.

Las utilidades de administración de este gestor son envidiables para

muchos de los gestores comerciales existentes, debido a su gran facilidad

e configuración e instalación.

Tiene una probabilidad muy reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que está.

Lo malo de mysql

Carece de soporte para transacciones, rollback’s y subconsultas.