

## Arquitectura de Netflix

Netflix necesita no solo almacenar grandes cantidades de datos, sino que también necesita enviarlos a los distintos clientes de manera constante para crear un flujo de datos que permita a los usuarios poder ver vídeos en streaming sin que el vídeo se corte.

Es por esto que actualmente, Netflix utiliza 3 “bases de datos” distintas, y cada una de ellas cumple un papel fundamental en la empresa.

Cabe destacar que todas las bases de datos que utilizaron rompen con el modelo relacional y son noSQL, debido a que para lograr los rendimientos que estaban buscando, una base de datos en la nube, noSQL les pareció la mejor opción.

En primer lugar utilizaron Amazon SimpleDB, debido a que se ajustaba a lo que estaban buscando, almacenamiento en la nube de manera sencilla y eficiente.

Amazon SimpleDB elimina las tareas de administración, permitirlas a los desarrolladores almacenar datos y consultarlos mediante servicios web. Otra ventaja de este sistema es que crea réplicas de sus datos y las administra regionalmente, ayudando al rendimiento a la hora de acceder a los datos o mandarlos.

En segundo lugar, para mejorar el rendimiento, Netflix utiliza sistemas basados en Hadoop. En este caso, optó por Apache HBase. Esta base de datos noSQL permite ampliar los clústers de manera sencilla, ayudando a redistribuir el procesamiento entre los distintos nodos en tiempo tiempo de ejecución.

Al igual que Amazon SimpleDB, posee la opción de utilizar la geografía. HBase utiliza regiones con servidores que ayudan a mejorar el desempeño al momento de manejar los datos.

Por último, Netflix decidió utilizar Cassandra, debido a que es extremadamente estable al momento de manejar datos crudos, ya que los datos entrantes son escritos en una estructura llamada *memtable* que actúa como una especie de cache. Una vez que esta memoria está llena, se escoba en una *SSTable*, que es un sistema para guardar los datos basado en archivos.

Además Cassandra es un sistema distribuido con el protocolo punto a punto, permitiendo una mejor comunicación entre los distintos clústers, sin sacrificar el rendimiento que requiere algún protocolo de comunicación basado en TCP.

Con lo anterior, Cassandra es una gran opción para guardar datos, debido a que cada clúster posee nodos independientes que distribuyen la carga de procesamiento y escritura, permitiendo ampliar o reemplazar clúster que estén dañados.

## **Arquitectura Lambda**

Ahora, si llevamos estos tres sistemas a la arquitectura lambda vista en clases, notamos una gran similitud con esta.

Si bien, cada base de datos podría tomar el rol de cualquiera de las capas debido a sus características similares, hay algunas que podrían tener un mejor desempeño según la capa en la que estén.

Por ejemplo, si tomamos la base de datos ApacheHBase, notamos que debido a su sistema basado en haddops, y por la redistribución del procesamiento, podría encajar de mejor manera en la capa de velocidad.

Por otro lado, si comparamos la capa batch, esta podría verse mejor reflejada con Cassandra, debido a que como posee un mejor manejo de los datos crudos, seria la mejor opción para almacenar los datos maestros, debido a que no seria necesario preocuparse del formato de estos, solo

Por ultimo, para la capa de servicios, creo que Amazon SimpleDB es la mejor opción, debido a que como logra eliminar las tareas de administración, seria la mejor forma para ir replicándola cada vez que se necesiten consultar datos pre calculados, o algún set de datos específicos, sin tener que intervenir mucho en la base de datos. Bastaría con un solo llamado al servicio web que Amazon provee.

En conclusión, las tres bases de datos que utiliza Netflix tienen características similares, como soporte en regiones, rapidez y estabilidad, por lo que cualquiera podría cumplir el rol necesario en algunas de las capas de la arquitectura Lambda, no obstante, la clasificación que realizamos anteriormente parece que la que mejor se puede adaptar a la arquitectura.

## **Bibliografía**

### General

<https://medium.com/netflix-techblog/nosql-at-netflix-e937b660b4c>

### Amazon simpleDB

<https://aws.amazon.com/es/simpliedb/>

### ApacheHBase

<http://hbase.apache.org/book.html#arch.overview>

### Cassandra

<http://cassandra.apache.org/>

[https://www.tutorialspoint.com/cassandra/cassandra\\_architecture.htm](https://www.tutorialspoint.com/cassandra/cassandra_architecture.htm)

[http://docs.datastax.com/en/cassandra/2.1/cassandra/architecture/architectureIntro\\_c.html](http://docs.datastax.com/en/cassandra/2.1/cassandra/architecture/architectureIntro_c.html)