

Real-Time Data

Introducción

Real Time Data indica la información que se transmite inmediatamente después de la recolección en esta no hay demora en la procedencia de la información correspondiente y es utilizada normalmente en navegacion y rastreo.

Real-time data

Es toda aquella información que se entrega inmediatamente después de su recolección. Es usada usualmente para navegación o rastreo, algunos usos de este término es confundido con dynamic data, en realidad la presencia de real time data es relevante incluso si es dinámica o estática.

Es usado de dos maneras :

- Real-Time data en economía
- Real-time data bidding

Real-Time data en economía usualmente son basadas en estimados preliminares, y además son ajustados frecuentemente como un mejor estimado se puede volver disponible, esta data que se ajusta despues es llamada 'revised data', los términos real time data y real time en economía fue acuñado por Francis Diebold y Glenn Rudebush, estos definieron el análisis de real-time data como 'El uso de información secuencial se vuelve necesario como la historia evoluciona.'

Para entender mejor la precisión de la información económica y como afecta a las decisiones económicas, algunas organizaciones, han desarrollado bases de datos disponibles que contienen ambos real time data y estimaciones posteriores de la misma data.

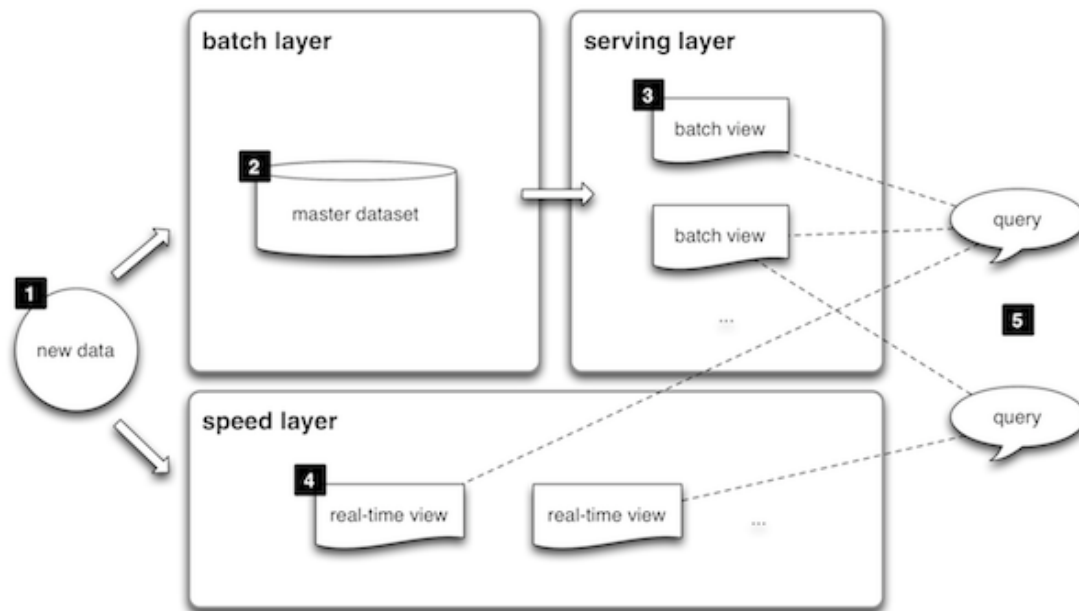
Real time bidding se refiere a la pragmática de subastas en tiempo real que venden impresiones de anuncios digitales. Entidades en ambos lados compra y venta, el área de venta requiere al menos acceso instantáneo a la información en orden para la toma de decisiones, forzando a la real-time data ajustarse a la vanguardia de sus necesidades. Para apoyar sus necesidades, nuevas estrategias y tecnologías como Druid¹ por ejemplo.

Arquitectura Lambda

Es una Arquitectura de procesamiento de datos escalable, generica y tolerante a fallos. Se enfoca en satisfacer las necesidades para Sistema robusto que es tolerante a fallos, ambos contra fallos de hardware y errores humanos, siendo capaz de servir una amplia gama de cargas de trabajo y casos de uso , y en el cual la baja latencia precisa de lecturas y actualizaciones. El Sistema resultante debe ser linealmente escalable.

¹ Open source data store

Se divide de la siguiente manera:



Toda la información que entra al sistema es enviada a la capa de lote y la de velocidad para el procesamiento.

Batch Layer tiene dos funciones:

- Copia maestra de toda la información
- gestión del conjunto de datos principal (inmutable que no cambia en el tiempo)
- validez de calcular los puntos de vista de lote.

Serving layer

Indexa la vista de lotes para que esta pueda ser usada en baja latencia y ad-hoc.

Speed Layer:

Compensa la alta latencia de cambios en la serving layer y se ocupa de sólo los datos más recientes.

Cualquier consulta entrante puede ser contestada mediante la fusión de los resultados de vistas lotes y vistas en tiempo real.

Los datos en tiempo real: la data puede ser procesada o devuelta en la misma tasa en que se esta generando.

- inserciones, actualizaciones, consultas en stream, data ingresada como un evento en el tiempo.
- hay que tener la habilidad de procesarlos en tiempo real
- en lugar de que la info entre como un lote, entra como un evento.
Ej. kafka, samza, spark

Bases de Datos NoSql

- Sin transaccionalidad
- sin bloqueos.
- Clave valor.
- Familias de columnas que no son necesariamente usadas.
- No hay bloqueos porque está en caché.

Bibliografía y fuentes de información

- Conferencia 'Real Time Data' impartida por el conferencista Jose Luis Guzman por por Katch - COECYS 2015.
- https://en.wikipedia.org/wiki/Real-time_data
- <http://lambda-architecture.net/>