Lección 3.5 Procesos de integración:

En lecciones anteriores vimos como la alta heterogeneidad o Variedad (5v's) es una de las características más significativas de las fuentes de datos Big Data. En un escenario real de procesamiento y análisis de Big Data tendremos una o más fuentes que hemos de combinar, las cuales presentan distinto nivel de estructura (recordemos datos estructurados, semi-estructurados y nada estructurados), distinto modelo o forma de organizar los datos y otras tantas diferencias, que complican su combinación o integración.

1. Introducción

Pero, ¿Por qué es necesaria la integración de la fuentes de datos en nuestras aplicaciones de Inteligencia de Negocio?

- Una de las ventajas del Big Data es el aumento del número y diversidad de las fuentes de datos disponibles para nuestros objetivos de análisis de nuestra organización, en la búsqueda de la mejora del rendimiento de nuestro negocio
- Sin embargo, si procesamos y analizamos cada fuente de datos de forma individual estamos perdiendo un gran parte del conocimiento implícito en esas fuentes, únicamente visible cuando las analizamos en conjunto.
- **Integración**: Proceso en el que las diversas fuentes de datos seleccionadas para nuestra aplicación de Inteligencia de Negocio (BI) son procesadas para su combinación y almacenamiento según un modelo de datos común (ej. modelo multidimensional)
    - Como paso previo a los procesos de análisis para extracción de conocimiento útil.

2. Características de los procesos de Integración

Sin embargo, y como ya hemos ido avanzando desde el inicio de este curso, la integración de las fuentes de datos en una aplicación de Inteligencia de Negocio es un proceso no trivial y más aún si cabe en el contexto del Big Data.

- La integración de las fuentes de datos en Big Data requiere de un gran esfuerzo humano debido a factores como:

    - Distinto nivel de estructura (recordemos datos estructurados, semi-estructurados y nada estructurados)
    - Distinto modelos o forma de organizar los datos
    - Campos y valores de filas que hacen referencia a la misma entidad pero que presentan ligeras diferencias sintácticas o semánticas
- Es necesario valorar dicho esfuerzo antes de diseñar un proceso de integración para múltiples fuentes Big Data
    - En ocasiones el coste de desarrollo puede llegar a ser inviable, caso el que hemos de reducir el número de fuentes usadas o relajar los requerimientos en la Calidad de los datos integrados.

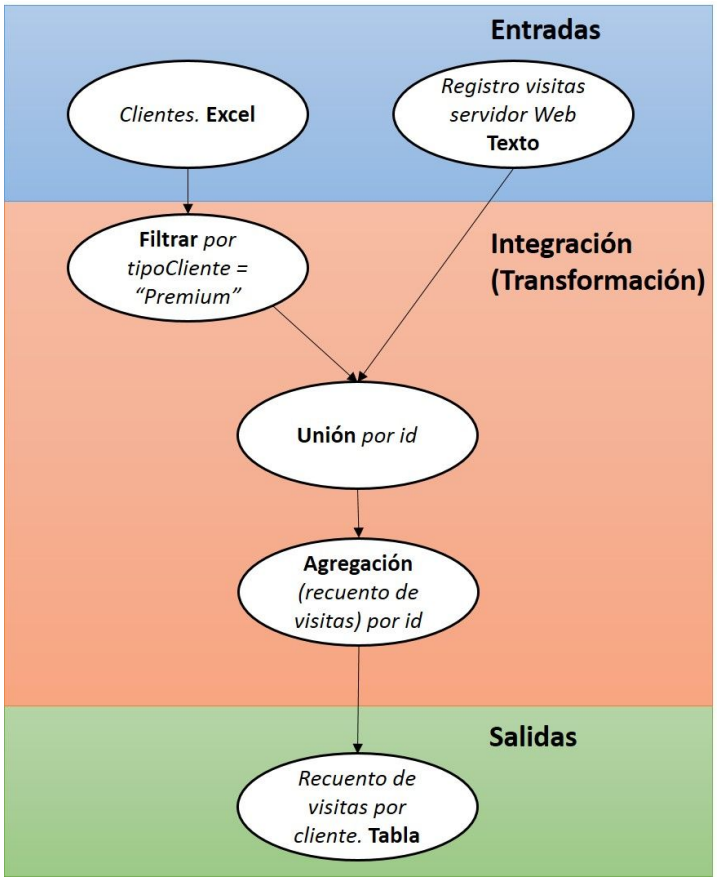
3. Procesos ETL

Pero, ¿Cómo desarrollar un proceso de integración?. La integración de las fuentes de datos se lleva realizando desde finales de la década de los 80 y principios de los 90 con la tecnología tradicional de Almacenes de Datos (DW), por lo que existen numerosas técnicas con una contrastada efectividad y que se siguen aplicando en el nuevo contexto de Big Data.

- Los procesos de integración se pueden implementar con procesos ETL (Extraction, Transformation and Load)
    - Extracción: Conexión con las fuentes datos para obtener los datos de las fuentes a integrar
    - Transformación: Aplicación de diversas operaciones sobre las fuentes de datos para hacer posible la **integración**
        - Operaciones de unión “join”, filtrado, limpieza,...
    - Load: Carga de los datos integrados bajo un nuevo modelo de datos común en el almacén o repositorio de datos Big Data
- Los **procesos ETL** se suelen diseñar e implementan como **flujos de datos**
    - (Por lo general) La unidad de datos es la fila o tupla, formada por una o más columnas y valores para esas columnas.
    - Podemos representarlos de forma matemática como un DAG (Grafo Acíclico Dirigido), donde las aristas son los datos y los nodos son los operadores que permiten procesar los datos.

Ejemplo:

Un ejemplo de ETL, es el Grafo Acíclico Dirigido que podemos ver en esta transparencia. En este caso tenemos como entradas un fichero de Excel con los datos de los clientes de una aplicación web (ej. id, nombre, teléfono, tipo de cliente (ej. estándar, premium)), y, por otro lado, un archivo de texto con el registro de visitas a las distintas páginas de nuestra aplicación web generado de forma automática por el servidor web correspondiente (ej. contiene idCliente, fechaVisita, urlVisitada). Es este último, los datos apenas tienen estructura, como algún delimitador que separa los distintos campos (ej. comas, espacios en blanco…) o los saltos de línea que determinan un registro o visita. Si por ejemplo, queremos estudiar el número de visitas de los clientes premium hemos de crear un proceso ETL similar al mostrado.



Como puede observarse, para la integración se aplica el filtrado por tipo de cliente premium sobre los datos de clientes de Excel. Tras esto se aplica una operación de unión (o join) que une los datos de cada cliente con los de cada visita que ha realizado ese cliente. Tras esto

aplicamos una función para la agregación o agrupación que computa el recuento de visitas para cada cliente. Por último, la salida es una tabla con los datos de número de visitas para los clientes premium. Es decir información ya integrada y estructurada en un modelo de datos común que podemos almacenar, por ejemplo, en un Almacén de Datos en un repositorio Big Data como Hadoop Hive. De esta forma, los datos integrados ya están optimizados y disponibles para su posterior análisis en forma de tabla OLAP, informe o cuadro de mando, por ejemplo.

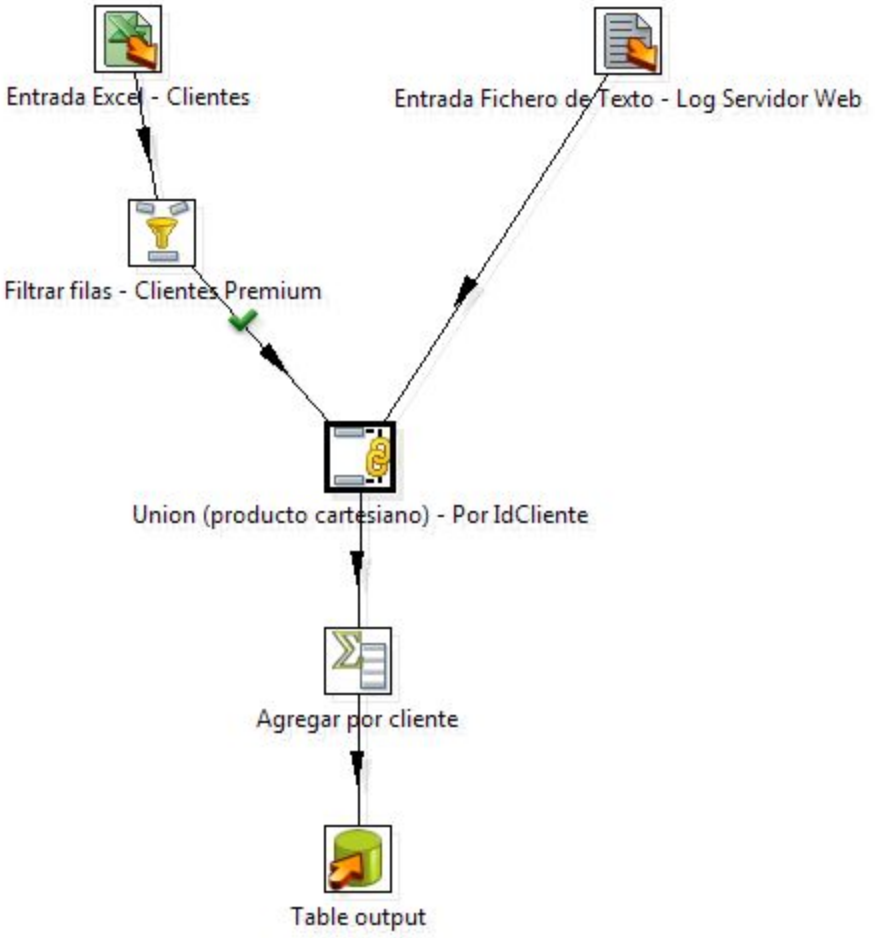
4. Implementación de procesos ETL

- Podemos desarrollar procesos ETL con cualquier lenguaje de programación (ej. Java)
    - Sin embargo, la implementación del flujo de datos y las operaciones típicas como la unión o el filtrado requeriría de un grandísimo esfuerzo
- Existen herramientas de inteligencia de negocio y lenguajes específicos para el desarrollo de procesos ETL en forma de flujos de datos
    - Herramientas BI: Pentaho Data Integration, Tableau, SSIS,...
        - Combinan el diseño visual de los procesos como características avanzadas como la introducción de código Java, JavaScript u otros lenguajes
    - Lenguajes de flujos de datos: Apache Pig dentro del entorno Hadoop
        - Más potencia y control sobre los flujos de datos sin aumentar mucho la dificultad
    - Para dar soporte a Big Data algunas de estas herramientas y lenguajes soportan la ejecución de forma distribuida sobre Hadoop / MapReduce

5. Ejemplo de implementación de ETL

Por ejemplo, una posible implementación del proceso ETL de integración que pusimos antes como ejemplo, usando la herramienta Pentaho Data Integration podría ser la que vemos en esta imagen.

Los programas de ETL como Pentaho Data Integration permiten diseñar el proceso ETL de forma visual como un flujo de datos. Estos programas incluyen numerosas funciones precreadas para las distintas partes de un proceso ETL, pero que nosotros podemos configurar para adaptarlas a nuestros objetivos de integración específicos, así como para la conexión con las fuentes y destinos de los datos, ya sean Bases de Datos Relacionales, NoSQL o el clúster HDFS.



6. Implementación de procesos ETL

- Aunque hasta ahora lo habitual era llevar a cabo la integración y almacenar los datos integrados en un repositorio antes de llevar a cabo cualquier proceso de análisis (ej. OLAP o Dashboard), en Big Data se aplica cada vez más la técnica de la integración “al vuelo” durante el análisis
    - Los datos se integran en el momento del análisis, sin necesidad de almacenarlos previamente en un Almacén de Datos
    - Hablaremos más de esta técnica cuando hablemos de las técnicas de análisis para Big Data en el módulo 5

- En los procesos ETL, además de las operaciones para integración, con frecuencia se aplica procesamiento para garantizar la Calidad de los datos antes y después de la integración