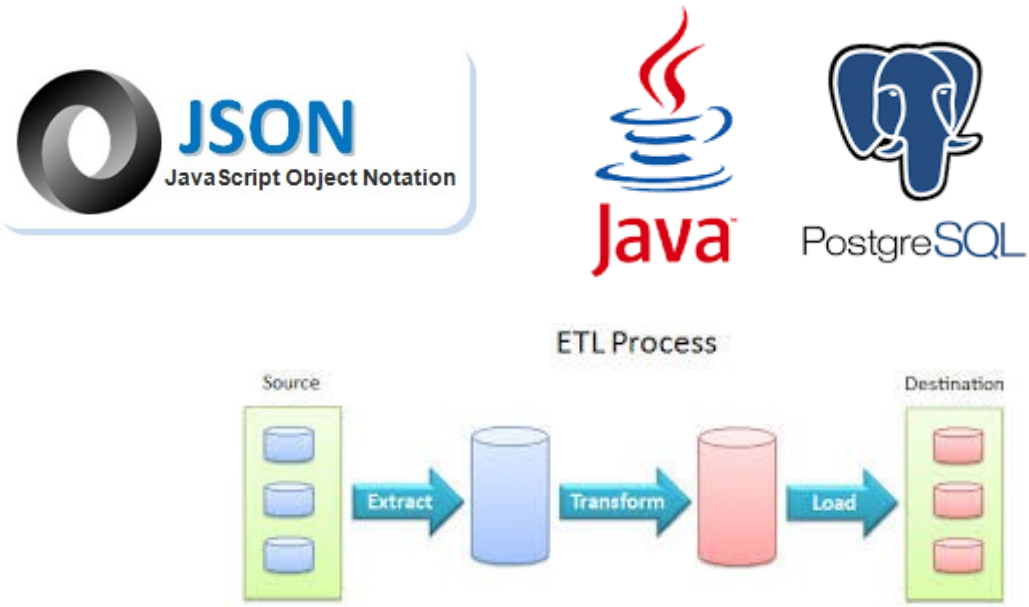
INTELIGENCIA DE NEGOCIO



IMPLEMENTACIÓN ETL (ACTIVIDAD C3 - EXTENDIDA)

MOISÉS FRUTOS PLAZA
MÁSTER DE BIG DATA

ENTREGADO EL 1 DE JUNIO DE 2017

ÍNDICE

Índice de contenido

1. INTRODUCCIÓN

Esta actividad pretende extender, **y corregir algunos aspectos de**, la Actividad C3, en la cual se presentó un modo de Extracción Transformación y Carga de datos (ETL), pero que no llegó nunca a implementarse. Por lo que se pretende precisamente eso su **implementación** y como a lo largo de la misma han surgido algunas cuestiones, que debido a ellas, han requerido, o bien **correcciones** o bien **cambios al planteamiento inicial**.

Esta memoria, aparte de explicar el proceso de implementación, pretende justificar el porqué de esas correcciones y cambios con el fin de garantizar un Data Warehouse lo más consistente y fidedigno a este modelo de negocio.

2. ¿QUÉ SE ESPERA AL FINAL?

Tenemos un modelo claro de negocio que hemos visto a lo largo de las actividades, sin embargo las fuentes de los datos, aunque todas ellas estén en JSON, sus arquitecturas son heterogéneas, porque proceden de distintas fuentes de datos. Eso significa que no se puede utilizar una herramienta como Spoon debido a que dicha herramienta sólo trabaja con arquitecturas homogéneas, lo cual es una gran limitación para el proceso ETL, que precisamente puede partir de una o varias fuentes de datos heterogéneas, y ya los procesos de limpieza y transformación se encargarían de homogeneizarlas. Pero Spoon se ha mostrado demasiado limitado en este aspecto por lo que se ha preferido utilizar Java ¿Porqué? Porque el usuario final, o el gerente de este modelo de negocio, lo que espera es poder visualizar los datos de forma correcta y que éstos estén almacenados de forma homogénea en un sólo sitio al cual denominamos precisamente Data Warehouse.

No es motivo de esta actividad crear ese cuadro de mandos, en el que el gerente pueda visualizar los datos, pero sí dejar los mismos preparados y almacenados en el DW para cuando dicho cuadro de mandos posteriormente se cree, si así se decide, pueda acceder a ellos de forma directa.

3. PROCESO DE EXTRACCIÓN

La mayoría de métodos de extracción utilizados han sido solicitudes directas a la API-REST de una web, o dispositivo RPC, para que explícitamente mediante dicha solicitud recibir como respuesta el fichero JSON correspondiente. Por lo que no ha sido necesario la utilización de wrappers al final, eso que nos ahorramos. Lamentablemente, no dispongo de los medios necesarios para hacer lecturas de los contadores de corriente, al no ser éstos contadores inteligentes, y su arquitectura JSON me la he tenido que inventar sobre la marcha: los podemos llamar JSON-MOCK. Lo mismo ha sucedido con el cliente minador Claymore que al ser privativo no ha habido forma posible de acceder a su API, por lo que también he tenido que adaptar la lectura de sus datos a un JSON-MOCK.

De todos modos, aunque sean JSON-MOCK, éstos contienen los datos fidedignos tanto

de los clientes minadores como de las lecturas de los contadores ya que los tomé de los mismos en persona durante su ejecución.

4. PROCESOS DE VERIFICACIÓN Y LIMPIEZA

Si algo bueno tiene Java es su gran versatilidad a la hora de tratar con distintas fuentes de datos. De ahí su elección para todos los procesos que van a venir a continuación.

La librería ***JSONObject*** de Java ya implementa de forma implícita la verificación del fichero JSON bien construido al momento de leerlo, y si no está bien construido da un error. Aún así tiene un método para validar el objeto JSON una vez éste se ha leído y guardado en memoria como un objeto JSON en Java.

Una vez validados los objetos JSON como bien construidos, el siguiente proceso era realizar la limpieza dando el formato correcto a los datos que necesitamos. Muchos de ellos vienen en modo texto, o con comillas, o con unos formatos que no son los que el DW espera. Y es aquí cuando se hace ese proceso de limpieza de los datos en bruto para refinarlos.

5. PROCESO DE TRANSFORMACIÓN

Una vez que ya tenemos los objetos JSON verificados y refinados, es el momento de hacer las transformaciones necesarias para terminar de adaptar todos los datos que necesitamos para nuestro DW. Este proceso es crucial ya que hay algunos datos que ya vienen dados pero hay otros que deben de ser calculados, de modo que al final creamos **objetos Entidad**, los cuales son los que el cargador al final va a cargar en el DW. Esto es un cambio con respecto a la Actividad C3, en la que se utilizaba crear un sólo JSON y cargarlo mediante una herramienta de terceros, pero una vez hecha la aplicación JAVA ví más conveniente crear los objetos Entidad y cargarlos en el DW mediante el propio Java utilizando una capa DAO mediante JDBC.

Cada objeto Entidad se corresponde con una de las dimensiones del DW y, por supuesto, con la tabla de hechos. De este modo, creo que el proceso de transformación quedaba de una forma más claro y lógico.

6. PROCESO DE CARGA EN EL DATA WAREHOUSE

Una vez tenemos los objetos Entidad creados y preparados. Es el momento de cargarlos finalmente en el DW. La base de datos elegida ha sido PostgreSQL, porque aparte que ha sido la que hemos visto para la asignatura, es una base de datos muy versátil, muy potente, que permite la realización de cubos y consultas en base a estos, lo cual nos viene de maravilla para la Inteligencia de Negocio en este modelo de negocio.

Para cargar los datos simplemente he tenido que usar el driver JDBC para Java y

crearme una capa de persistencia DAO que se encargara de coger los objetos Entidad y persistirlos en dicha base de datos.

Con el fin de que usted pueda probar la aplicación ETL, he procurado crear un script con el esquema DW vacío en SQL para que se pueda cargar en PostgreSQL, así como los fuentes JSON ya bajados o creados, así como todo el código JAVA de la implementación ETL (posiblemente haya que cambiar los parámetros de conexión a la base de datos en dicho código según su servirdor PostgreSQL).

7. CONCLUSIÓN FINAL Y CONFESIÓN DE SECRETOS

La conclusión final que extraigo es que los datos por sí mismos no valen para nada si no sabemos como tratarlos primero. Si queremos extraer conocimiento de los datos tenemos que refinarlos y persistirlos, en este caso mediante un proceso ETL. Con los datos refinados y persistidos en un sólo DW, y en el cual se seguirán actualizando cada x tiempo, tendremos en un sólo sitio y de forma confiable toda la información que necesitamos, la cual podremos consultar, extraer, visualizar, e incluso si quisieramos posteriormente realizar minería de datos y obtener modelos predictivos en base a ellos para futura toma de decisiones, quiero decir que los datos recopilados en el DW no sólo nos sirven para tomar decisiones ahora, también nos pueden servir para tomar decisiones de cara al futuro.

Ahora viene mi confesión, me da un poco de vergüenza pero en todo momento este modelo de negocio se ha basado en un negocio real que yo mismo estoy llevando desde el año pasado a nivel personal. Después de monitorizar datos durante 3 meses, realicé un pequeño modelo predictivo para saber como iba a evolucionar el Bitcoin, eso fue en Junio de 2016, mi modelo me dijo que la tendencia pese a las enormes fluctuaciones era que la moneda iba a doblar su precio en unos 8 meses, en aquel entonces estaba por los 600$. Quino, que fue el profesor que me llevó el TFG, me dijo que: “el primero que tienes que creer en tu trabajo eres tú”, y eso hice. Invertí todos mis ahorros en formar ese modelo de negocio. Me satisface poder decir que ya he alcanzado el ROI ya que el precio se ha casi cuadriplicado. Fue una decisión arriesgada, a lo mejor un tanto alocada, pero al final de esto se trata el Big Data ¿no? ¿Cómo puedo recomendar a alguien que se juegue su dinero sin antes jugarme el mío? Con problemas de juguete obviamente no. Por eso decidí hacer esto. Porque pienso que aparte de conocimientos todo esto conlleva una gran responsabilidad y quería sentir ese peso, el peso de la responsabilidad. No sé que nota sacaré al final, pero el modelo ha funcionado, y creo que al final lo que importa es que las cosas funcionen, porque somos personas y porque somos profesionales. Muchas gracias por todo Manolo.

Nota: No te meto código en la memoria para no hacerla demasiado densa de forma innecesaria. De hecho no quería que me saliera demasiado larga porque quería hacer hincapié en las ideas importantes, porque pienso que hay que centrarse en lo importante y no perderse en los detalles. Aún así si quieres más detalles sobre esto siempre me tienes a tu entera disposición.