Курсовой проект по направлению «Data-Engineering (BDS)»

Задание №1

В некотором банке внедрили новую frontend-систему для работы с клиентами, а так же обновили и саму базу данных. Большую часть данных успешно были перенесены из старых БД в одну новую централизованную БД. Но в момент переключения со старой системы на новую возникли непредвиденные проблемы в ETL-процессе, небольшой период (конец 2017 начало 2018 года) так и остался в старой базе. Старую базу отключили, а не выгруженные данные сохранили в csv-файлы. Недавно банку потребовалось построить отчёт по 101 форме. Те данные что остались в csv-файлах тоже нужны. Загрузить их в новую БД не получиться из-за архитектурных и управленческих сложностей, нужно рассчитать витрину отдельно. Но для этого сначала нужно загрузить исходные данные из csv-файлов в детальный слой (DS) хранилища в СУБД Oracle / PostgreSQL.

**Задача 1.1**

Разработать ETL-процесс для загрузки «банковских» данных из csv-файлов в соответствующие таблицы СУБД Oracle или PostgreSQL. Покрыть данный процесс логированием этапов работы и всевозможной дополнительной статистикой (на усмотрение вашей фантазии). В исходных файлах могут быть ошибки в виде некорректных форматах значений. Но глядя на эти значения вам будет понятно, какие значения имеются в виду.

Исходные данные:

* Данные из 6 таблиц в виде excel-файлов:
  + md\_ledger\_account\_s – справочник балансовых счётов;
  + md\_account\_d – информация о счетах клиентов;
  + ft\_balance\_f – остатки средств на счетах;
  + ft\_posting\_f – проводки (движения средств) по счетам;
  + md\_currency\_d – справочник валют;
  + md\_exchange\_rate\_d – курсы валют.
* Файл «Структура таблиц.docx» – поможет создать таблицы в детальном слое DS.

Требования к реализации задачи:

* В своей БД создать пользователя / схему «DS».  
  Примеры команд:  
  <https://oracle-dba.ru/docs/architecture/schemas/basics/>

<https://postgrespro.ru/docs/postgresql/9.6/sql-createschema>

* Создать в DS-схеме таблицы под загрузку данных из csv-файлов.
* Начало и окончание работы процесса загрузки данных должно логироваться в специальную логовую таблицу. Эту таблицу нужно придумать самостоятельно;
* После логирования о начале загрузки добавить таймер (паузу) на 5 секунд, чтобы чётко видеть разницу во времени между началом и окончанием загрузки. Из-за небольшого учебного объёма данных – процесс загрузки быстрый;
* Для хранения логов нужно в БД создать отдельного пользователя / схему «LOGS» и создать в этой схеме таблицу для логов;
* (В случае реализации процесса в Talend) В зависимости от мощностей рабочей станции – сделать загрузку из всех файлов одним потоком в параллели или отдельными потоками (может не хватить оперативной памяти для Java-heap);
* Для корректного обновления данных в таблицах детального слоя DS нужно выбрать правильную Update strategy и использовать следующие первичные ключи для таблиц фактов, измерений и справочников (должно быть однозначное уникальное значение, идентифицирующее каждую запись таблицы):

|  |  |
| --- | --- |
| **Таблица** | **Первичный ключ** |
| DS.FT\_BALANCE\_F | ON\_DATE, ACCOUNT\_RK |
| DS.FT\_POSTING\_F | OPER\_DATE, CREDIT\_ACCOUNT\_RK, DEBET\_ACCOUNT\_RK |
| DS.MD\_ACCOUNT\_D | DATA\_ACTUAL\_DATE, ACCOUNT\_RK |
| DS.MD\_CURRENCY\_D | CURRENCY\_RK, DATA\_ACTUAL\_DATE |
| DS.MD\_EXCHANGE\_RATE\_D | DATA\_ACTUAL\_DATE, CURRENCY\_RK |
| DS.MD\_LEDGER\_ACCOUNT\_S | LEDGER\_ACCOUNT, START\_DATE |

Технологические требования

ETL-процесс по загрузке файлов вы можете сделать с помощью различных технологий, которые вам будут удобней. Возможные варианты технологий:

* Talend – бесплатная (для учебных целей) ETL-платформа;
* Python – для данного языка существует множество библиотек, в том числе и для работы с базами данных и с различными файлами;
* Java / Scala – для этих языков так же существует различные способы для работы с БД и файлами;
* (\*) Оркестрация процесса загрузки с помощью Airflow. Данный критерий не обязательный, но если вдруг вы сможете самостоятельно понять, установить и применить этот инструмент – это будет большим плюсом.

Требования к демонстрации работы:

* Записать видео с экрана компьютера, в котором вы демонстрируйте и комментируете в слух, то что вы делаете / уже разработали;
* Нужно продемонстрировать создание или подробно прокомментировать разработанный вами поток (ETL-процесс);
* Продемонстрировать, что поток работает – показать, что в таблице «**DS.ft\_balance\_f**» не было записей, потом запустить поток и показать, что таблица наполнилась;
* Запись в таблицы должна выполняться в режиме «Запись или замена». Поэтому не забудьте определиьт ключевые поля для возможности обновлять информацию по уже существующим записям;
* Продемонстрируйте как вы в файле «ft\_balance\_f.csv» меняете баланс для какого-нибудь <account\_rk>, показываете что в таблице «DS.ft\_balance\_f» сперва была одна сумма у этого <account\_rk> - потом запускаете ETL-процесс и показываете, что в таблице сумма обновилась;
* Это видео загрузите к себе на облако (гугл-диск, яндекс-диск и т.д.) и предоставьте доступ по ссылке;
* Приложите в домашнее задание внутри архива текстовый файл с ссылкой на ваше видео и все скрипты решения. В случае работы в Talend выгрузите и прикрепите поток.

Примечания:

* Вам в помощь будет предоставлен доступ к дополнительным лекциям на темы: Введение в ETL; Основы хранилищ данных; Работа с Talend; Основы ООП в PL/SQL; Основы написания пакетов и процедур в Oracle; Физическая организация таблиц и индексов.  
  Изучите эти лекции внимательно, они могут вам очень помочь. Для PostgreSQL основные принципы те же самые, но синтаксис может отличаться. Однако в интернете много открытой информации;
* Если вы решили, что будете применять ETL-инструмент «Talend», то лучше установите версию 7.3.1. Скачать можно отсюда (TOS\_DI-Win32-20200219\_1130-V7.3.1.exe)  
  [https://sourceforge.net/projects/talend-studio/files/Talend%20Open%20Studio/7.3.1/](https://sourceforge.net/projects/talend-studio/files/Talend Open Studio/7.3.1/)
* Для Talend обязательно понадобиться установить Java не ниже 8-й (jdk1.8.0);
* Может оказаться так, что у вас в Talend не хватает каких-то элементов (например, таймер «tSleep») – значит вам нужно будет догрузить пакет дополнительных элементов. Talend сам об этом предложи;
* Среды разработки для Python / Java / Scala можете использовать любые, какие вам удобны, главное, чтобы скрипт запускался и работал исправно;
* Допустима трансформация csv-файла в excel-файл, если вам так будет удобней.