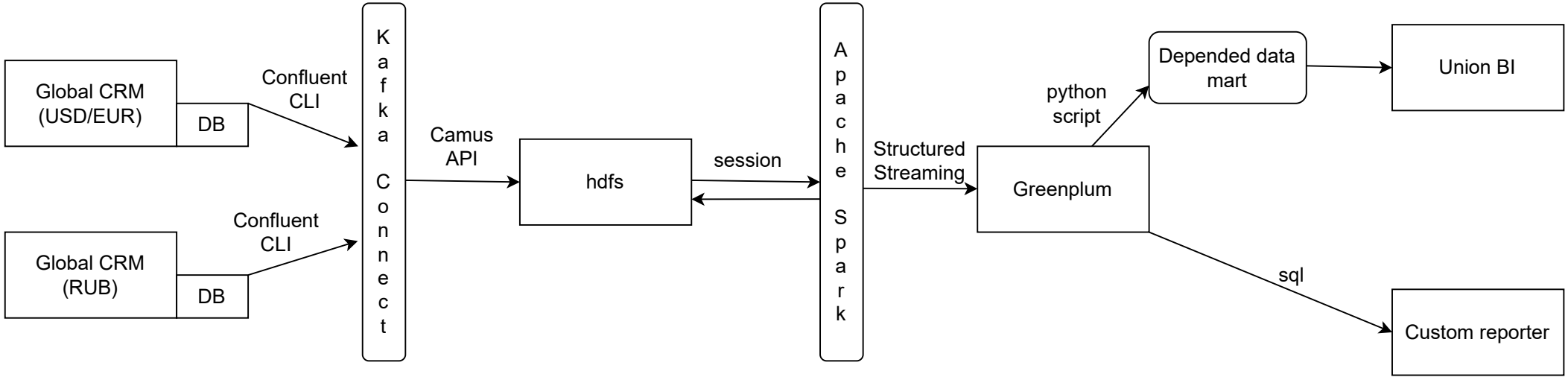

Case 2

Пусть есть 2 CRM-системы, собирающие информацию о продавце, покупателе, товаре и и стоимости.
Данные собираются с помощью kafka connect'a с confluent cli для настройки db. Далее с помощью camus api (MapReduce job'a) переливаем сырые данные в hdfs.
Благодаря спарку преобразуем данные из hdfs и сохраняем их в Greenplum. Далее есть 2 варианта:
1) Использование для шаблонного/постоянного анализа - зависимая от общего хранилища данных витрина данных к которой обращается BI.
2) Получение произвольного отчета, несмотря на не высокую скорость выполнения запроса (если нужно не часто то можно и подождать).

Холодное хранилище - hdfs.
Теплое - postgres.
Горячее - data mart под bi. (в случае custom reporter'a не нужен в силу малого спроса).

Решаемые проблемы кейса:
1) Локальность и глобальность CRM, разные валюты, разная структура данных.
Решается на стадии обработки данных с помощью spark. Предлагаемая общая структура (не схема greenplum): {страна, валюта, компания, услуга/товар, стоимость, глобальный id, локальный id}.
2) Иерархия рабочих и финальных версий.
Рабочие данные - теплые данные в postgres, финальная - data mart (hot), с обновлением раз в N дней (python script).