**Запуск пайплайна в Streamsets**

1. Настройка системы под домашнее задание.

Так как до момента выполнения ДЗ у меня стояла "чистая" Windows 10, к его выполнению добавились пункты по

настройке системы для беспребойной работы контейнеров. На курсе по DE, скорее всего уже предполагалось, что у всех всё настроено.

Думаю данное краткое руководство могло бы пригодиться тем, кто оказался в такой же ситуации. В данном пункте будут перечислены источники и действия, которые помогут решить

данный вопрос.

- Прежде всего необходимо установить ядро Linux в своей системе, так как команды make, sudo, как бы мы ни хотели,

не работают в консоли Windows 10, хоть и с ее помощью можно запустить контейнеры. Настройка и установка ядра пошагово

описаны здесь [https://docs.microsoft.com/ru-ru/windows/wsl/install-win10#step-4---download-the-linux-kernel-update-package](https://docs.microsoft.com/ru-ru/windows/wsl/install-win10%23step-4---download-the-linux-kernel-update-package).

- До текущего курса обучения я никогда не работала с докером, поэтому мне пришлось потратить немного времени на литературу по нему.

Далее настройка и проверка корректности Docker - <https://docs.docker.com/get-started/>

- Было досадно узнать о том, что всего этого недостаточно, так как команды make, sudo и т.д. не работали в консоли Kali Linux (остановила выбор на ней).

Но для этого необходимо было настроить систему, установить различные пакеты, бибилотеки. Источник: <https://losst.ru/nastrojka-kali-linux-posle-ustanovki>

Итог: "подсистема" Kali Linux сейчас "живет" вместе с Windows 10 на ноутбуке и пока работает без проблем.

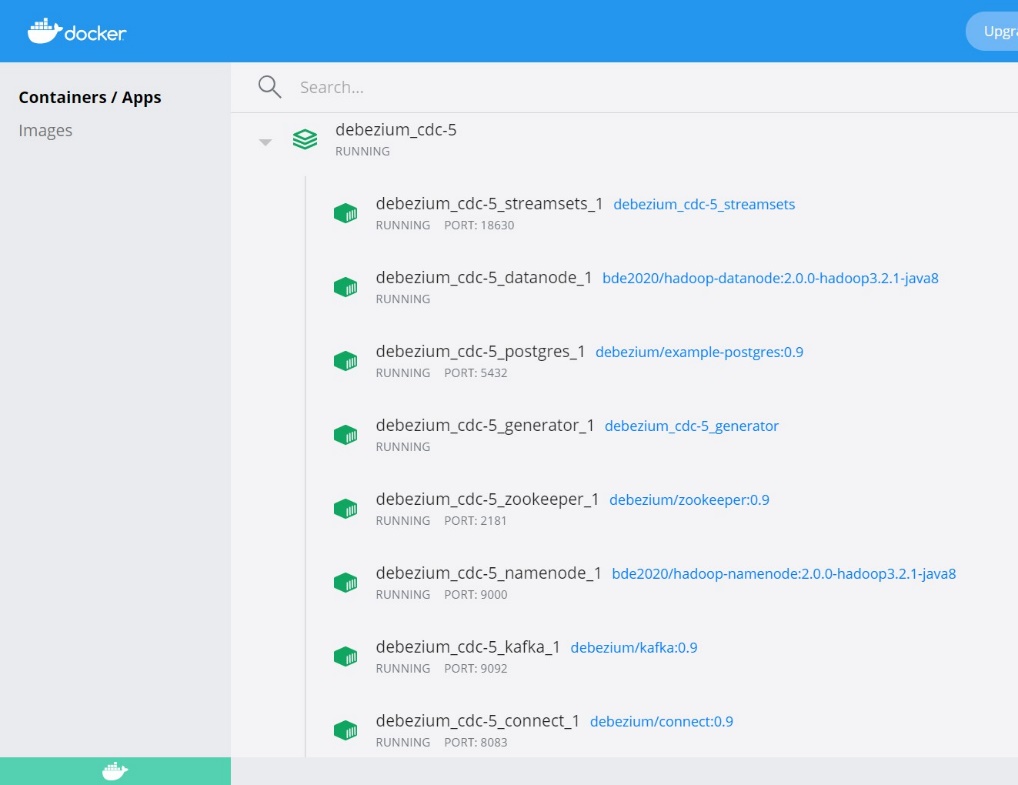
2. Запуск контейнеров по ДЗ.

Через Visual Studio Code клонируем репозиторий https://github.com/Gorini4/debezium\_cdc/ именно в систему Kali Linux.

Docker Desktop при этом запущен. Заходим в терминал и в папке debezium\_cdc вводим команду: docker-compose up -d

(-d необходим для того,чтобы увидеть "чистую" информацию о запуске контейнеров). Далее появляется в папке streamsets папка sdc-data. Переходим в нее и

даем права на эту папку: sudo chmod 777 sdc-data. Запускаем снова docker-compose up -d, чтобы запустились уже все контейнеры. Проверяем в Docker Desktop.

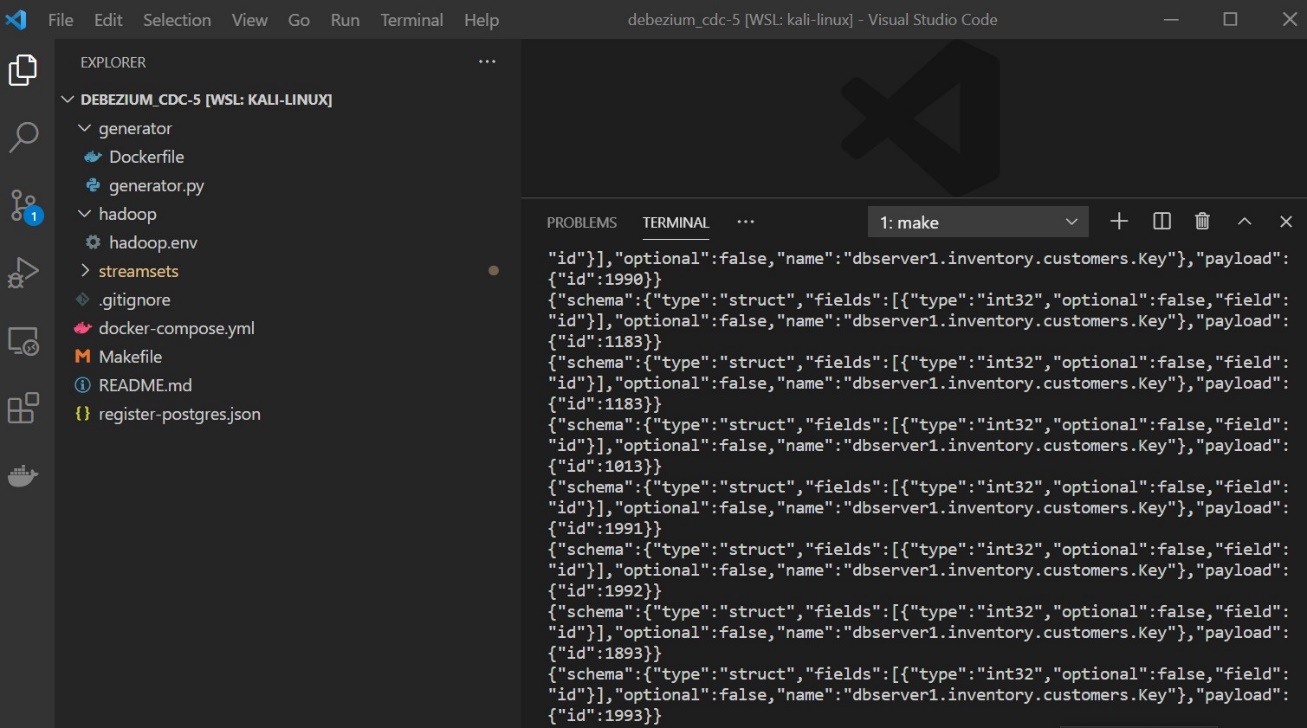


Всё запустилось. Идем дальше.

3. Регистрируемся в Postgres и запускаем consumer.

В папке debezium\_cdc вводим команду make register-postgres и запускаем make create-consumer.

поучаем наименование топика и генерируется схема с данными.



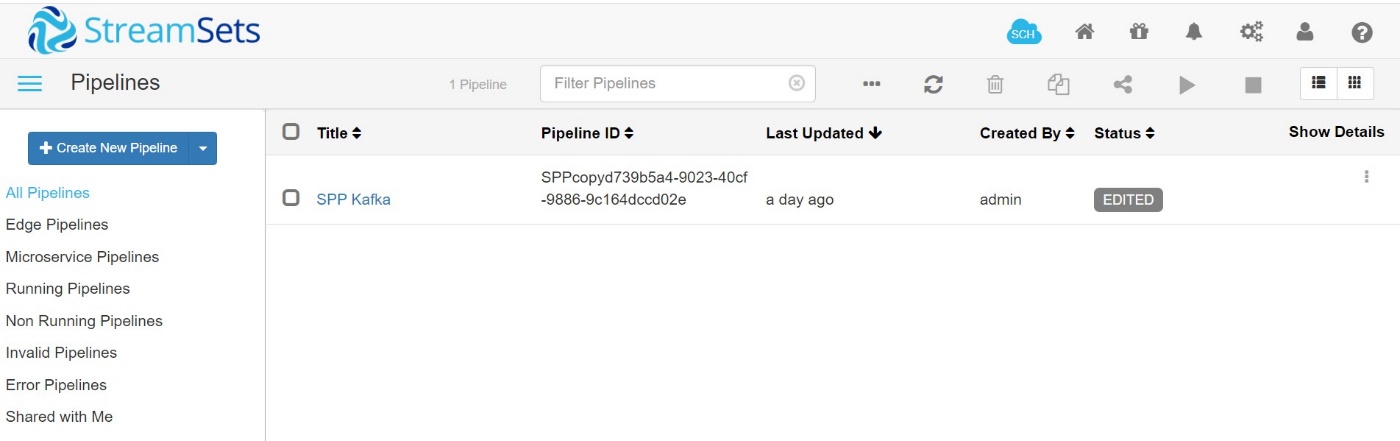
4. Запуск и тестирование streamsets

Заходим в streamsets через localhot:18630. Вводим логин/пароль admin/admin. Create new pipeline. На всякий случай протестировала лекционный вариант:

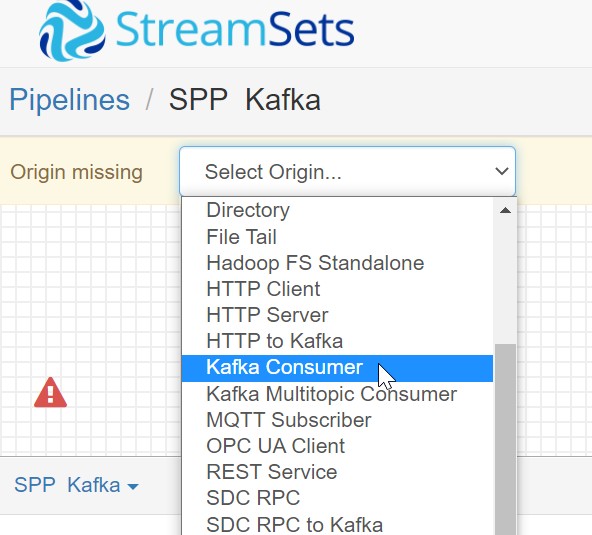
<https://github.com/Gorini4/divolte-streamsets-quickstart>. Всё сработало корректно. Теперь переходим к варианту из ДЗ.

4. Создание Pipelyne в Streamsets по репозиторию из ДЗ.

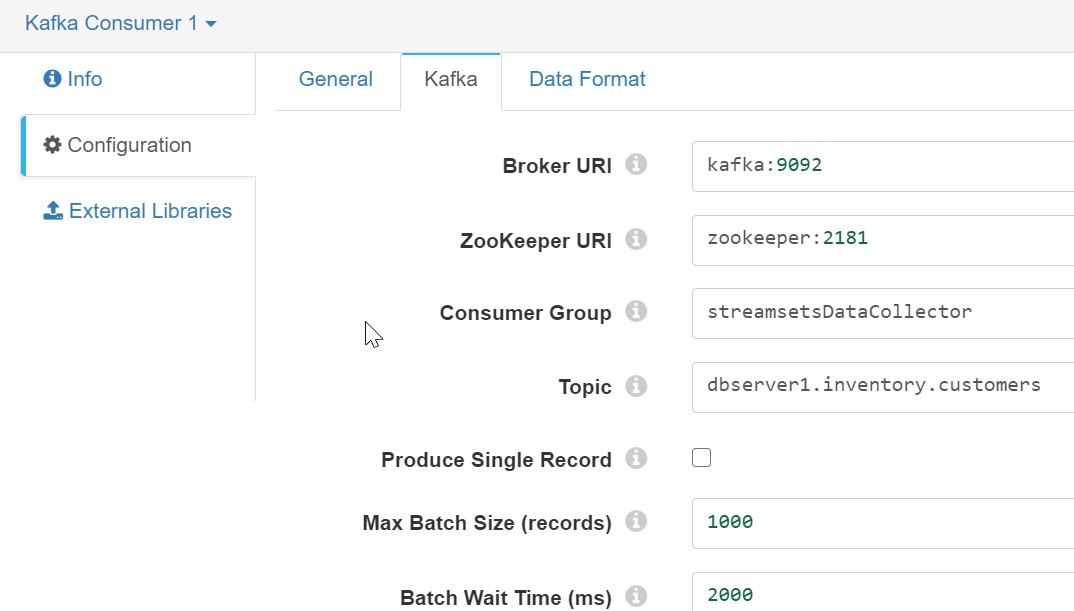
- даем название "SPP Kafka"



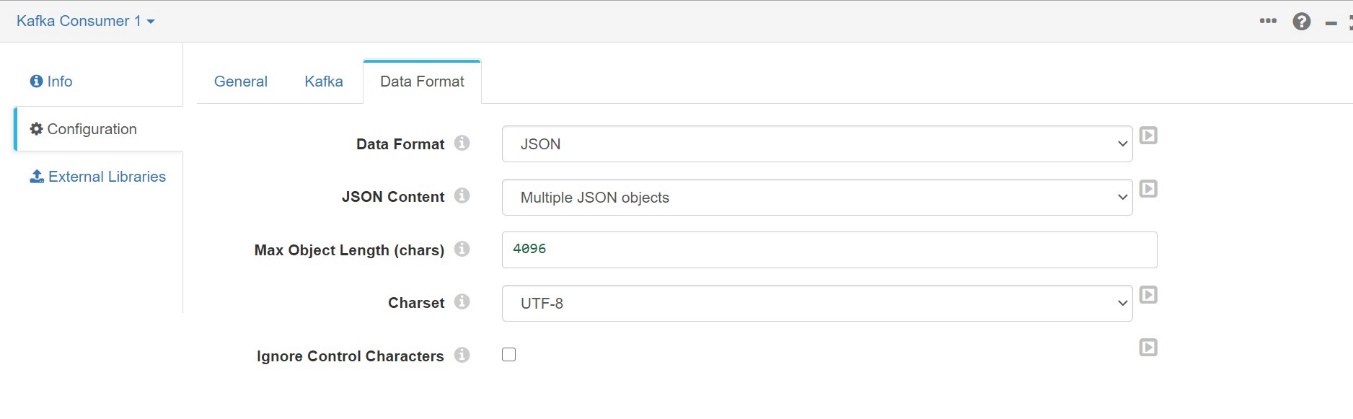
- выбираем источник. Kafka Consumer.

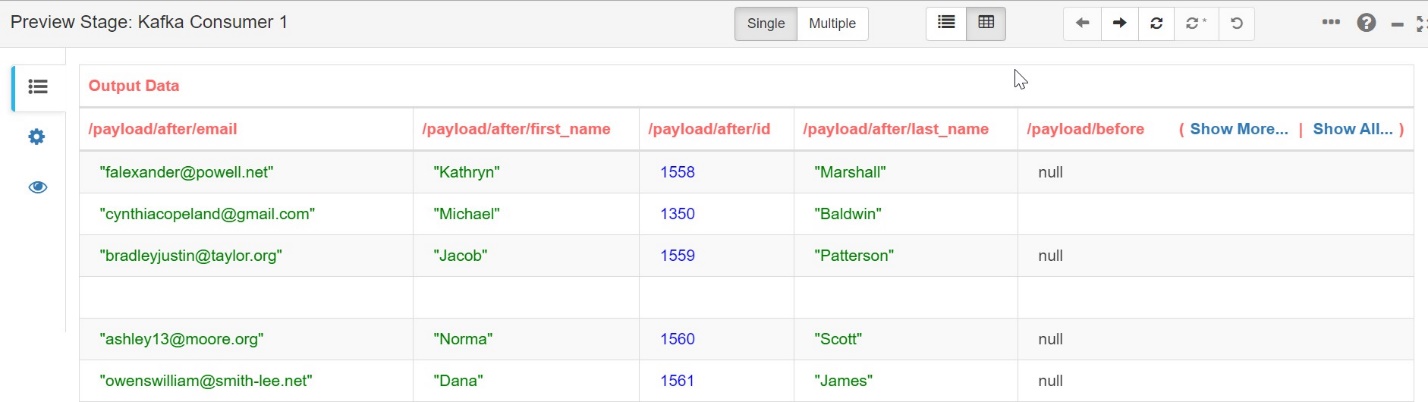


- Настраиваем источник (порты).



- Настраиваем формат исходных данных. Мы получаем данные в JSON.



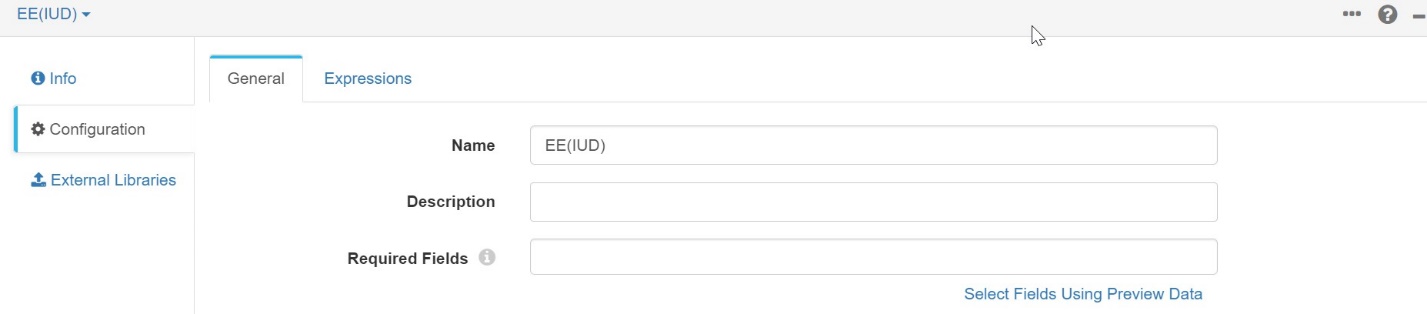


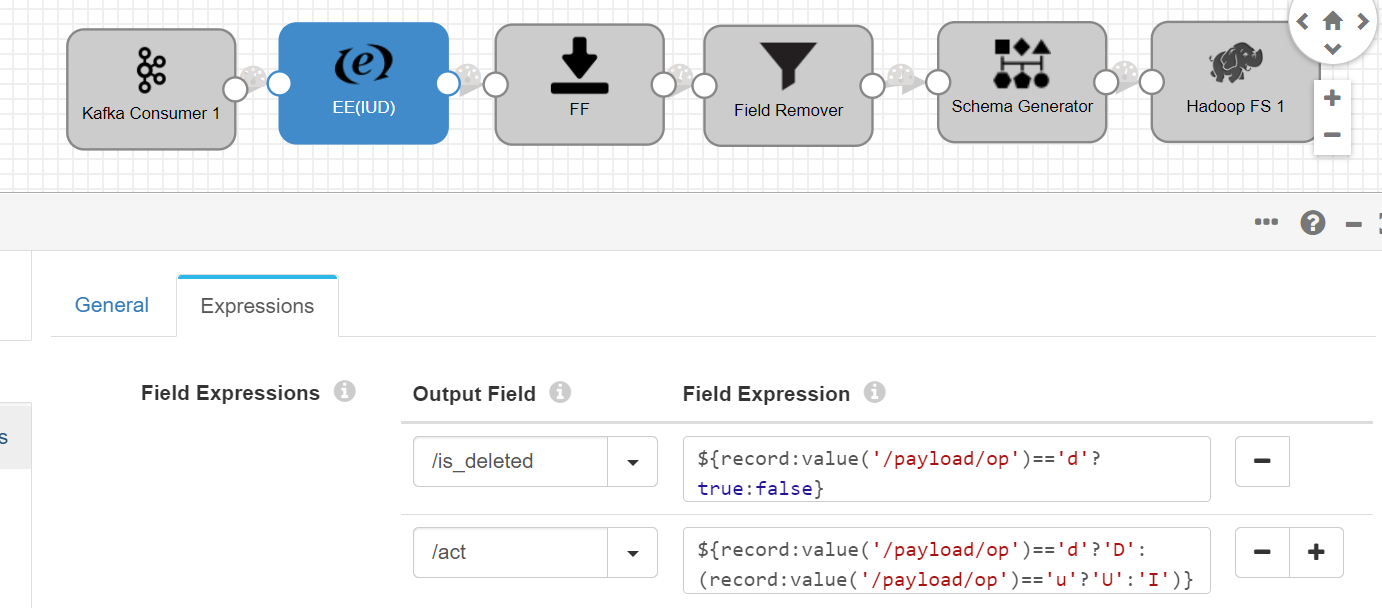
Очень удобно просматривать из в табличном виде в Preview. Тестируем с

Trash и переходим к настройке перед загрузкой в Hadoop.

- Добавляем поля для контроля операций над исходными данными: удаление, обновление, вставка. Для этих целей используем процессор Expression Evaluator.

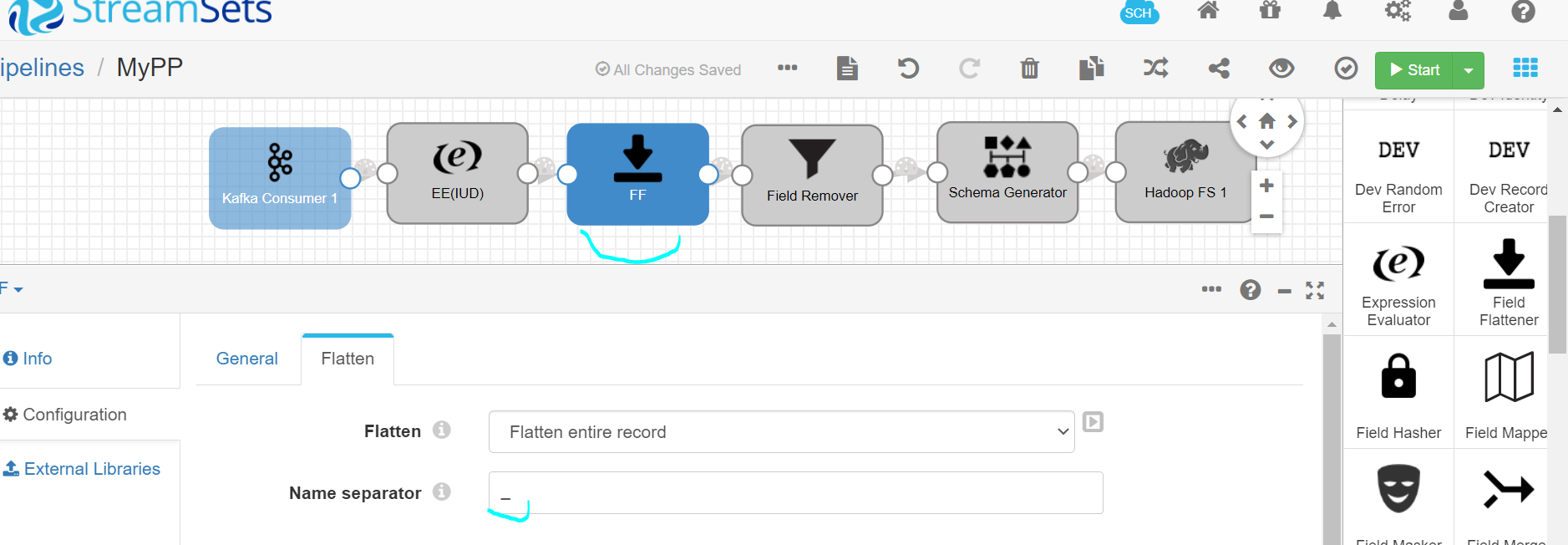
Даем ему имя,



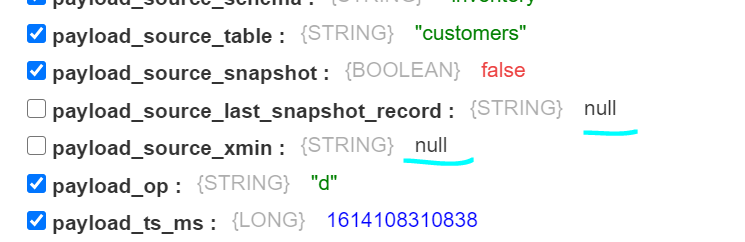


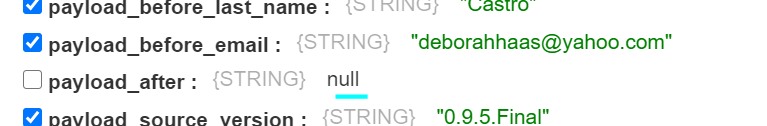
и настриаиваем новые поля.

- Для перевода в "плоскую" структуру для Avro схемы воспользовалась Field Flatenner. Разделитель наименований (иерархии) "\_" выбран неспроста, так как другие символы (/ или .) либо обрастают апострофами, мешающими распознавать имена колонок, либо в принципе не распознаются в генераторе схемы.

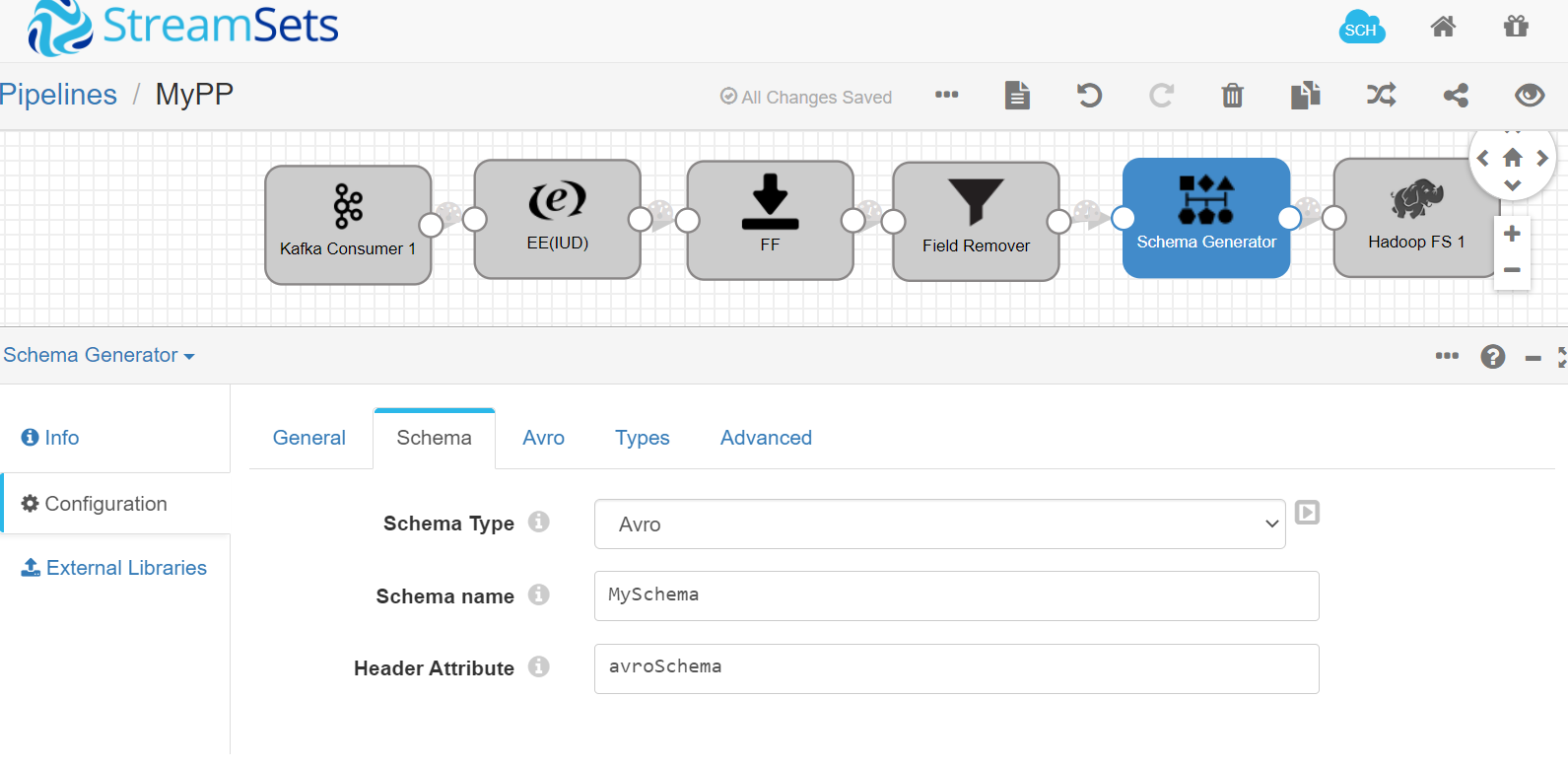


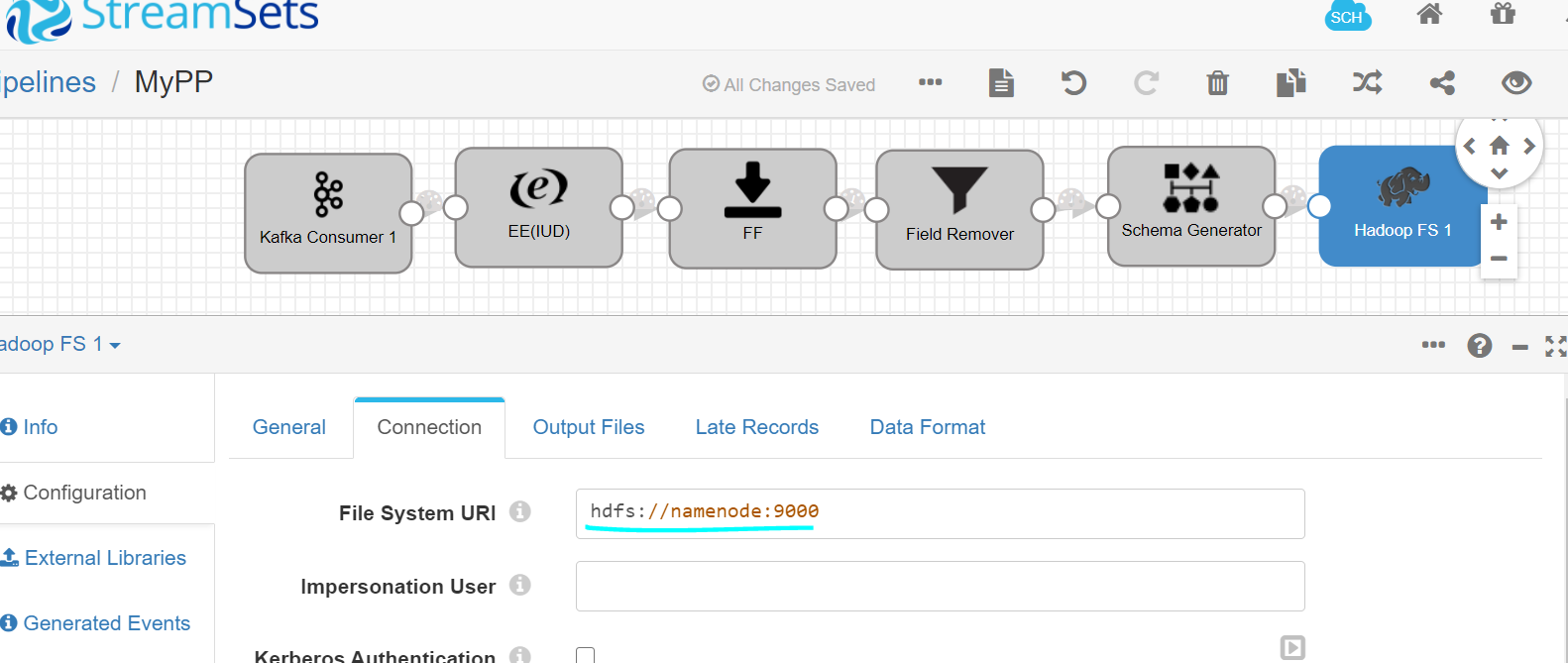
- Некоторые строки в схеме содержат NULL и дают ошибки при выгрузке данных в пункт назначения. Однако очистка от всех нулевых значений и пустых строк не дала результатов, поэтому поля с нулевыми значениями были всё же исключены (3 шт.) в процессоре Field Remover.

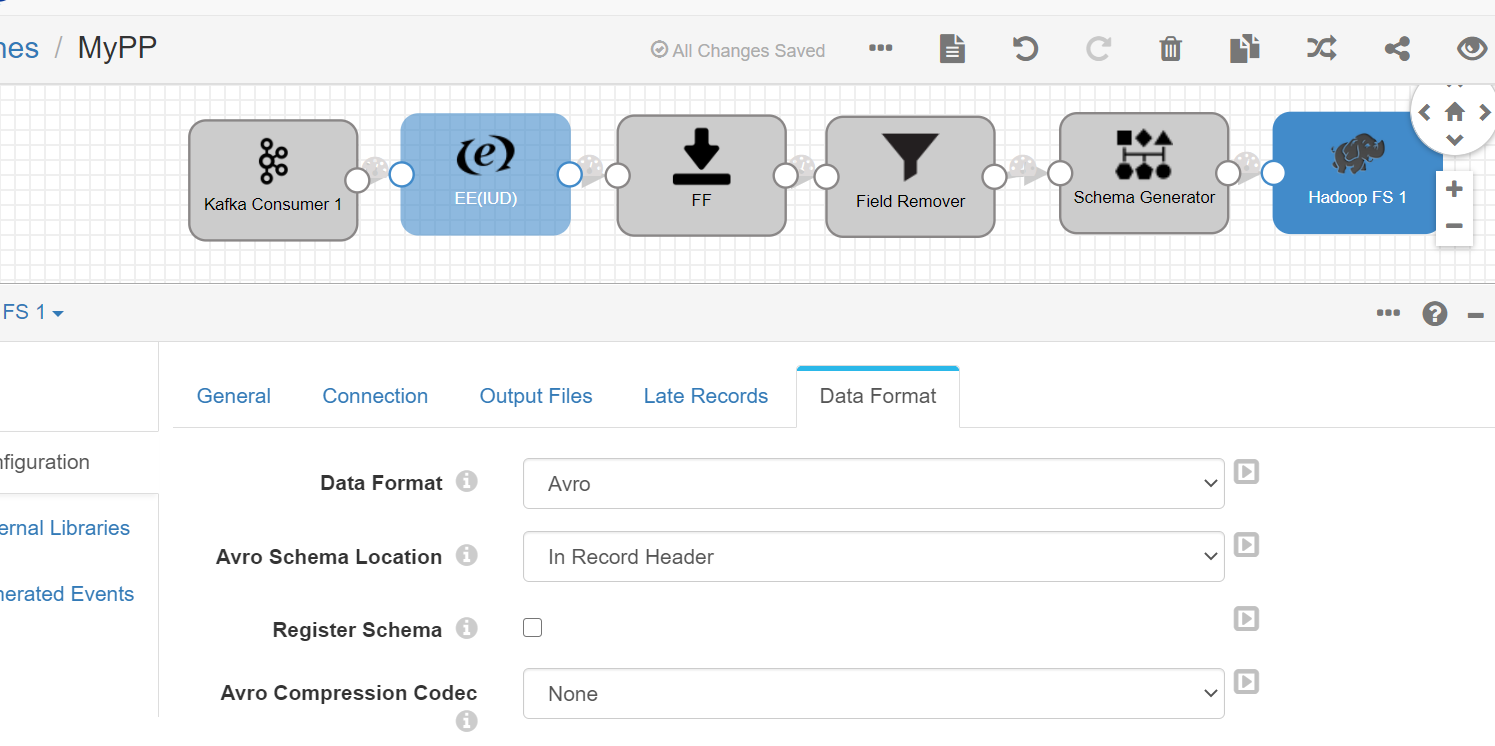




Была предпринята попытка использовать для исключения или обработки нулевых значений Field Mapper, но так и не удалось решить эту проблему (очень много времени ушло на настройку и обработку других этапов). Думаю, что в конце концов удастся найти решение.

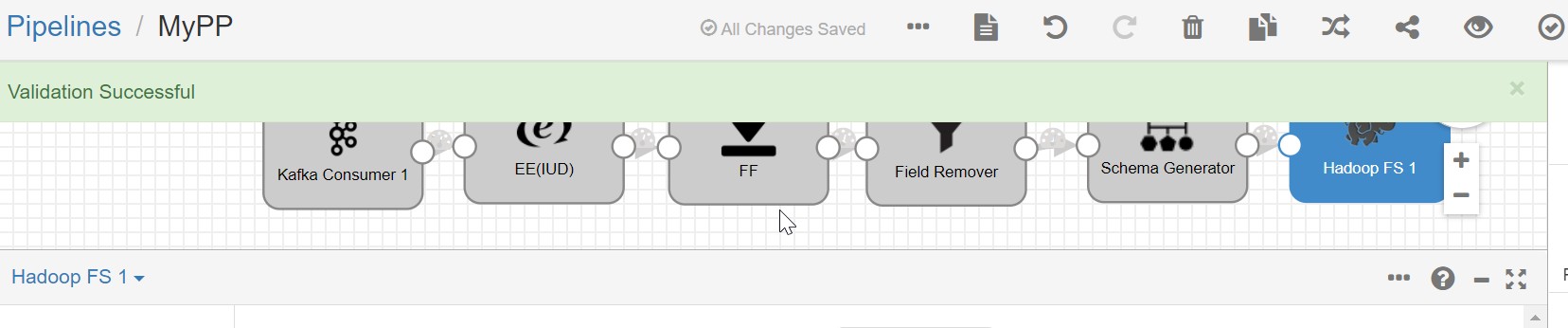
- Подготовка данных в формате Avro (без готовой схемы) требует их подготовки через генератор схемы. 

- Перед загрузкой в Hadoop FS заполняем поле URI.

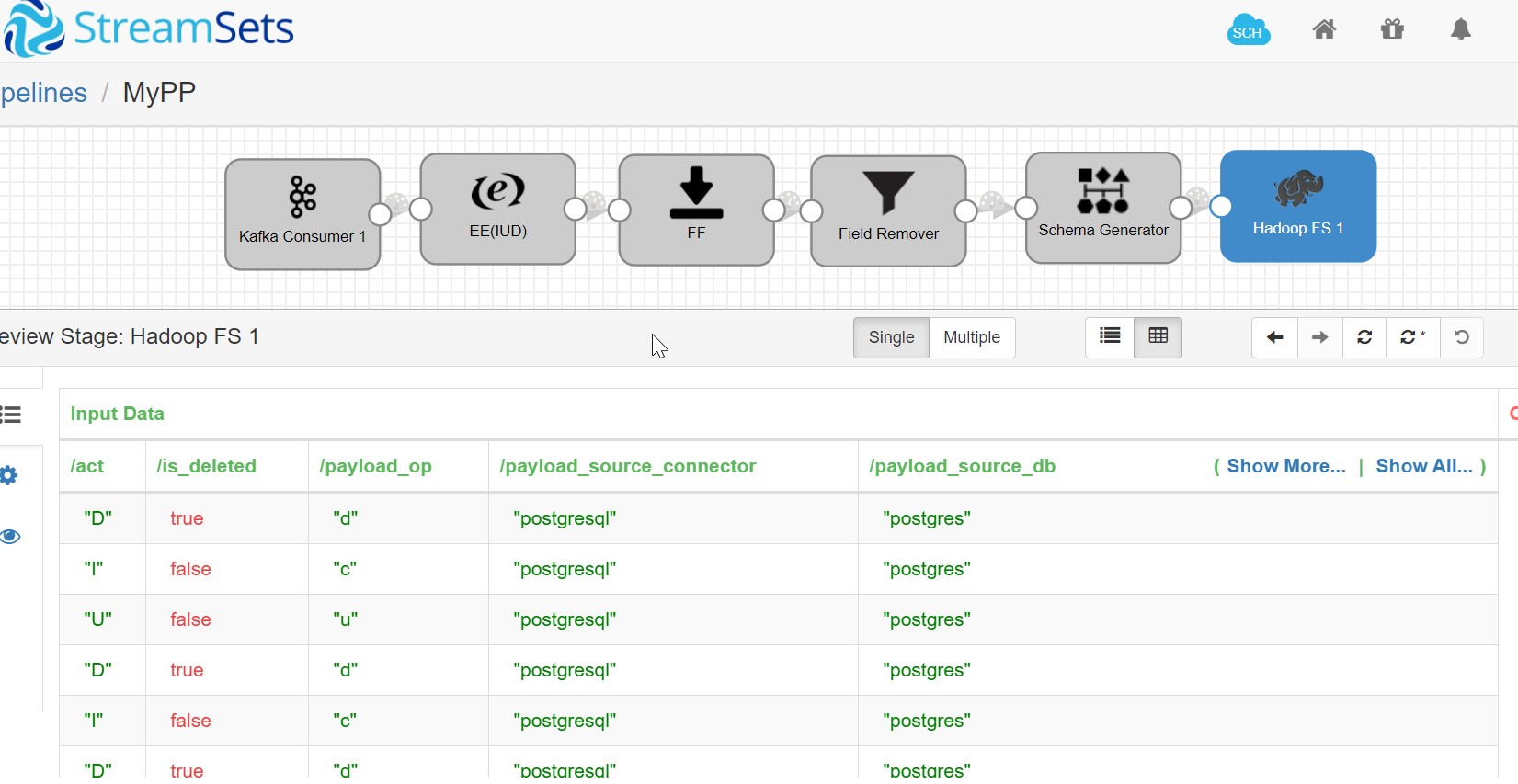
Прописываем формат и схему данных. 

5. Проверка и запуск.

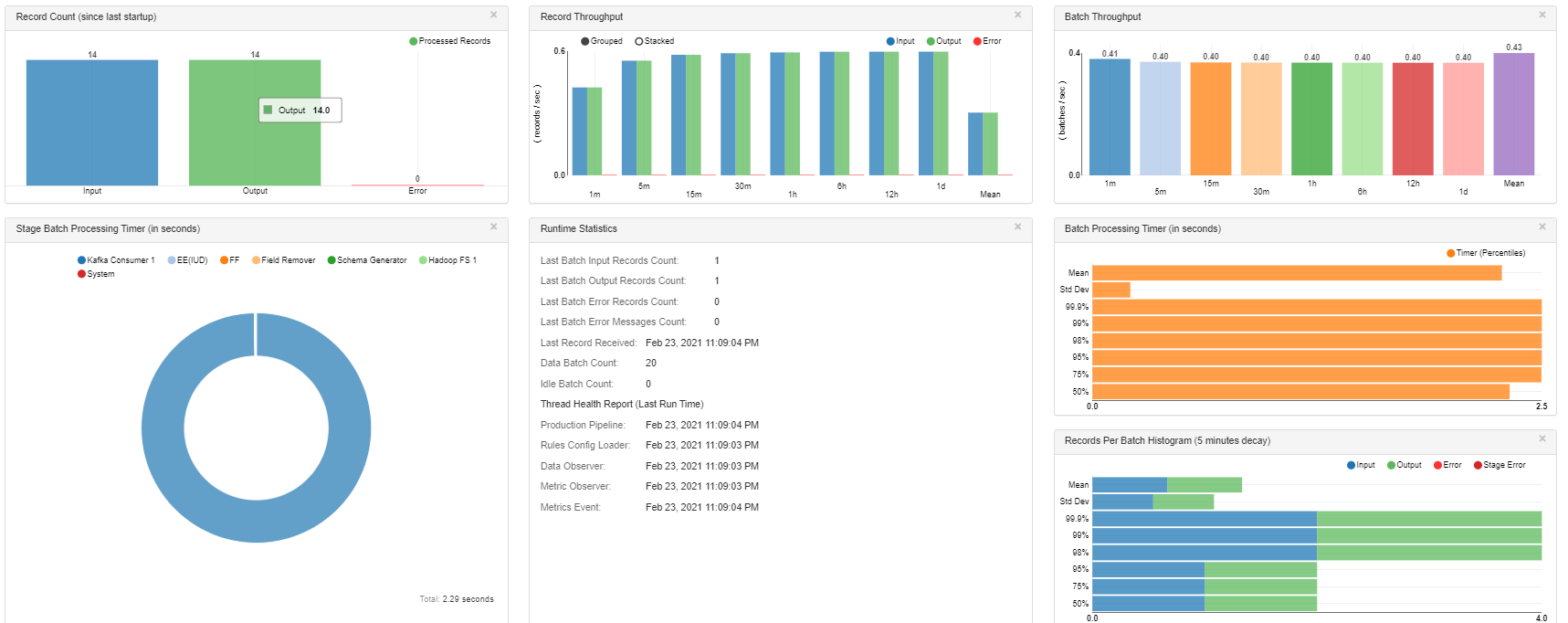
Проверяем валидацию.

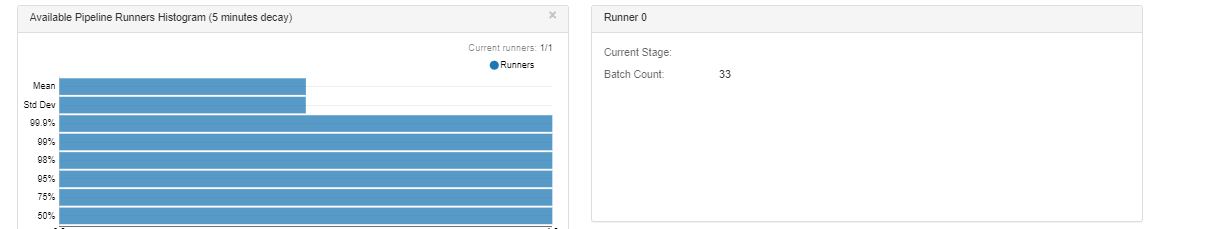


Preview и смотрим табличную схему.



Запускаем пайплайн и смотрим аналитику.





Проверяем наличие ошибок.

