Инструкция по работе с интеграцией с брокером сообщений Kafka

# Назначение

Инструкция предназначена для ознакомления с шаблоном разработки «Интеграция с брокером сообщений Kafka».

# Настройки

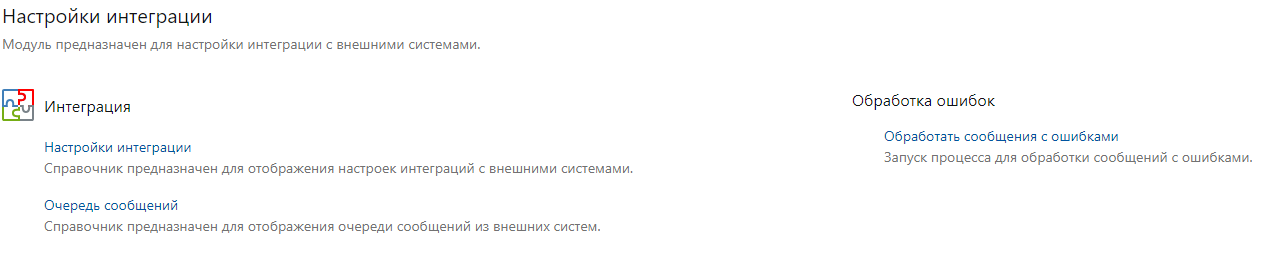
Настройки решения доступны пользователям, входящим в системную роль «Администраторы».

# Общее описание

## Решение позволяет гибко настраивать подключение к Kafka, настраивать объекты на конкретные топики, детализировать на входящие или исходящие сообщения. Получать сообщения из брокера Kafka и записывать в справочник «Очереди сообщений», для дальнейшей обработки на стороне Directum RX, отслеживать статус обработки сообщений, распараллеливать процесс обработки (Одно сообщение – один АО).

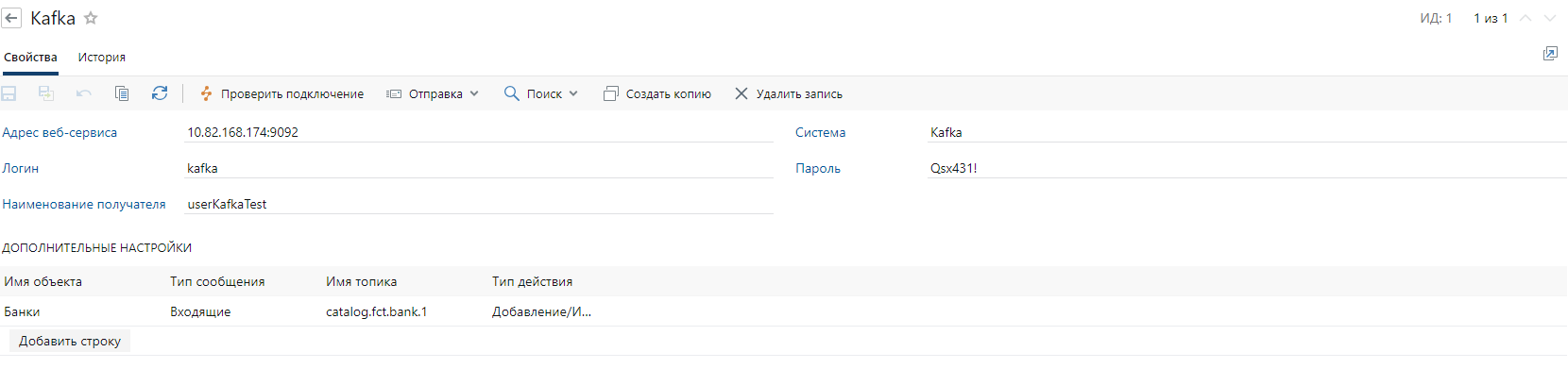
## Представление модуля

В представлении модуля по умолчанию представлены ссылки на справочники модуля.



## Создание настроек для подключения к Kafka

Для создания настроек подключения к Kafka, а также топиков для принимаемых сообщений, необходимо создать запись справочника «Настройки интеграции».



Необходимо заполнить свойства:

* Система.
* Адрес веб-сервиса.
* Логин (в шаблоне не используется).
* Пароль (в шаблоне не используется)
* Наименование получателя (Consumer)

Дополнительные настройки:

* Имя объекта
* Тип сообщения
* Имя топика
* Тип действия

## Серверные функции

IntegrationSettings

* **HandleErrorRecordsServer(string objectName) -** Отправить сообщения с ошибками на обработку.

KafkaIntegration

***CreateKafkaConnector(IntegrationSettings.IConnectSettings settings)*** - Создать экземпляр подключения Kafka.

***CheckConnectToService(string topicName)*** - Проверить подключение к Kafka сервису.

***GetCountryByAlphaCode3(string alphaCode3)*** - Получить страну по альфа коду 3.

***GetSynchBank(string externalLink, string bic, string swift, bool nonResident)*** - Получить синхронизированный банк.

***GetCompanySynch(string externalLink, string tin, string trrc)*** - Получить синхронизированную организацию.

***GetPersonSynch(string externalLink, string tin)*** - Получить синхронизированного физического лица.

***GetPersonByGuid(string externalLink)*** - Получить персону по внешней ссылке.

***WriteToLog(string message, bool isError)*** - Записать сообщение в лог-файл.

***SetCounterpartyInfo(KafkaIntegration.Structures.Module.ICounterpartiesFromKafka counterpartyInfo, IKafkaQueueItem queueItem)*** - Заполнить данные контрагента.

***SetBankInfo(KafkaIntegration.Structures.Module.IBanksFromKafka bankInfo, IKafkaQueueItem queueItem)*** - Заполнить данные банка.

# Варианты расширения функциональности на проектах

1. В справочнике Настройки интеграции расширять способы авторизации (SSL).

2. Расширение функционала для загрузки / отправки кастомных объектов;

3. Расширение функции логирования.

Настройки окружения

Особенности NativeClient Conflument.Kafka – необходимо прописать настройки 127.0.0.1 kafka в файлах /host/etc.

Использование на Linux

Для использования в системе DirectumRx, развернутом на ОС Linux, необходимо передавать библиотеку librdkafka.so в контейнер Веб сервера и Generic:

1. Библиотеку добавить в папку с образом контейнера /etc/\_builds/Platform/web/Server

2. В файле развертывания докера добавить В dockerfile надо добавить:

ADD librdkafka.so /lib64/

RUN ln -s /lib64/libdl.so.2 /lib64/libdl.so

RUN ln -s /lib64/libsasl2.so.3 /lib64/libsasl2.so.2

Необходимо учесть, что при обновлении системы, файлы изменяются на эталонные, необходимо повторно выполнить данные действия.