

# Eventbus. Холдинг

Руководство пользователя

# Содержание

Содержание	2
1. Об этом документе	3
2. О программном комплексе	4
2.1. Назначение и цель	4
2.2. Функциональные характеристики компонентов	4
2.2.1 ui	4
2.3. Поддерживаемые сценарии интеграции	4
2.4. Поддерживаемые типы данных и форматы полей	5
2.5. Структура интеграционного объекта	5
2.6. Структура топика	6
3. Системные требования	8
3.1. Требования для ПК пользователя	8
4. Использование на персональном компьютере	9
5. Подключение целевой системы к Apache Kafka	26

# 1. Об этом документе

Данное руководство содержит пошаговые инструкции об использовании программного комплекса EventBus. Холдинг» на персональном компьютере.

## 2. О программном комплексе

### 2.1. Назначение и цель

«Eventbus. Холдинг» представляет собой программный комплекс, состоящий из нескольких программных компонентов, каждый из которых имеет собственное назначение, объединенный единым веб-интерфейсом, предоставляющим пользователю доступ к использованию всех функциональных возможностей программного комплекса (далее – Программный комплекс)

Программный комплекс разработан для создания интеграционных объектов (мета-информации и описания структуры данных), создания сценариев интеграционных потоков для взаимодействия систем и сервисов на предприятии. Программный комплекс спроектирован с возможностью эффективного вертикального и горизонтального масштабирования.

Основными задачами данного программного комплекса являются упрощенный способ создания интеграционных потоков, упрощение просмотра состояния интеграции, ускорение реализации взаимодействия между системами за счет генерации кода и подготовленных SDK.

Программный комплекс представляет собой корпоративное решение, установка и настройка которого осуществляется в соответствии с Руководством администратора. После развертывания программного комплекса, его использование осуществляется при помощи веб-интерфейса в соответствии с настоящим Руководством пользователя.

Основными пользователями программного комплекса являются сотрудники, обладающие квалификацией “инженер-программист”.

### 2.2. Функциональные характеристики компонентов

Для взаимодействия с программным комплексом со стороны пользователя используются следующие компоненты:

#### 2.2.1 ui

Пользовательский веб-интерфейс, в котором авторизованным пользователям предоставляется возможность конфигурировать и описывать интеграционные объекты, создавать интеграционные сценарии между системами, просматривать и контролировать состояние интеграции.

### 2.3. Поддерживаемые сценарии интеграции

Формат данных, используемый в программном комплексе - JSON.

- Входящий сценарий через “proxy” - система отправляет через протокол HTTP в компонент proxy, который, в свою очередь, проверят присланные данные и доставляет их до “Apache Kafka”
- Исходящий сценарий по webhook - компонент “callback” получает данные из “Apache Kafka”, и, по настройкам, осуществляет доставку до целевой системы через HTTP вызов
- Исходящий сценарий через “Apache Kafka” - целевая система самостоятельно подключается к “Apache Kafka” и производит обработку данных. Можно использовать различные варианты подключения, например:
  - Через готовое SDK для языков java и python
  - Через сторонние библиотеки
  - Через готовые фреймворки (например, Kafka Connect)

## 2.4 Поддерживаемые типы данных и форматы полей

Для описания структуры данных интеграционного объекта используется спецификация JSON Schema 2020-12. Программным комплексом поддерживаются следующие типы данных и ограничения, которые будут учитываться при проверки данных на корректность:

- UUID
- Строка, с возможностью указания минимального и максимального размера
- Целое число (размерностью 64 бита), с возможностью указания минимального и максимального значения
- Дробное число (размерностью 64 бита), с возможностью указания минимального и максимального значения
- Логическое значение
- Дата-время (в формате ISO 8601)
- Дата (в формате ISO 8601)
- Время (в формате ISO 8601)
- Email (с возможностью указания максимального и минимального размера)
- Продолжительность (в формате ISO 8601)
- Хост, IPv4 и IPv6 адреса
- Ссылка и ее составные части (например, домен, путь и т.д.)
- Произвольное регулярное выражение, например, для реализации проверки на номер телефона

## 2.5 Структура интеграционного объекта

Интеграционный объект состоит из полей, которые образуют разные уровни вложенности, и при этом, поля интеграционного объекта могут иметь особые требования к заполнению. Правила следующие:

- У интеграционного объекта есть возможность указать первичный ключ - это обязательно хотя-бы для одного поля на первом уровне вложенности. Такие поля всегда должны передаваться и быть заполненными
- Можно указать, что поле является обязательным. Такие поля должны быть

переданы в JSON запросе

- Nullable поля, означает, что значение поле может быть не заполнено

Для организации многоуровневой структуры интеграционного объекта, добавлены комплексные типы данных - объект и массив. При их добавлении, появляется новый уровень, позволяющий добавлять новые поля. Варианты комбинации видов полей:

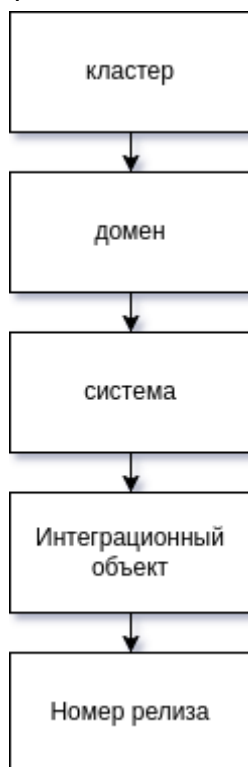
- объект - примитивный тип
- объект - объект
- объект - массив
- массив - примитивный тип
- массив - объект

## 2.6 Структура топика

Топик в "Apache Kafka" реализован с учетом "multi-tenancy" и его имя формируется из следующих составных частей:

- кластер "Apache Kafka" - (их может быть много, для разных сред и потребителей)
- домен данных
- имя системы
- имя объекта
- номер релиза интеграционного объекта

Порядок выглядит следующим образом:





## 3. Системные требования

### 3.1. Требования для ПК пользователя

Как указывалось выше, пользователь взаимодействует с программным комплексом посредством веб-интерфейса, в связи с чем особых системных требований к ПК пользователя не предъявляется. Требования:

- Операционная система Windows с версией 7 и выше
- Веб-браузер
  - Chrome 57+
  - Edge 16+
  - Firefox 54+
- Доступ в интернет (или интранет, в зависимости от среды развертывания, которую выберет заказчик)



## 4. Использование на персональном компьютере

Для доступа в систему, обратитесь к администратору организации за учётными данными - ссылкой на веб-интерфейс, логином и паролем.

Откройте веб-браузер и перейдите по предоставленной ссылке (веб-приложение event-bus). В первый раз вас переадресует на страницу входа(ссылки на приведенных ниже рисунках указаны в качестве примера):

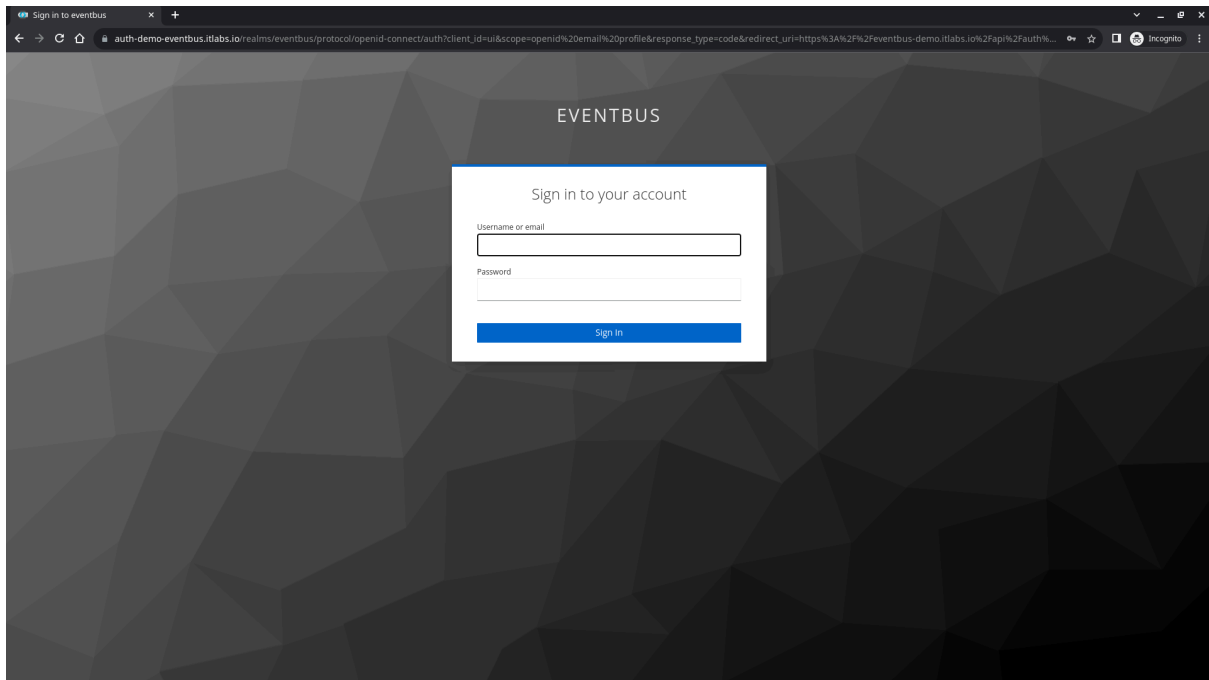


Рисунок 4. Форма авторизации

Выполните вход под своими учетными данными, предоставленными Вам администратором организации.

После успешной авторизации, вы попадаете на страницу с информацией о вас, на которой доступна информация:

- как о члене команды (команд) разработки и вашей ролью в команде, периоде нахождения в команде
- наборе продуктов, к которым вы относитесь
- мои системы / сервисы, к которым у меня есть доступ на создание сценария интеграции
- мои интеграционные объекты - состав объектов, которые созданы и пользователь может управлять их ЖЦ

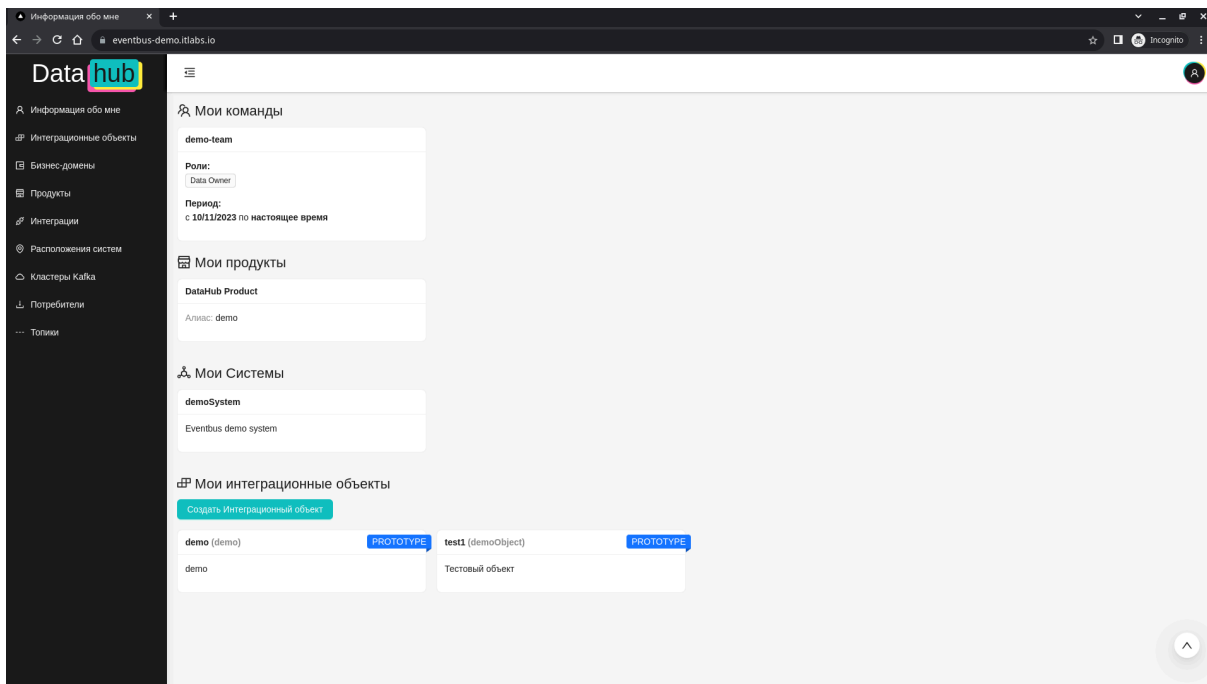


Рисунок 5. Информация о доступных сотруднику объектов

Бизнес-домен - сущность отражающая подход DDD (Domain-driven design - далее по тексту - DDD) для разбиения и группировки интеграционных объектов по смыслу и принадлежности к конкретной области действия предприятия. На странице доступен список доменов, с возможностью фильтрации по имени.

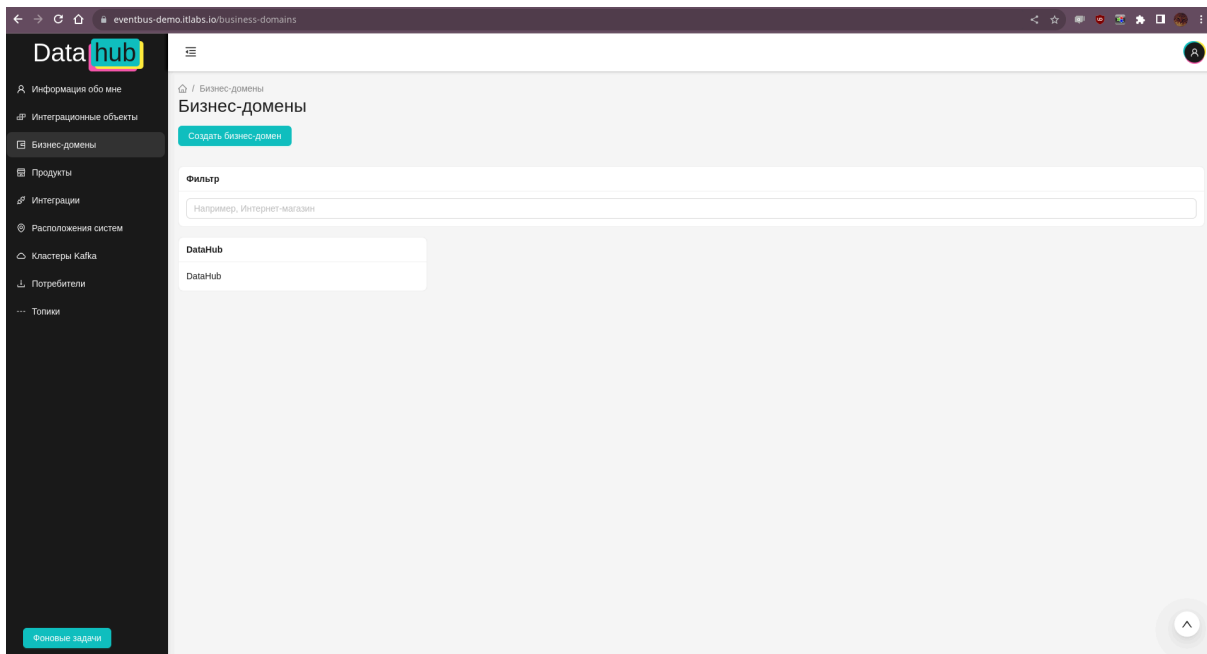


Рисунок 6. Страница с бизнес-доменами

Создавать новые домены может сотрудник с ролью dh-integration-admin или dh-admin. Для создания, необходимо кликнуть на кнопку "Создать бизнес-домен", произойдет переход на форму создания.

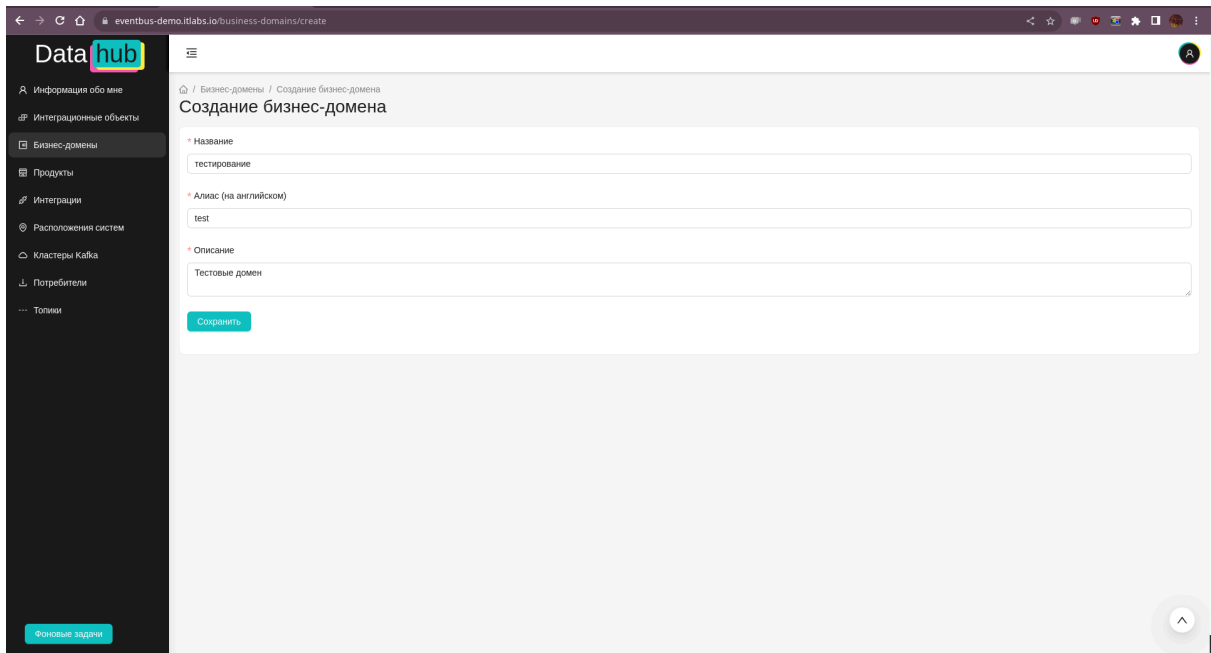


Рисунок 7. Создание бизнес-домена

Необходимо заполнить 3 обязательных поля:

- Название - по полю производится поиск в веб-приложении
- Алиас - наименование на английском языке, которое не изменяется и участвует в формировании имени топика
- Описание - расширенное описание, помогающее другим сотрудникам понять контекст

Следующим пунктом необходимо создать сущность продукт. Продукт олицетворяет программный комплекс, который является смысловых источников для части интеграционных объектов в рамках домена данных. Важно - продукт относится только к одному домену, поэтому необходимо корректно прорабатывать доменную модель.

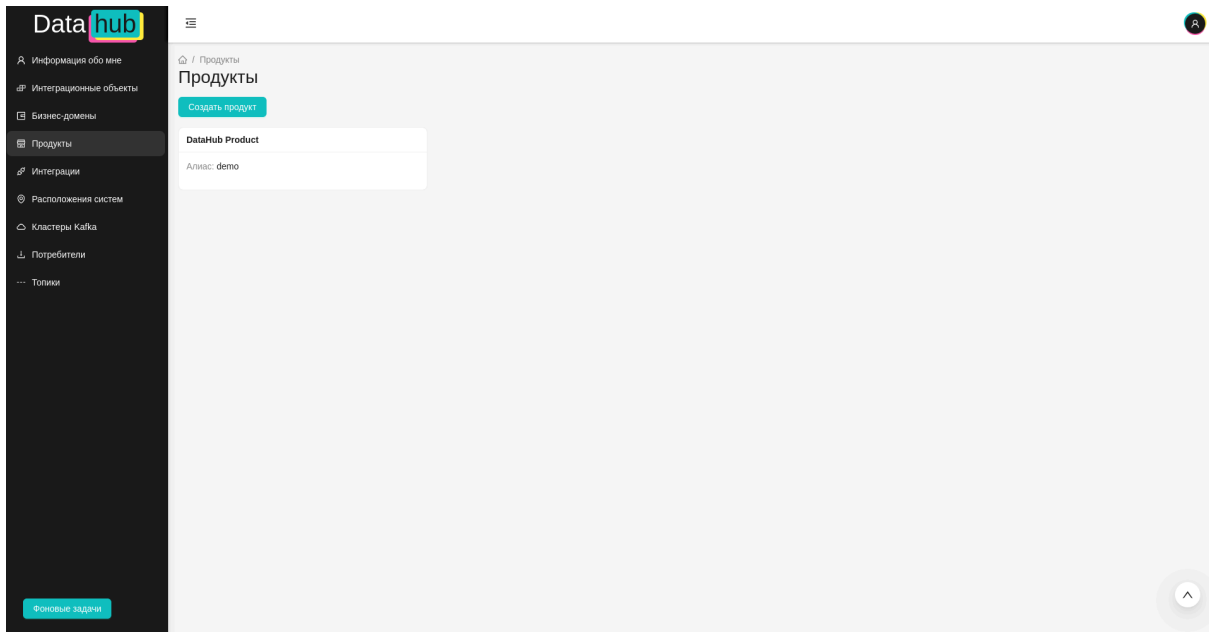


Рисунок 8. Список продуктов

Для создания продукта, нужно перейти на страницу “Продукты” и нажать на кнопку “Создать продукт”.

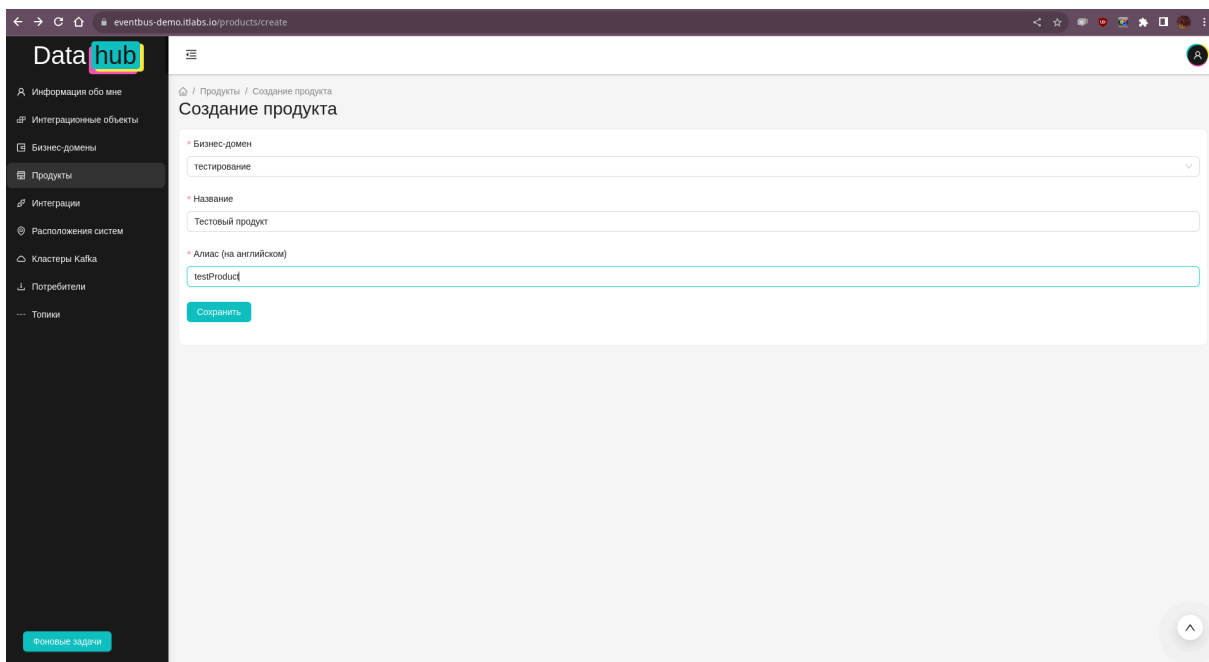


Рисунок 9. Форма создания продукта

После создания продуктов, произойдет возврат на страницу продуктов.

Теперь можно приступить к созданию интеграционного объекта и его структуры. На странице интеграционного объекта можно производить фильтрацию по имени, продукту, авторству и статусу. О статусной модели более детально будет описано позднее.

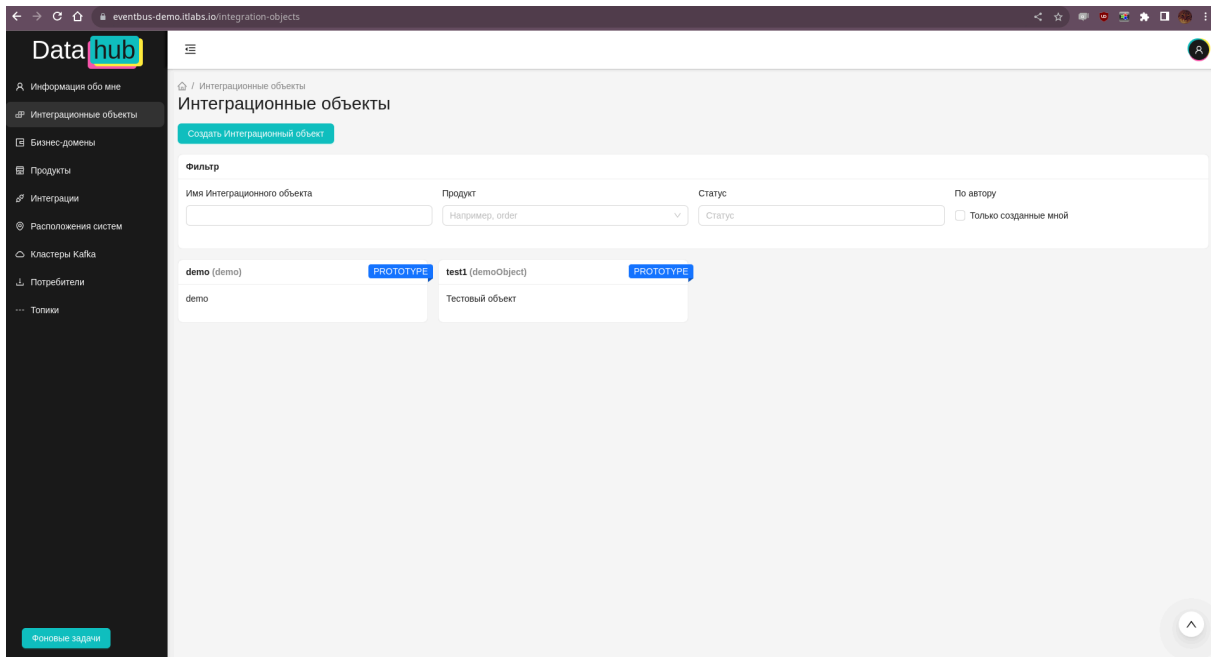


Рисунок 10. Список интеграционных объектов

Для создания интеграционного объекта, необходимо кликнуть на кнопку “Создать Интеграционный объект”.

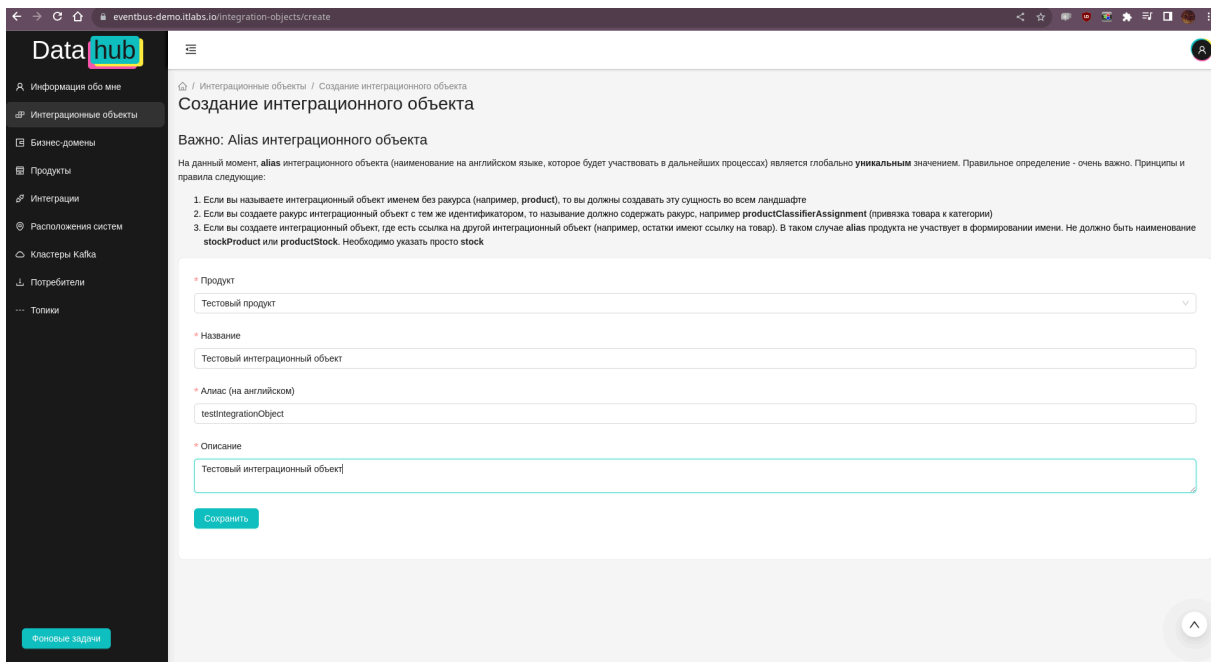


Рисунок 11. Форма создания интеграционного объекта

При создании необходимо заполнить 4 обязательных поля:

- 1) связь с продуктом (не меняется в дальнейшем)
- 2) Название интеграционного объекта
- 3) Алиас на английском языке, который участвует в формировании наименования топика

#### 4) Детальное описание

После описания мета-информации об интеграционном попадаем на страницу управления интеграционным объектом. На этой странице доступен функционал:

- Добавления/изменения структуры интеграционного объекта и полей
- Просмотр схемы данных
- Просмотр и управление примерами
- Просмотр сгенерированных SDK для языков Java и Python
- Управления статусами интеграционного объекта и его релизов

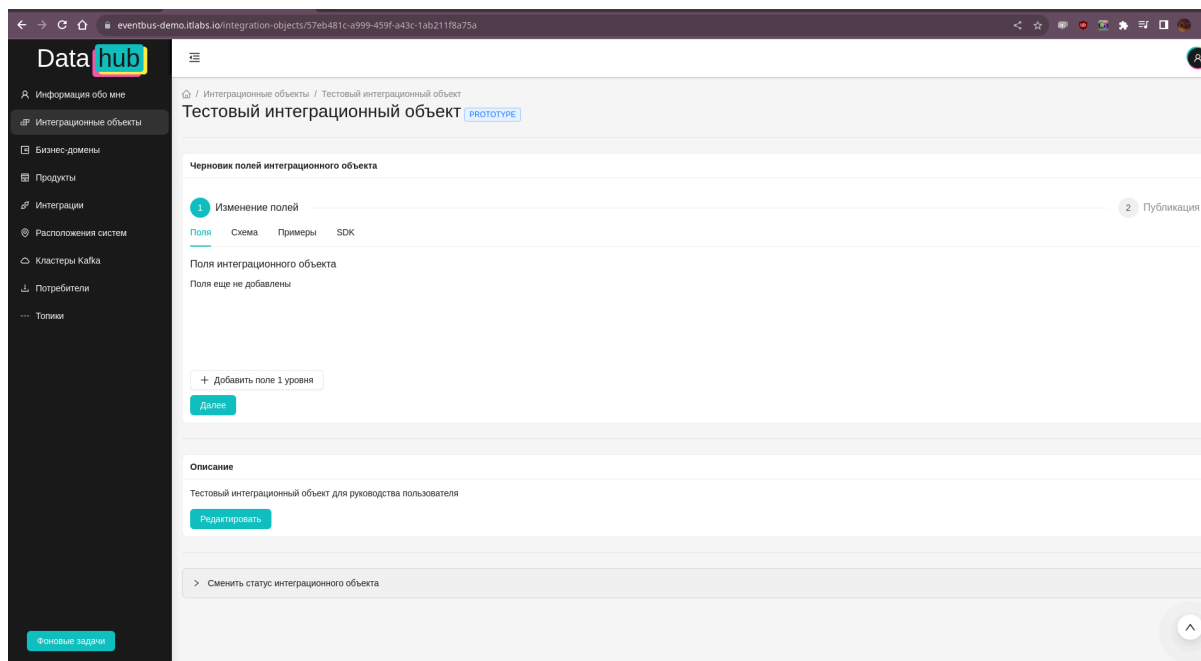


Рисунок 12. Форма создания интеграционного объекта

Следует начинать с заполнения полей и структуры интеграционного объекта. Для добавления первого уровня структуры интеграционного объекта, необходимо нажать на кнопку “Добавить поле 1 уровня”. Появившееся модальное окно позволяет создавать поля. При добавлении нового поля, необходимо заполнить:

- название (на английском языке)
- формат поля (детально описано в пункте 2.4)
- если у формата поля возможны ограничения (например, минимальное кол-во символов) - такие ограничения тоже будут добавлены и будут доступны для просмотра и заполнения
- описание поля
- параметры поля (детально описано в пункте 2.5)

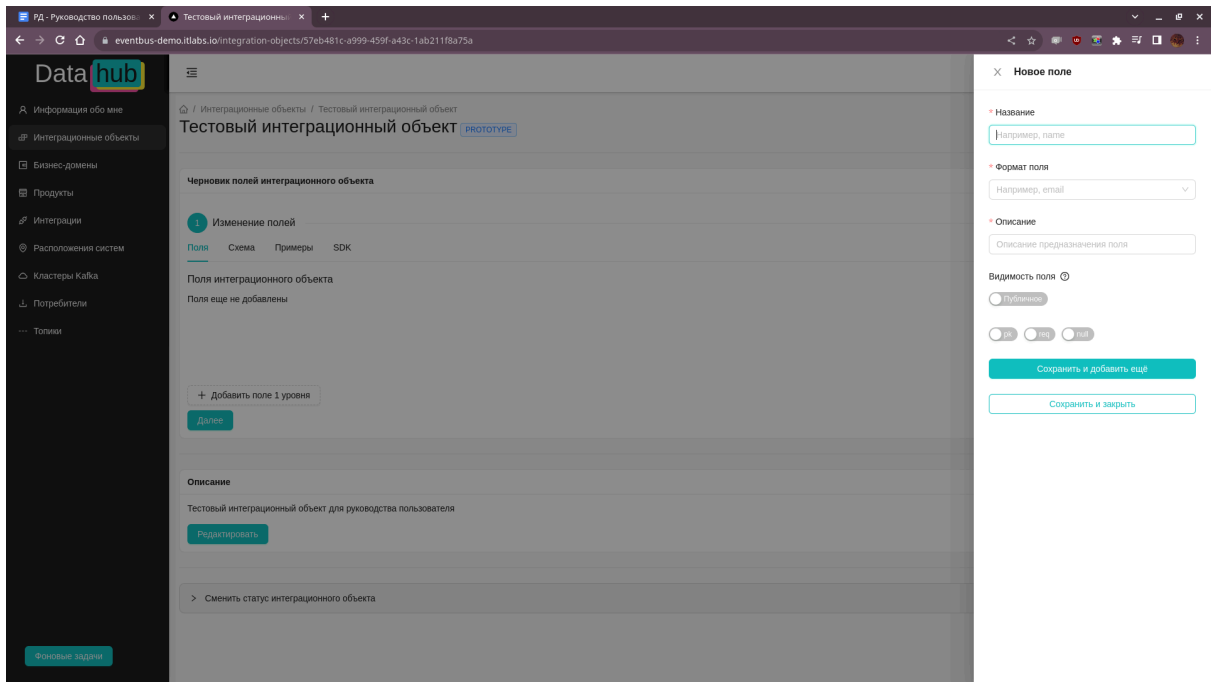


Рисунок 13. Модальное окно добавления полей

Для добавления поля, нажмите на одну из двух кнопок:

- сохранить и добавить еще - позволит продолжить добавлять поля, без закрытия модального окна
- сохранить и закрыть - поле будет добавлено и модальное окно будет скрыто

Для создания поля второго уровня вложенности - необходимо выйти из модального окна создания новых полей и вернуться на просмотр списка полей. У поля типа "объект" появляется слева от тега "Объект" кнопка. При ее нажатии - будет также открыто модальное окно, но добавление полей будет производиться на другой уровень вложенности.

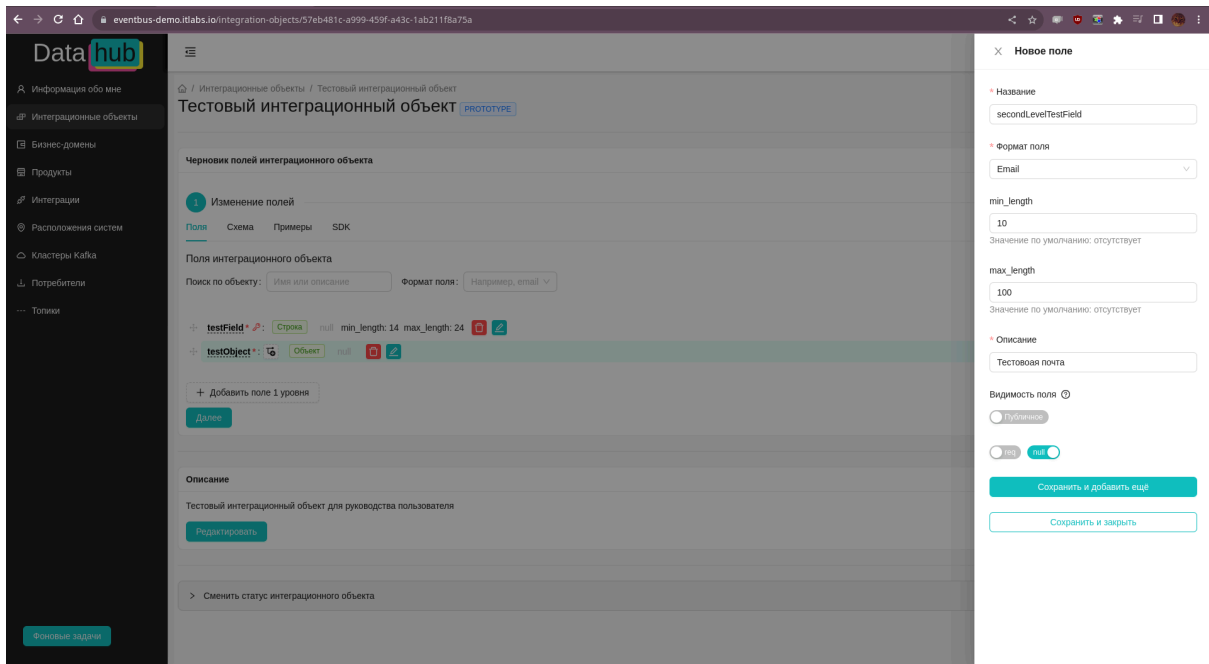


Рисунок 14. Модальное окно добавления полей

После создания и подготовки структуры интеграционного объекта, необходимо перейти на элемент “таб” (вкладку, далее по тексту - “таб”) с наименованием “схема”.

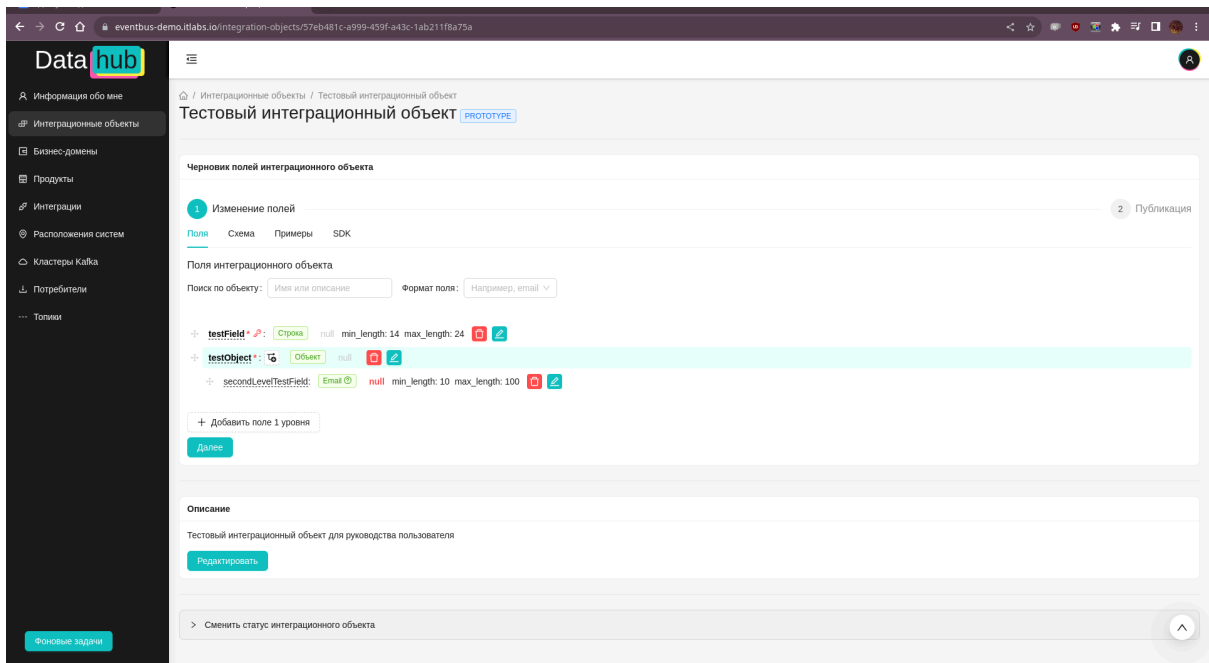


Рисунок 15. Форма создания интеграционного объекта

Можно увидеть JSON схему, которая сформировалась по структуре интеграционного объекта, которая была заполнена ранее. На текущий момент, ссылок между схемами не предусмотрено.



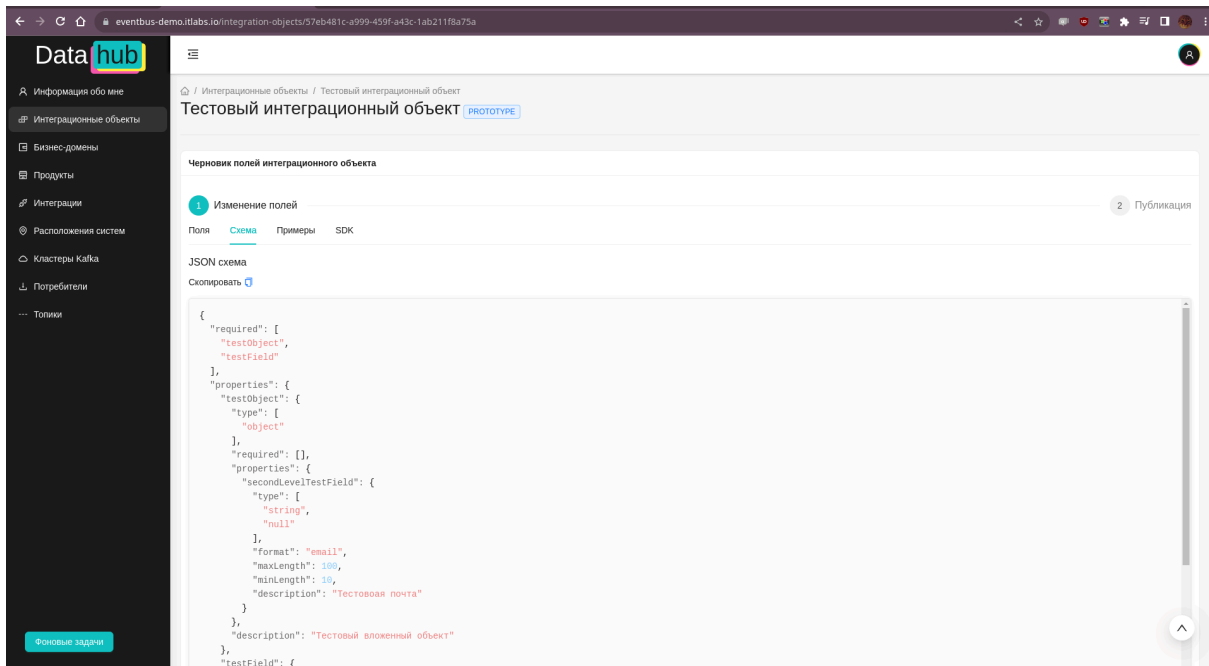


Рисунок 16. Просмотр схемы данных в формате JSON

При переходе на таб “Примеры” виден список примеров, а также кнопка добавления примера. При ее нажатии - появляется модальное окно, в котором можно:

- 1) указать имя
- 2) дать детальное описание
- 3) добавить сам пример в формате JSON

При заполнении примера, сразу проверяется корректность примера на формат JSON.

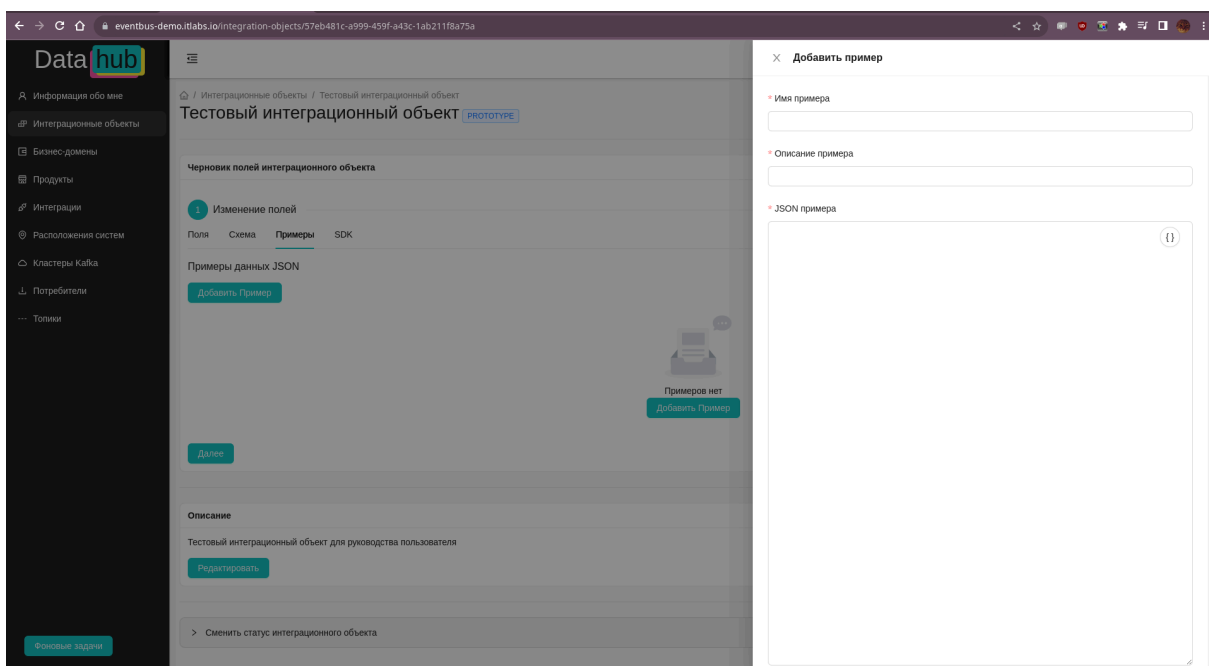


Рисунок 17. Модальное окно добавления примеров данных

При сохранении примера, производится валидация на соответствие структуре интеграционного объекта и показываются ошибки, на английском языке, формата “Поле \$ не удовлетворяет условию \$”. После этого, стоит исправить пример.

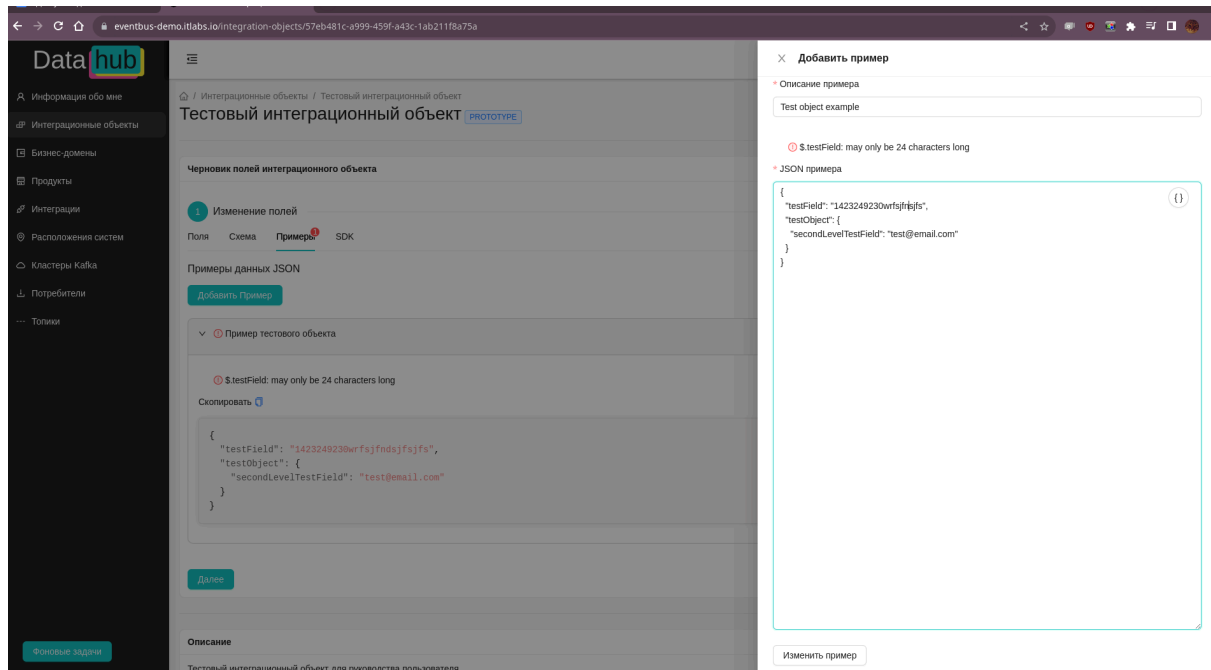


Рисунок 18. Модальное окно редактирования примеров данных

После изменения примера и нажатии на кнопку “Изменить пример”, производится обновление примера и повторная валидация. Если ошибок нет, значит пример соответствует заявленной структуре интеграционного объекта.

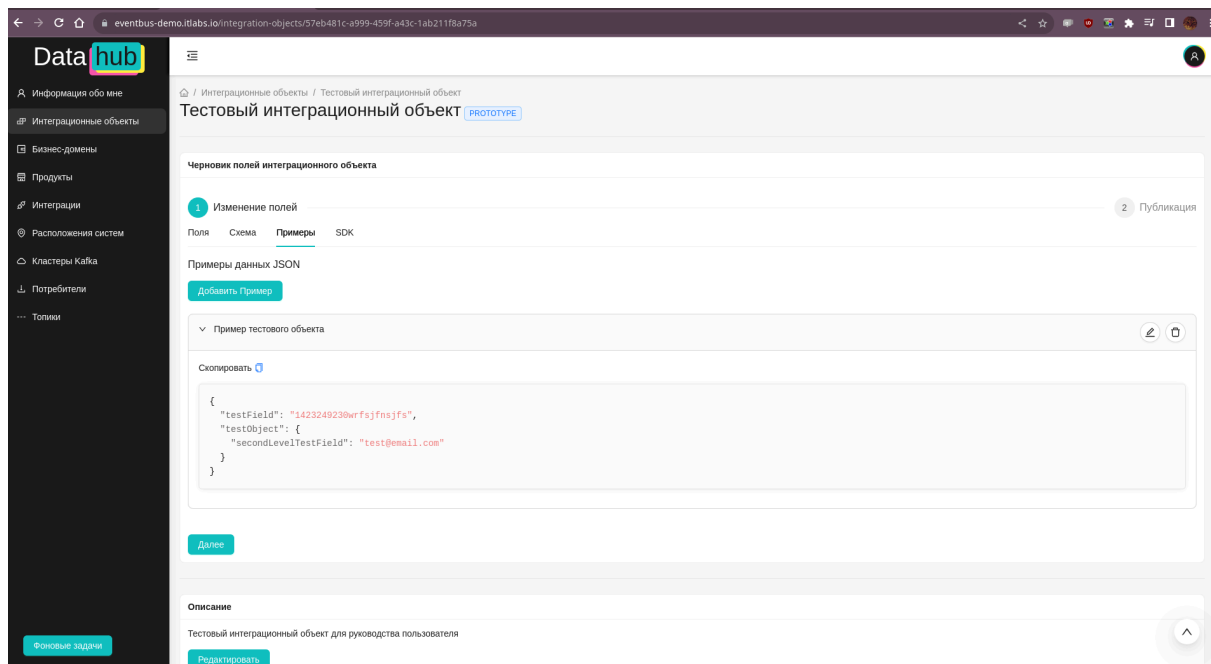


Рисунок 19. Форма создания интеграционного объекта

После заполнения одного или нескольких примеров, можно произвести активацию интеграционного объекта. Сделать это можно при помощи нажатия на кнопку “Далее”.

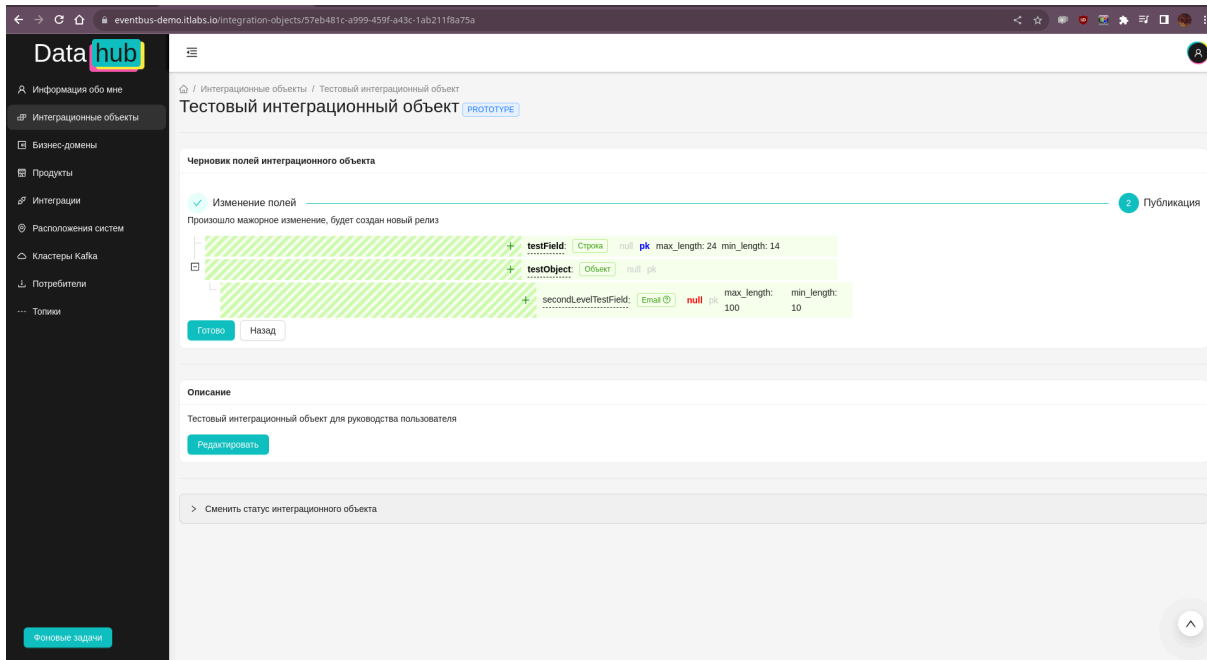


Рисунок 20. Форма просмотра изменений интеграционного объекта

Перед публикацией новой структуры (или измененной) показываются изменения во всех параметрах полей и показывается разница между текущей активной версией и новыми изменениями. При подтверждении, будет создано фоновое задание, статус выполнения которого будет показан в модальном окне.

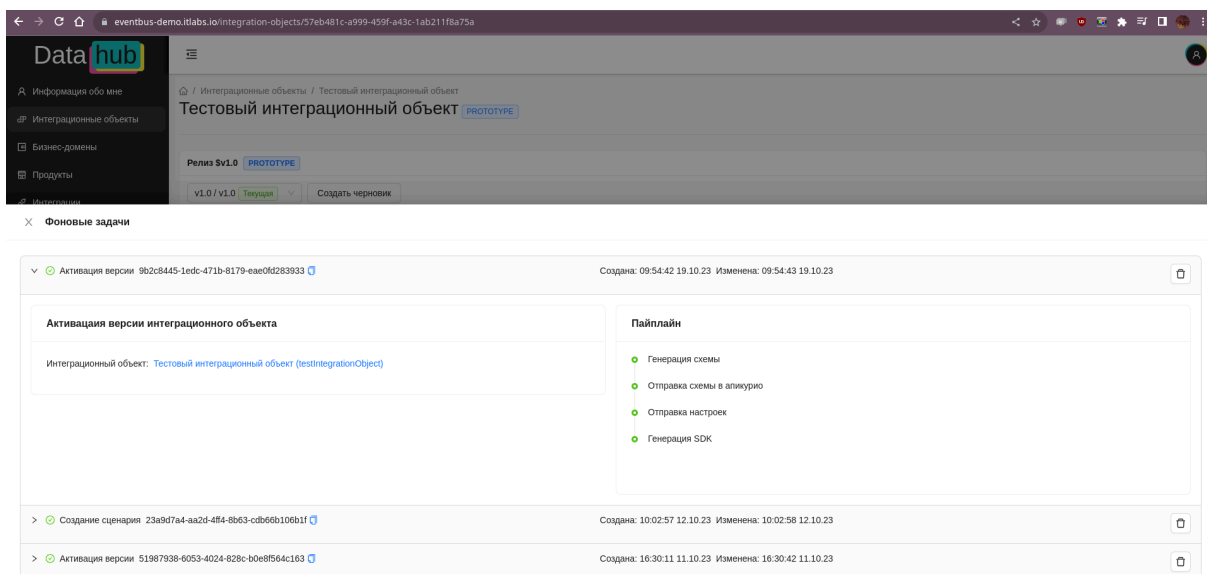


Рисунок 21. Модальное окно просмотра фоновых заданий

При активации версии интеграционного объекта, происходит следующий набор действий:

- повторно генерируется схема данных
- схема данных загружается в arisurigo
- если по интеграционному объекту были созданы сценарии интеграции, а изменение было минорным - производится рассылка в топики настроек задачи на обновление состояния
- производится отправка запроса на создания SDK

Спустя пару минут, SDK будут сгенерированы и отображены в табе "SDK".

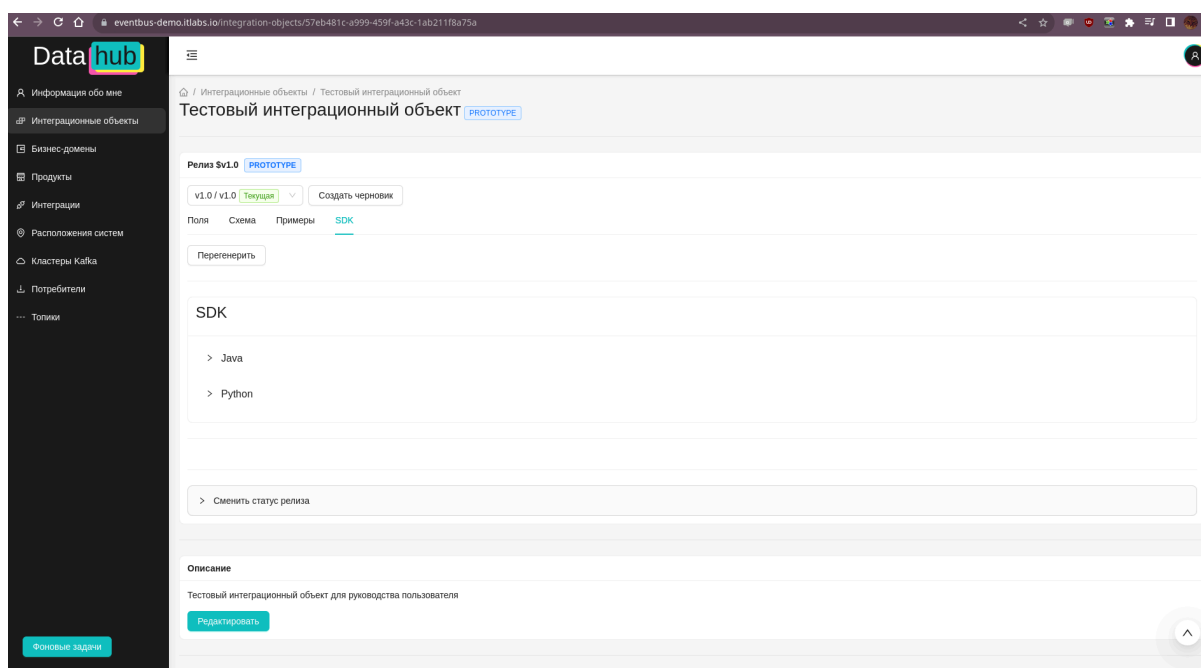


Рисунок 22. Список SDK

Интеграционные объекты состоят из релизов и версий релизов. Интеграционный объект и релиз имеют статусы:

- prototype - статус, означающий что над интеграционным объектом ведется работа и его состав может измениться:
  - можно создавать интеграционные сценарии только в окружении разработки
  - можно менять алиас интеграционного объекта
  - можно пересоздавать топик с данными
  - при изменении интеграционного объекта
- stable - статус, означающий, что интеграционный объект готов к эксплуатации, изменения в ближайшее время не планируются
- deprecated - статус, означающий, что эксплуатация этого интеграционного объекта и/или релиза подходит к завершению, стоит готовиться к изменению
- disabled - статус, означающий, что больше нельзя создать сценарий на этот интеграционный объект/релиз, текущие больше не исполняются

При добавлении любого изменения в интеграционный объект и его релиз, создается версия, которая является черновиком. Пока изменения не применены - они ни на что не влияют, после того как будет производится активация изменений, будет определен тип изменений:

- минорные - изменения не критичные и потребители данных могут продолжать получать данные без простоя
- мажорные - изменения критичные. Потребители остановятся и их работа будет прервана. Такие изменения создают новый релиз, необходимо создавать новые сценарии интеграции

Функционал просмотра и создания интеграции доступен на странице “Интеграции”

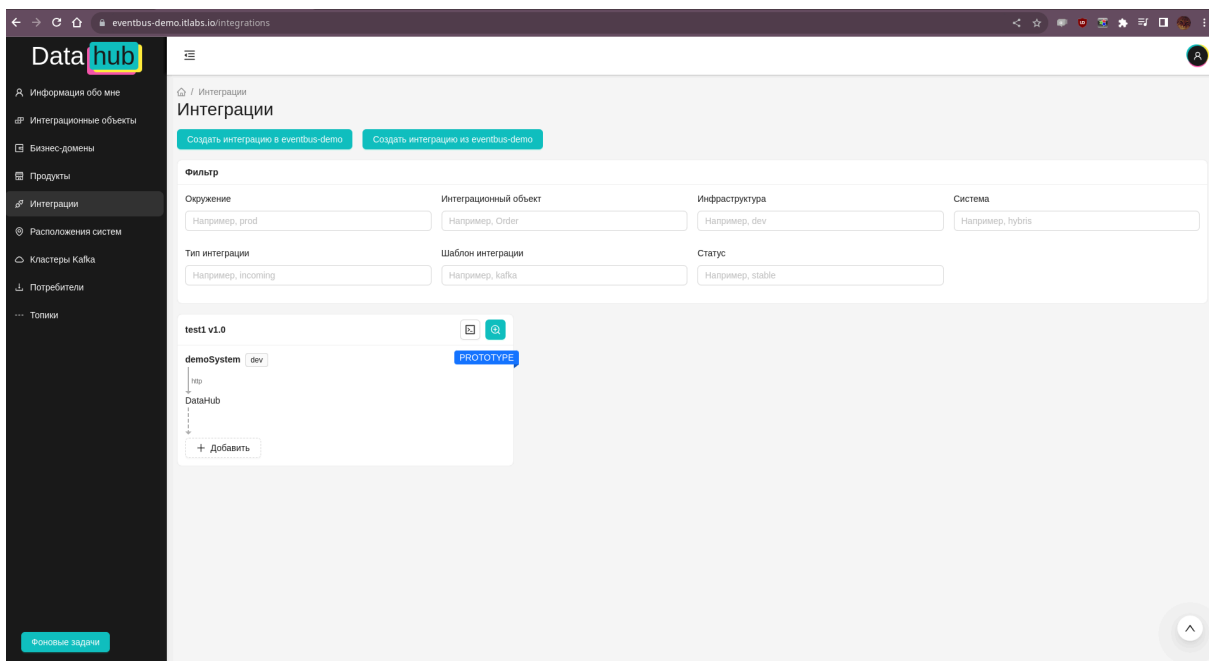


Рисунок 23. Список сценариев интеграции

Фильтры, доступные для просмотра:

- выбор окружения (development, stage, production)
- интеграционный объект
- инфраструктура (заполняется при инсталляции и/или системным администратором)
- система (заполняется при инсталляции и/или системным администратором)
- тип интеграции (подробно в пункте 2.3)
- шаблон интеграции
- статус интеграции

Для создания интеграционного сценария, необходимо выбрать тип сценария (пункт 2.3) и нажать соответствующую кнопку.

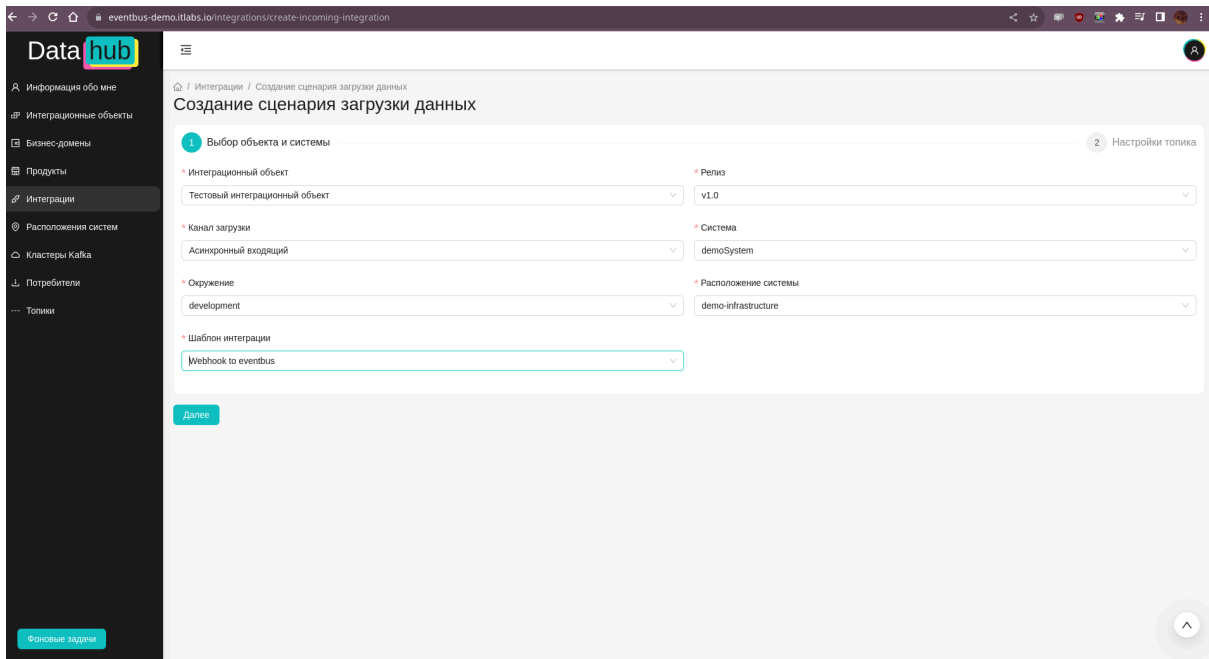


Рисунок 24. Форма создания интеграционного сценария

Необходимо указать интеграционный объект, канал загрузки, окружения, шаблон. Выбрать релиз и расположение системы.

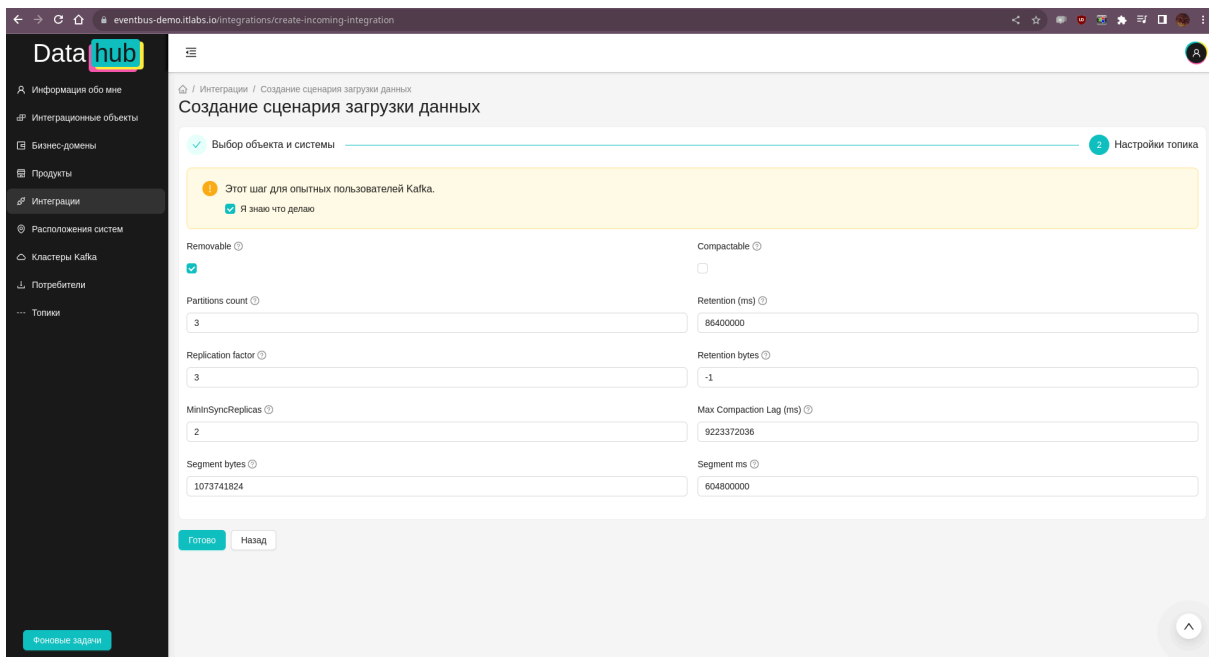


Рисунок 25. Форма конфигурации топика Apache Kafka

Для каждого типа интеграционного сценария, свой набор действий, которые необходимо выполнить. Для входящей интеграции, необходимо указать конфигурацию топика "Apache Kafka", или оставить ту, которая задана по умолчанию.

При создании входящего сценария, в результате выполнения фонового задания будет создан топик в Apache Kafka. Чтобы посмотреть его состояние, конфигурацию, топика и состояние потребителей - необходимо перейти на страницу "Кластеры кафка" и выбрать необходимый кластер.

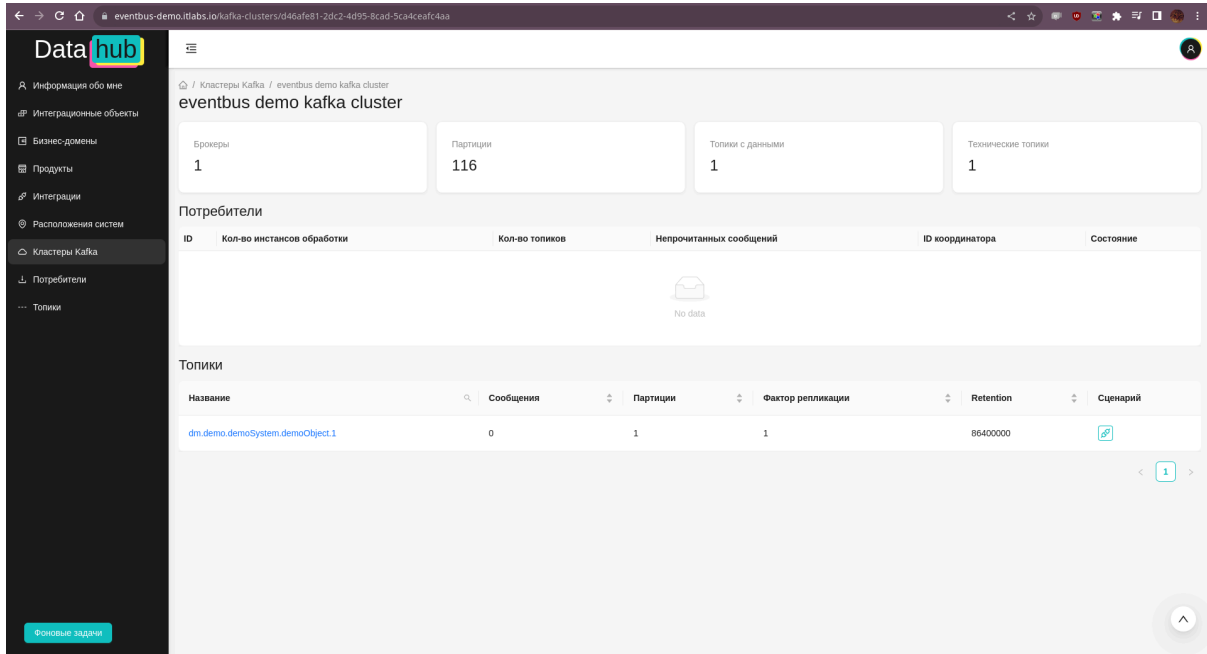


Рисунок 26. Форма просмотра объектов кластера Apache Kafka

Страница кластера "Apache Kafka" позволяет:

- посмотреть статистику кластера:
  - количество брокеров
  - количество партиций
  - количество топиков с данными и технических топиков
- список потребителей, которые получают данные из кластера
- список топиков

Для просмотра статистики топика и его партиций, необходимо кликнуть на конкретный топик и откроется его страница.

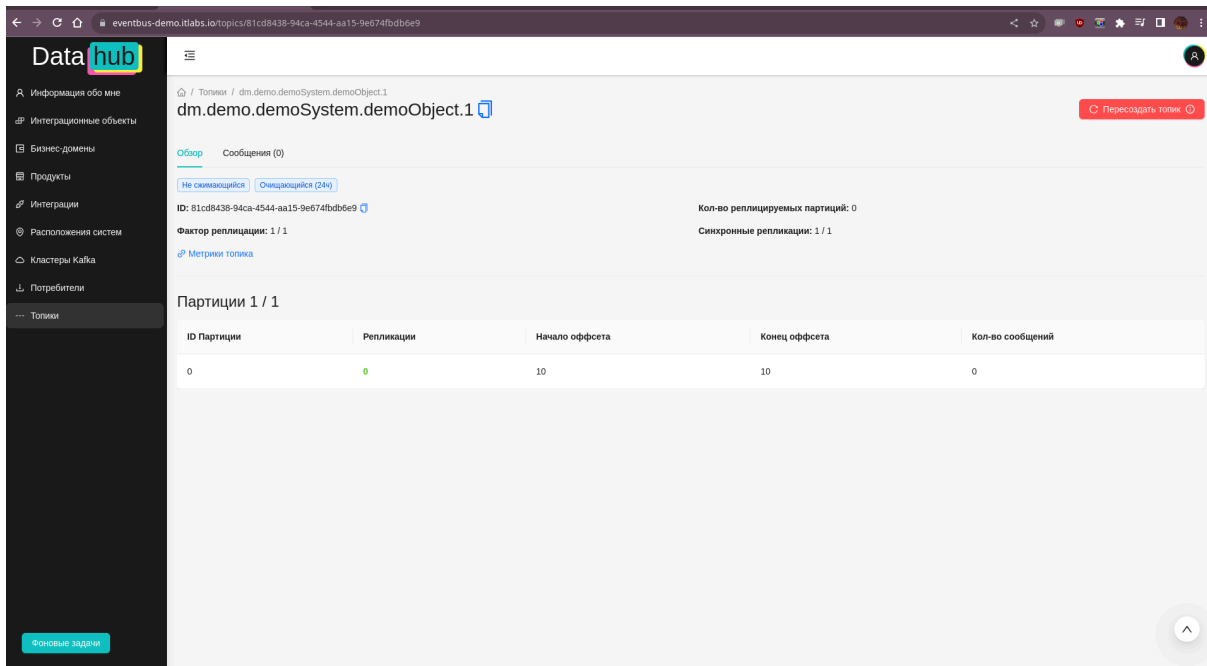


Рисунок 27. Форма создания интеграционного объекта

При просмотре информации о топике, доступна следующая информация:

- показ кол-во партиций, фактора репликации, синхронных партиций
- детальная статистика по каждой партиции:
  - идентификатор партиции
  - идентификаторы брокеров, на которых хранится реплика
  - начало и конец оффсета
  - кол-во сообщений

Также, можно перейти на там “сообщения”. В этом блоке доступен широкий диапазон фильтров, который позволяет искать и отображать сообщения, доступные на текущий момент в кластере. Список фильтров:

- по партициям:
  - все партиции
  - конкретный набор партиций
- тип фильтрации:
  - по офсету. Тогда необходимо указать номер оффсета, с которого будет начат поиск
  - по времени. Необходимо указать дату-время, с которых будет осуществляться поиск
- статус сообщения:
  - все - статус удаления не учитывается
  - удаленные - показываются только удаленные сообщения
  - не удаленные - показываются только не удаленные сообщения

Помимо фильтров существуют два режима просмотра:

- FORWARD - поиск производится сначала (с доступного оффсета в каждой партиции)



- LIVE - поиск производится с момента запроса, при этом соединение с компонентом UI живет пока не будет отменено пользователем. При добавлении новых данных в топик - они будут отображены в списке данных

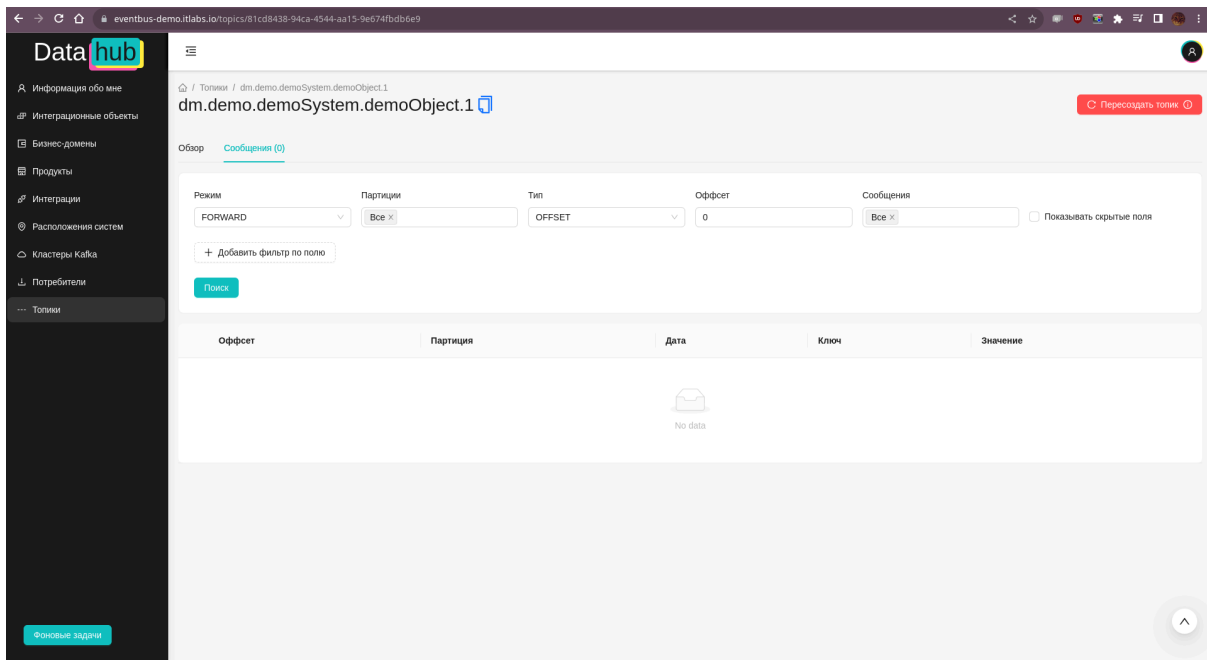


Рисунок 28. Список сообщений в топике

Для фильтрации по составу самих данных, доступна функция “фильтр по полю”. Для его запуска, необходимо нажать на кнопку “Добавить фильтр по полю” и заполнить:

- поле объекта - выбор показывается из текущей версии релиза интеграционного объекта. Можно выбирать поля разных уровней
- значение поля
- операция - способ фильтр и определения подходящего набора данных. Для каждого типа данных доступен свой набор операций:
  - равно - полное соответствие между данными в сообщении и значением поля
  - не равно - любое соответствие между данными в сообщении и значением поля
  - содержит - только для полей, которые имеют представление в JSON в виде строки. Означает наличие значения поля в данных
  - не содержит - только для полей, которые имеют представление в JSON в виде строки. Означает отсутствие значения поля в данных
  - больше - только для чисел. Значения поле больше чем данные в поле
  - больше или равно - только для чисел. Значения поле больше или равно чем данные в поле
  - меньше - только для чисел. Значения поле меньше чем данные в поле
  - меньше или равно - только для чисел. Значения поле меньше или равно чем данные в поле
  - NULL - значение у поля с данными не определено
  - not NULL - значение у поля с данными определено

Помимо фильтрации, можно пересоздать топик (только для среды разработки).

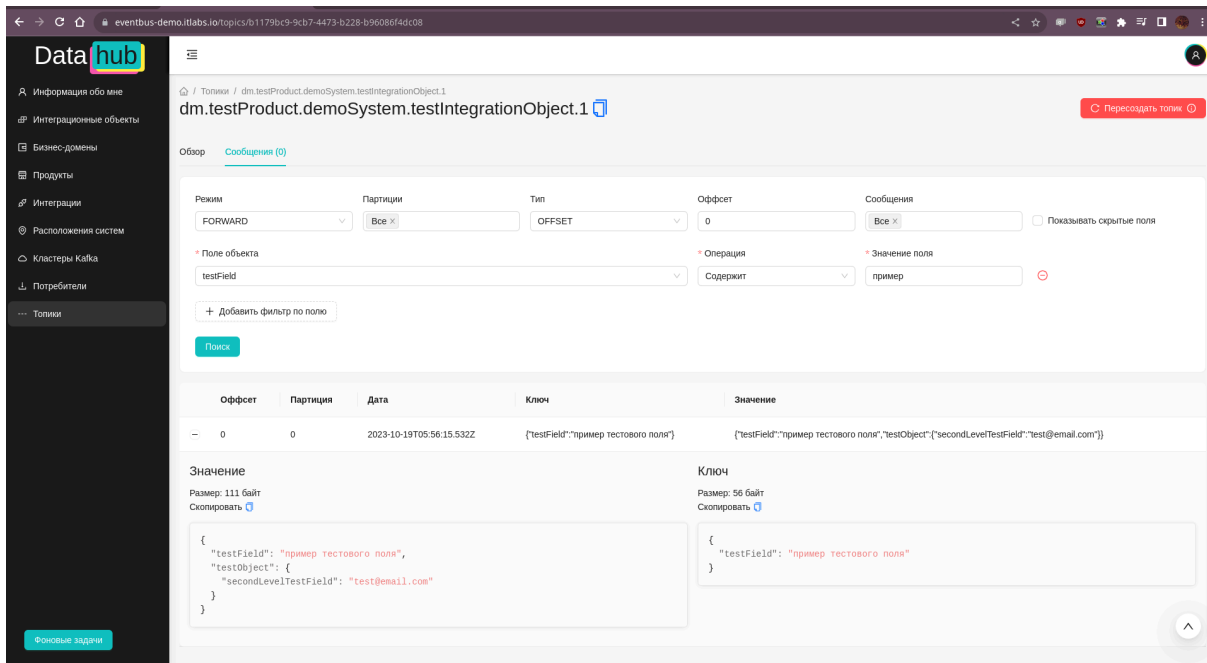


Рисунок 29. Отфильтрованный список сообщений в топике

Когда сообщение будет добавлено в топик, оно будет отображено в двух блоках (для ключа и для значения).

## 5. Подключение целевой системы к Apache Kafka

Целевой системой является система заказчика, которая умеет взаимодействовать с "Apache Kafka" с целью отправки и/или получения данных из топиков, которые были созданы в веб-приложении. Системой может являться:

- серверное приложение собственной разработки заказчика
- экземпляр фреймворка "Kafka connect"

На этой странице показана следующая справочная информация, которая необходима, чтобы подключиться к кластеру. Для каждого типа сценария доступна :

- мета-информация об интеграции - имя интеграционного объекта, релиз, формат данных
- адрес сервиса прокси
- информация о способе аутентификации.
- конфигурация окружения приложения, для языков Java и Python, с учетом использования готовых SDK
- OpenAPI схема, которая позволит реализовать свой клиент доступа к прокси на любом языке программирования
- JSON schema

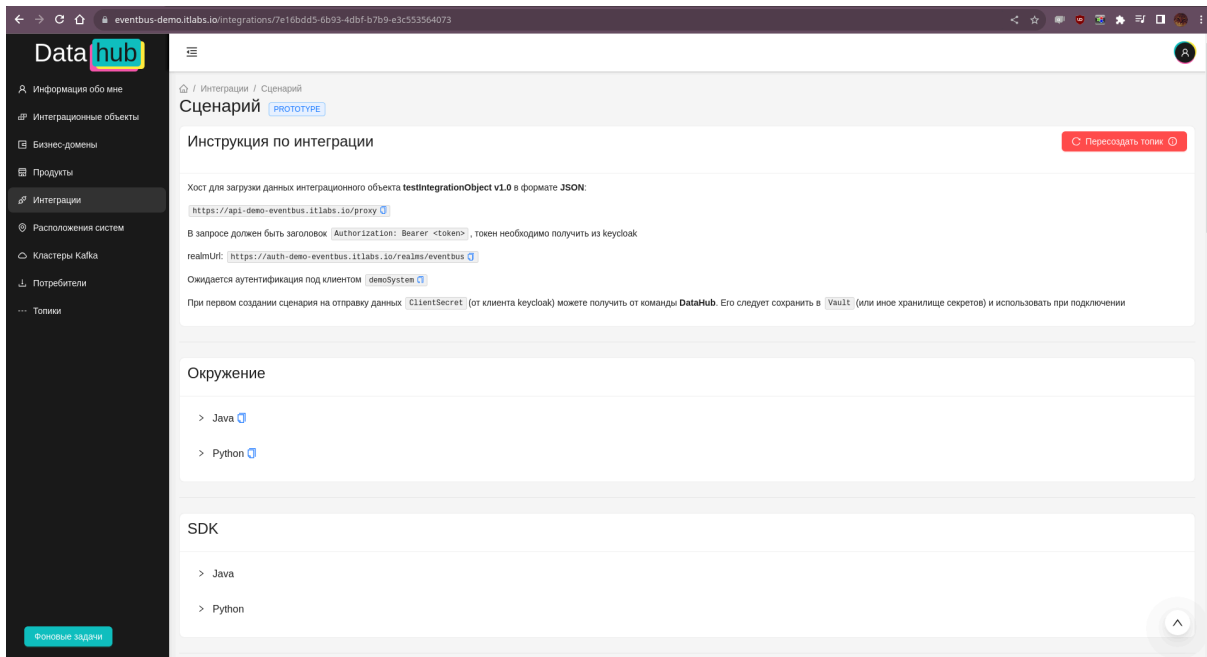


Рисунок 30. Просмотр информации о сценарии интеграции