

УТВЕРЖДЕН

29484562.62.01.29.685.001 34-ЛУ

ПРОГРАММА

**PERASPERA ESB. Программное обеспечение для оптимального
информационного обмена между различными корпоративными
информационными системами, включая исполнение в интегрированной
среде приложений. Версия 2.0**

ИНСТРУКЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ

29484562.62.01.29.685.001 34

Листов 31

Москва 2023

СОДЕРЖАНИЕ

1 НАЗНАЧЕНИЕ ПРОГРАММЫ	4
1.1 Наименование и условное обозначение программы	4
1.2 Краткое описание возможностей.....	4
1.3 Требования к пользователю	5
1.4 Функциональные обязанности пользователя	5
2 УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ.....	6
2.1 Требования к программному обеспечению	6
2.2 Требования к аппаратной части программы «PERASPERA ESB » версия 2.0.....	6
2.3 Требования к АРМ пользователя.....	8
3 ВЫПОЛНЕНИЕ ПРОГРАММЫ	9
3.1 Инструменты администрирования	9
3.2 Типовые операции администрирования программы «PERASPERA ESB » версия 2.0	10
3.2.1 Инициализация коннекторов.....	10
3.2.2 Просмотр состояния коннекторов	11
3.2.3 Просмотр состояния кластера Apache Kafka	13
3.2.4 Просмотр состояния кластера Kubernetes.....	15
3.2.5 Остановка системы.....	16
3.2.6 Запуск системы	16
3.3 Действия при отключении питания.....	17
3.3.1 Просмотр логов	17
3.3.2 Добавление нового узла.....	17
3.3.3 Управление из консоли (перечень команд)	20
3.3.4 Мониторинг состояния Ceph.....	20
3.4 Мониторинг состояния программы «PERASPERA ESB » версия 2.0	21
3.5 Конфигурирование и диагностика коннекторов	22
3.5.1 Конфигурирование коннекторов	22
3.5.2 Диагностика коннекторов.....	27
3.5.3 Действия при недоступности web-интерфейса Peraspera ESB UI	29
3.6 Нештатные ситуации	29

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

АРМ	– автоматизированное рабочее место
ГОСТ	– Государственный стандарт
ОС	– операционная система
ПД	– программная документация
ПО	– программное обеспечение
ПС	– программное средство
СУБД	– система управления базами данных

1 НАЗНАЧЕНИЕ ПРОГРАММЫ

1.1 Наименование и условное обозначение программы

Полное наименование системы: «Peraspera ESB. Программное обеспечение для оптимального информационного обмена между различными корпоративными информационными системами, включая исполнение в интегрированной среде приложений Версия 2.0».

Условное обозначение: Программа «PERASPERA ESB » версия 2.0.

1.2 Краткое описание возможностей

Программа «PERASPERA ESB » версия 2.0 предназначена для обеспечения информационного обмена между различными корпоративными информационными системами, а также исполнения в ее интегрированной среде приложений, осуществляющих сбор, обновление, агрегацию и обработку данных, поступающих от различных систем-источников данных, передачу информации системам-приемникам данных, а также, при необходимости, предоставление возможности выполнения аналитических запросов с возможностью создания отчетов для принятия управленческих решений и бизнес-решений во внешних системах.

Программа «PERASPERA ESB » версия 2.0 создана на основе цифровой платформы российской разработки, зарегистрированной в реестре российского ПО, являющейся готовой универсальной средой для быстрой установки и исполнения прикладного программного обеспечения или сервисов, использующих технологии и методы работы с большими данными (Big Data), машинного обучения и искусственного интеллекта, ориентированных на решение конкретных информационно-аналитических задач.

Основным назначением программы «PERASPERA ESB » версия 2.0 является обеспечение информационного обмена между системами-источниками данных и системами-приемниками данных и заключается в выполнении следующих функций:

- обеспечение управления очередностью сообщений в режиме реального времени и выполнение функции посредника между системой-источником и системой-приемником на основе брокера сообщений;
- обеспечение связи приложений и преобразование одного интерфейса в другой и в формате, воспринимаемом брокером сообщений, на основе универсальных или специализированных программных компонентов;
- обеспечение централизации и унификации взаимодействия систем-источников и систем-приемников данных;
- обеспечение независимости работоспособности и распределение функционала между микросервисами на основе архитектурных принципов микросервисной архитектуры.

В процессе выполнения программы «PERASPERA ESB » версия 2.0 технологического процесса должно обеспечиваться выполнение решения следующих задач:

- обеспечение доступа информационных систем к транспортной среде через опубликованные сервисы (REST);
- поддержка синхронного и асинхронного способа вызова сервисов;
- обеспечение доступа к опубликованным сервисам из сторонних информационных систем с помощью готовых или специально разработанных модулей;
- обеспечение унифицированного событийно-ориентированного обмена данными между различными информационными системами;
- обеспечение механизмов контроля и управления (аудиты, протоколирование).

1.3 Требования к пользователю

Уровень подготовки пользователя программы «PERASPERA ESB » версия 2.0 должен удовлетворять следующим требованиям:

- высшее техническое образование;
- знание и опыт работы с операционной системой CentOS;
- знание и опыт работы с Docker и Docker Swarm;
- знание и опыт работы с Keycloak;
- знание и опыт работы с Kubernetes;
- знание и опыт работы с Apache Kafka;
- знание и опыт работы с Cluster Manager for Apache Kafka (CMAK);
- знание технического английского языка.

1.4 Функциональные обязанности пользователя

В функциональные обязанности пользователя входят:

- 1) Конфигурирование и диагностика коннекторов.
- 2) Мониторинг состояния программы «PERASPERA ESB » версия 2.0.
- 3) Остановка и запуск системы.
- 4) Администрирование программы «PERASPERA ESB » версия 2.0, а именно:
 - инициализация коннекторов;
 - мониторинг состояния коннекторов, кластера Kubernetes, кластера Apache Kafka.

2 УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

2.1 Требования к программному обеспечению

Для установки и эксплуатации программы «PERASPERA ESB » версия 2.0 необходимо следующее системное программное обеспечение:

- ubuntu сервер 18.04 и выше;
- docker 19.03.2 и выше;
- kubernetes 1.18.4 и выше.

Для функционирования программы «PERASPERA ESB » версия 2.0 необходимо следующее программное обеспечение:

- Kafka Connect;
- Schema Registry;
- KsqlDB;
- Kubernetes;
- Kubernetes Dashboard;
- Kube Metrics Service;
- Kubernetes Ingress;
- Apache Zookeeper;
- Apache Kafka;
- Rook;
- Ceph;
- CMAK (Cluster Manager for Apache Kafka);
- Peraspera ESB UI.

Администрирование вышеуказанного программного обеспечения выполняется в соответствии с документацией на соответствующее программное обеспечение.

2.2 Требования к аппаратной части программы «PERASPERA ESB » версия 2.0

Состав и характеристики технических средств определяются требованием обеспечения отказоустойчивой конфигурации программы «PERASPERA ESB » версия 2.0, которая должна функционировать в штатном режиме при условии выхода из строя определённой части технических средств, а также составом необходимых программных продуктов.

Серверная часть программы «PERASPERA ESB » версия 2.0 реализована в виде семи серверов, требования к вычислительным ресурсам которых приведены в нижеприведенной таблице (Таблица 2.1).

Таблица 2.1 - Требования к вычислительным ресурсам серверов программы «PERASPERA ESB » версия 2.0

Наименование	Описание	Системные требования
Сервер 1	Master node	4 Core + 8 Gb RAM
Сервер 2	Master node	4 Core + 8 Gb RAM
Сервер 3	Master node	4 Core + 8 Gb RAM
Сервер 4	Worker node	12 Core + 24 Gb RAM
Сервер 5	Worker node	12 Core + 24 Gb RAM
Сервер 6	Worker node	12 Core + 24 Gb RAM
Сервер 7	Worker node for KSQL DB server	16 Core + 32 Gb RAM

Конфигурация технических средств программы «PERASPERA ESB» и перечень необходимого программного обеспечения, устанавливаемого на каждую виртуальную машину, приведены в таблице ниже (Таблица 2.2).

Таблица 2.2 - Системные требования для программных продуктов программы «PERASPERA ESB» версия 2.0

№	Продукт	Системные требования
1	Kubernetes Ingress (3 instances)	2 Core + 4 Gb RAM
2	Kubernetes dashboard + Kube metrics	1 Core + 1 Gb RAM
3	KSQL DB	16 Core + 32 Gb RAM
4	Kafka Connect	8 Core + 16 Gb RAM
5	Schema Registry	2 Core + 2 Gb RAM
6	Kafka Rest	2 Core + 2 Gb RAM
7	Apache Zookeeper (3 instances)	3 Core + 3 Gb RAM
8	Apache Kafka (5 instances)	10 Core + 10 Gb RAM
9	Rook Ceph	4 Core + 16 Gb RAM
10	Операционная система	2 Core + 2 Gb RAM

Ниже представлена схема организации кластера Kubernetes для программы «PERASPERA ESB» версия 2.0.



Рисунок 2-1 – Схема кластера Kubernetes программы «PERASPERA ESB» версия 2.0

2.3 Требования к АРМ пользователя

В качестве АРМ пользователя программы «PERASPERA ESB» версия 2.0 используются персональные компьютеры, конфигурации которых включают:

- центральный процессор Intel 2xCore;
- объем оперативной памяти 4Gb;
- ОС Microsoft Windows 7 (не ниже);
- браузеры Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Edge последних версий;
- общее офисное ПО для работы с текстовыми документами и электронными таблицами – Microsoft Office 2013 (15.0.5127.100) 32-bit.

3 ВЫПОЛНЕНИЕ ПРОГРАММЫ

3.1 Инструменты администрирования

В нижеприведенной таблице (Таблица 3.1) указан перечень инструментов администрирования программы «PERASPERA ESB » версия 2.0.

Таблица 3.1 - Список инструментов администрирования программы «PERASPERA ESB » версия 2.0

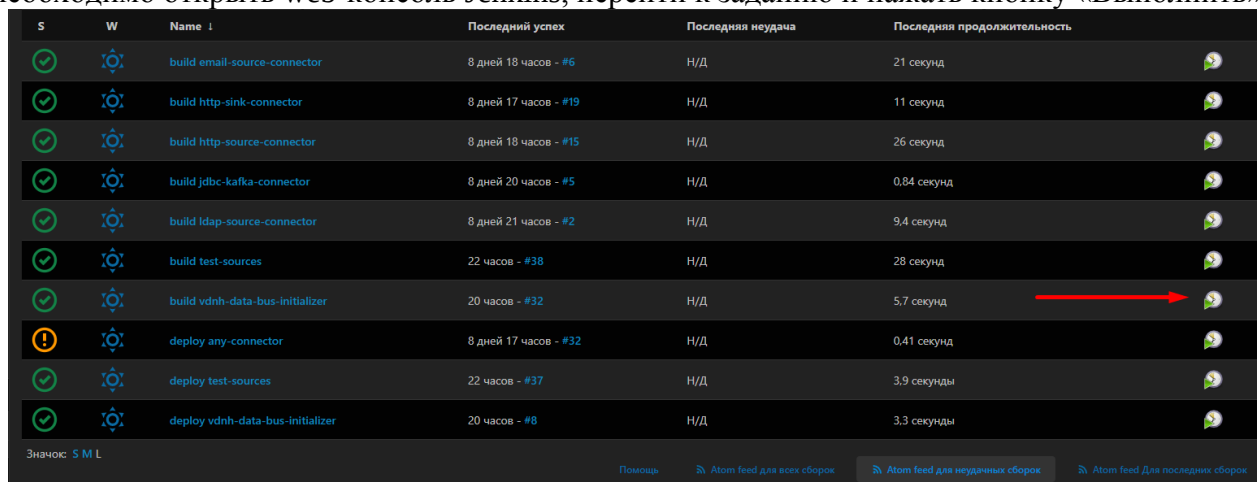
Субъект администрирования	Инструмент	Описание
Сервис непрерывной интеграции и развертывания jenkins	Web-консоль, доступная по адресу https://devops.vdnh.ru/jenkins/	Функции: 1) Запуск сервиса инициализации коннекторов. 2) Просмотр результатов запуска инициализации коннекторов.
Сервис мониторинга и администрирования коннекторов Peraspera ESB UI	Web-консоль, доступная по адресу https://peraspera-esb.ec-leasing.ru	Функции: 1) Проверка состояния коннекторов. 2) Просмотр топиков коннекторов и данных в типах. 3) Просмотр схем данных коннекторов. 4) Создание коннекторов.
Сервис администрирования Apache Kafka	Web-консоль, доступная по адресу: https://devops.vdnh.ru/cmak/	Функции: 1) Проверка состояния брокеров. 2) Администрирование топиков. 3) Просмотр состояния получателей (consumers).
Kubernetes Dashbord - сервис мониторинга кластера Kubernetes	Web-консоль, доступная по адресу: https://devops.vdnh.ru/kubernetes/#/workloads?namespace=vdnh-data-bus	Функции: 1) Мониторинг и управление кластером Kubernetes. 2) Просмотр состояния работающих сервисов (pods, services, deployments).

Субъект администрирования	Инструмент	Описание
		3) Просмотр логов.
Ceph Dashbord – сервис мониторинга Ceph	Web-консоль, доступная по адресу: https://devops.vdnh.ru/ceph-dashboard	Функции: мониторинг и управление Ceph

3.2 Типовые операции администрирования программы «PERASPERA ESB » версия 2.0

3.2.1 Инициализация коннекторов

Инициализация происходит при первом запуске коннектора, а также в случае аварийной остановки одного или нескольких коннекторов. Инициализация реализована в виде задания для сервиса непрерывной интеграции и развертывания Jenkins. Для запуска задания необходимо открыть web-консоль Jenkins, перейти к заданию и нажать кнопку «Выполнить».



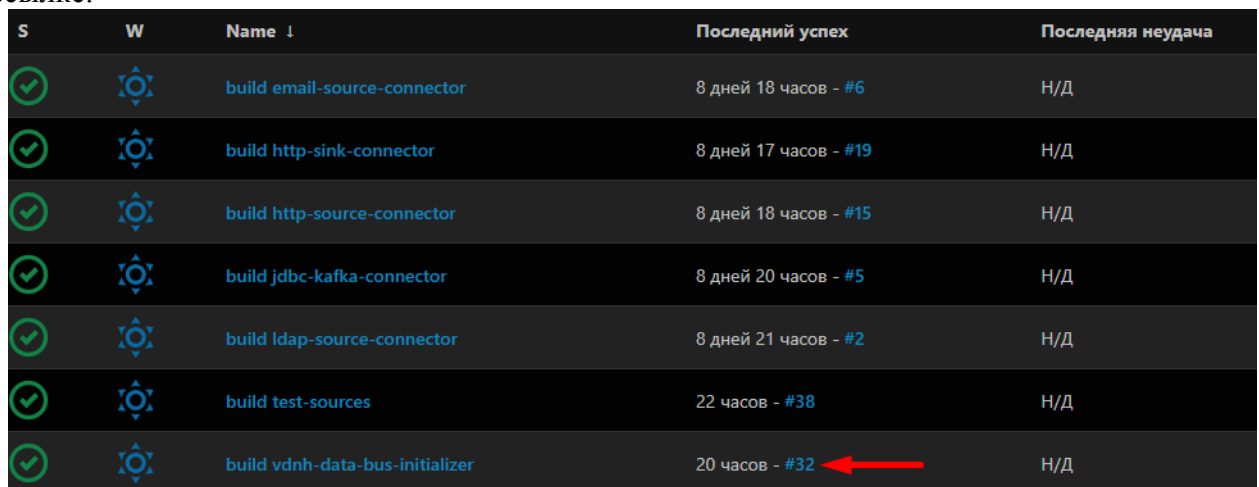
S	W	Name ↓	Последний успех	Последняя неудача	Последняя продолжительность
✓	⚙	build email-source-connector	8 дней 18 часов - #6	Н/Д	21 секунд
✓	⚙	build http-sink-connector	8 дней 17 часов - #19	Н/Д	11 секунд
✓	⚙	build http-source-connector	8 дней 18 часов - #15	Н/Д	26 секунд
✓	⚙	build jdbc-kafka-connector	8 дней 20 часов - #5	Н/Д	0,84 секунд
✓	⚙	build ldap-source-connector	8 дней 21 часов - #2	Н/Д	9,4 секунд
✓	⚙	build test-sources	22 часов - #38	Н/Д	28 секунд
✓	⚙	build vdnh-data-bus-initializer	20 часов - #32	Н/Д	5,7 секунд
⚠	⚙	deploy any-connector	8 дней 17 часов - #32	Н/Д	0,41 секунд
✓	⚙	deploy test-sources	22 часов - #37	Н/Д	3,9 секунды
✓	⚙	deploy vdnh-data-bus-initializer	20 часов - #8	Н/Д	3,3 секунды

Значок: S M L

Помощь Atom feed для всех сборок Atom feed для неудачных сборок Atom feed для последних сборок

Рисунок 3-1 - web-консоль Jenkins

Просмотр состояния выполнения можно осуществить, перейдя по соответствующей ссылке.



S	W	Name ↓	Последний успех	Последняя неудача
✓	⚙	build email-source-connector	8 дней 18 часов - #6	Н/Д
✓	⚙	build http-sink-connector	8 дней 17 часов - #19	Н/Д
✓	⚙	build http-source-connector	8 дней 18 часов - #15	Н/Д
✓	⚙	build jdbc-kafka-connector	8 дней 20 часов - #5	Н/Д
✓	⚙	build ldap-source-connector	8 дней 21 часов - #2	Н/Д
✓	⚙	build test-sources	22 часов - #38	Н/Д
✓	⚙	build vdnh-data-bus-initializer	20 часов - #32	Н/Д

Рисунок 3-2 – Ссылка для перехода к просмотру состояния

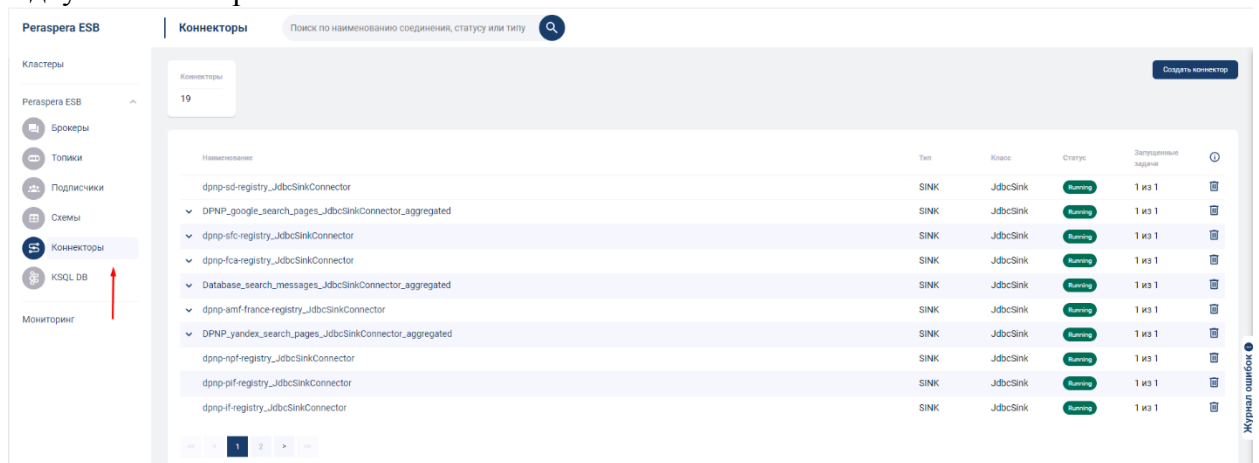
В результате отобразится консоль с результатами выполнения задания. Статус задания можно прочитать в последней строке консоли. В случае возникновения ошибочных ситуаций они также регистрируются в консоли.

```
689780320e34: Layer already exists
7252e4afe941: Layer already exists
6b10d46d1b5f: Layer already exists
5c34c0e8f6a2: Layer already exists
c08e7f3d2e3d: Layer already exists
0b911edbb97f: Layer already exists
54e42005468d: Layer already exists
f340e69db130: Pushed
9e7e505b0487: Pushed
1.0: digest: sha256:1c7d61db20d3230bdd6024788cd6407cd443a9a12d114204c8fbcdcb8330ab57 size: 4706
Docker Build Done
Strings match run condition: string 1=[test], string 2=[test]
Run condition [Strings match] enabling perform for step [BuilderChain]
Triggering projects: deploy vdnh-data-bus-initializer
Strings match run condition: string 1=[test], string 2=[prod]
Run condition [Strings match] preventing perform for step [BuilderChain]
Strings match run condition: string 1=[test], string 2=[test]
Run condition [Strings match] enabling perform for step [Builder to mark whether executed]
Finished: SUCCESS
```

Рисунок 3-3 - Консоль с результатами выполнения задания

3.2.2 Просмотр состояния коннекторов

Для просмотра состояния коннекторов следует перейти в сервис Peraspera ESB UI на вкладку «Коннекторы».



Наименование	Тип	Класс	Статус	Загруженные задачи	Действия
dpnp-sd-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ
DPNP-google_search_pages_jdbcSinkConnector_aggregated	SINK	JdbcSink	Running	1 из 1	ⓘ
dpnp-sfc-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ
dpnp-fca-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ
Database_search_messages_jdbcSinkConnector_aggregated	SINK	JdbcSink	Running	1 из 1	ⓘ
dpnp-amf-france-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ
DPNP-yandex_search_pages_jdbcSinkConnector_aggregated	SINK	JdbcSink	Running	1 из 1	ⓘ
dpnp-npf-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ
dpnp-pf-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ
dpnp-if-registry_jdbcSinkConnector	SINK	JdbcSink	Running	1 из 1	ⓘ

Рисунок 3-4 – Отображение состояния коннекторов

В открывшейся панели отображается список имеющихся коннекторов, а также отображается их состояние. При переходе по ссылке из таблицы отображается детальная информация о коннекторе, в частности его конфигурация (вкладка «Настройки»).



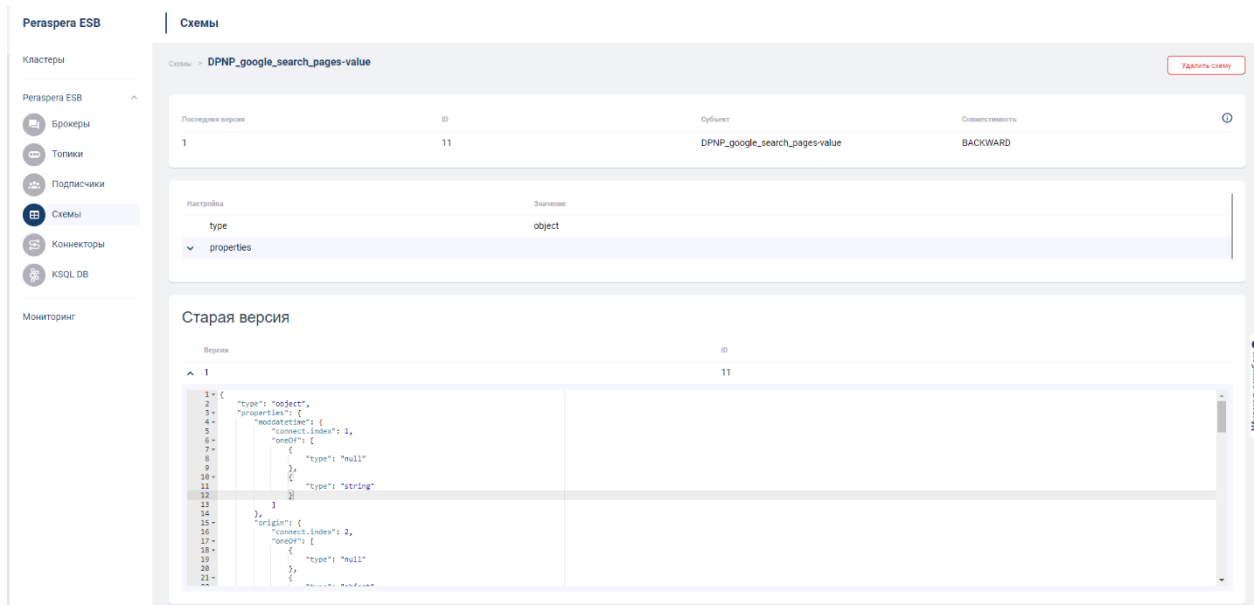


Рисунок 3-8 – Отображение детальной информации о конкретной схеме

3.2.3 Просмотр состояния кластера Apache Kafka

Для администрирования кластера Apache Kafka необходимо открыть сервис администрирования СМАК (Cluster Manager for Apache Kafka). Сервис СМАК в частности позволяет производить мониторинг топиков Apache Kafka и выполнять действия в случае возникновения нештатных ситуаций. Для этого следует перейти на соответствующую закладку.

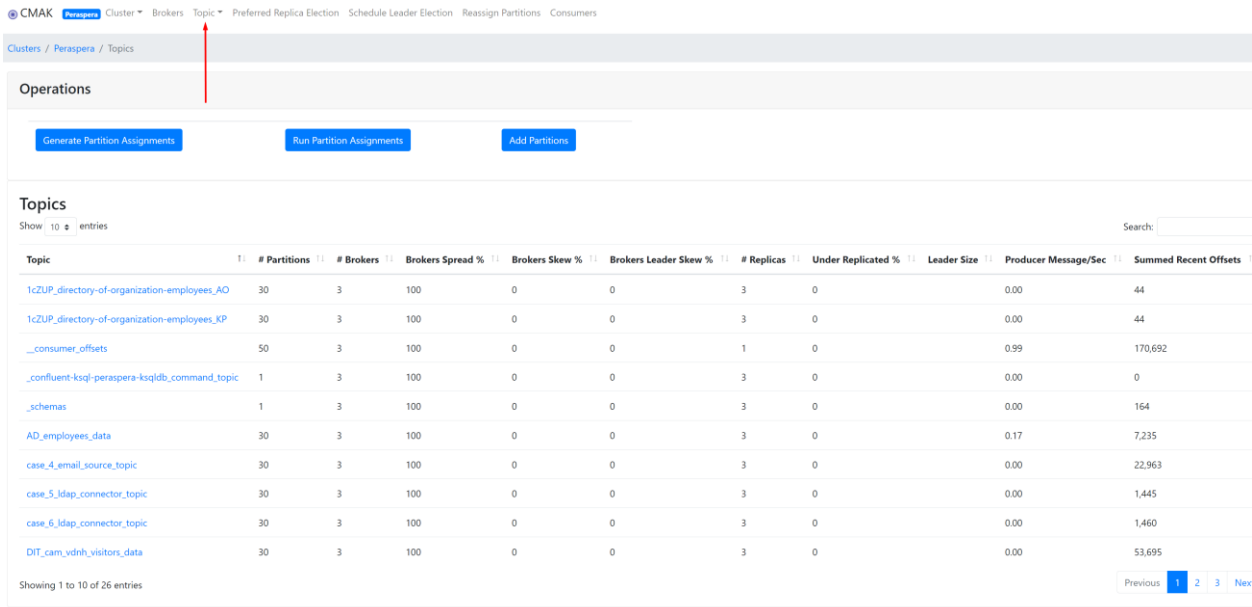


Рисунок 3-9 – Отображение состояния кластера Apache Kafka

На вышеприведенном рисунке показан перечень топиков, а также приведена информация, позволяющая оценить состояние кластера. Например, если в колонке Under Replicated указано число процентов, отличное от нуля, это означает, что запущен процесс репликации данных. Такое бывает в случае потери связи или временного выхода из строя одного из брокеров Apache Kafka.

Generate Partition Assignments – кнопка отвечает за распределение партиций между брокерами с учетом фактора репликации. Это может быть полезно при изменении фактора репликации, когда нужно автоматически выполнить распределение. Действие, выполняемое по данной кнопке создаст таблицу распределения партиций.

Run Partition Assignments – кнопка отвечает за физическое выполнения распределения партиций между брокерами в соответствии с построенной на предыдущем шаге таблицей. Эта операция может занимать продолжительное время в зависимости от объема данных, хранящихся в топиках.

В случае потери (физической или в результате недоступности) одного или нескольких лидеров для партиций может возникнуть нештатная ситуация. В этом случае следует запустить процедуру Preferred Replica Election.

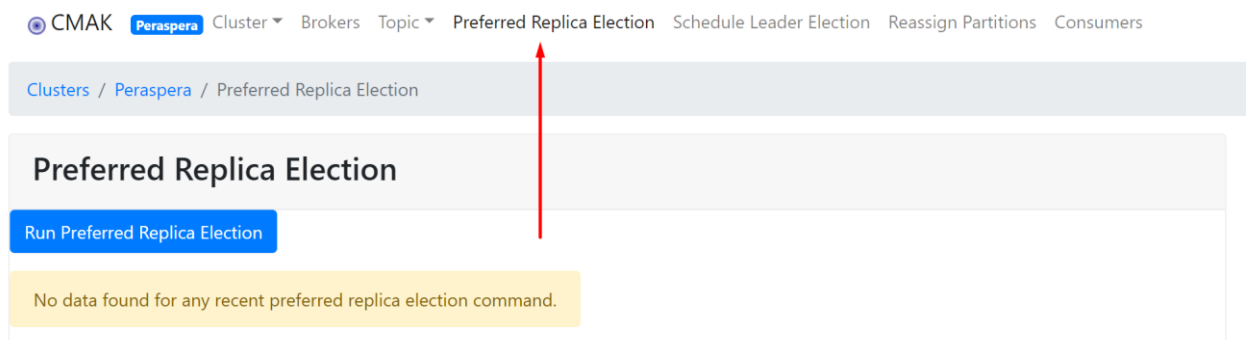


Рисунок 3-10 – Запуск процедуры Preferred Replica Election

Результатом выполнения процедуры будет назначение новых лидеров из числа доступных для тех партиций, у которых лидеры отсутствуют.

Для просмотра состояния получателей (consumers) данных следует перейти на соответствующую вкладку.

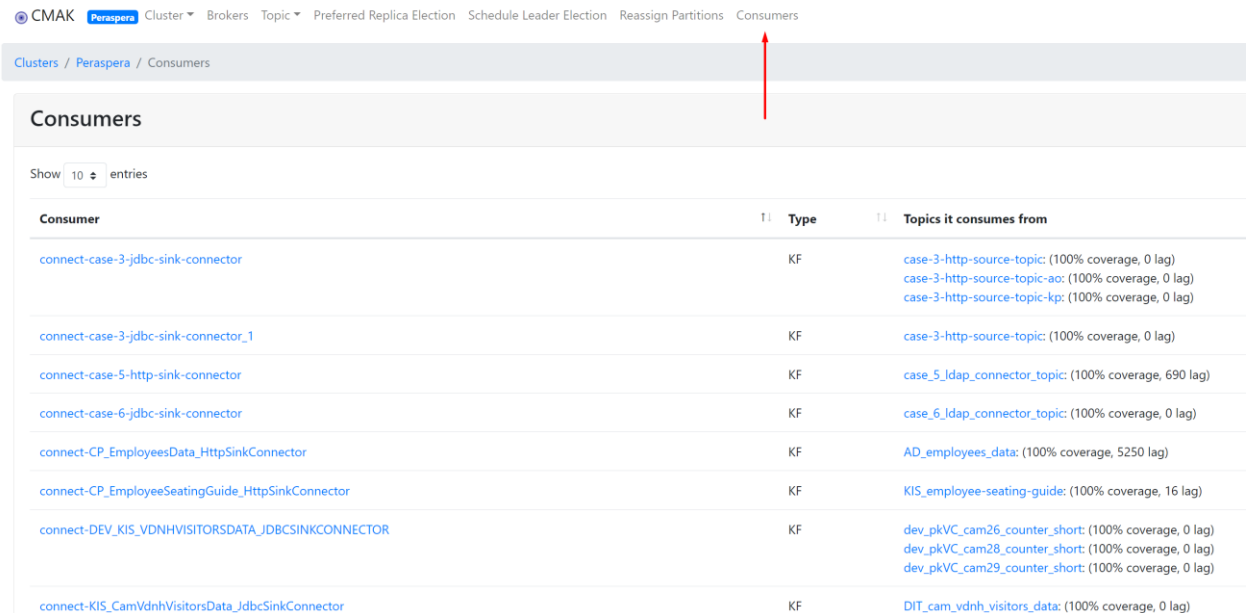


Рисунок 3-11 - Просмотр состояния получателей данных

В списке указаны получатели данных, топика, из которых читает данные каждый получатель, а также число непрочитанных сообщений из каждого топика. Для просмотра

детальной информации о получателе данных следует перейти по ссылке, нажав наименование получателя.

Clusters / Peraspera / Consumers / connect-case-3-jdbc-sink-connector		
connect-case-3-jdbc-sink-connector		
Consumed Topic Information		
Topic	Partitions Covered %	Total Lag
case-3-http-source-topic	100	0
case-3-http-source-topic-ao	100	0
case-3-http-source-topic-kp	100	0

Рисунок 3-12 – Просмотр информации о получателе

3.2.4 Просмотр состояния кластера Kubernetes

Для администрирования кластера Kubernetes необходимо открыть сервис Kubernetes Dashboard.

С помощью сервиса Kubernetes Dashboard можно получить информацию обо всех компонентах кластера в рамках программы «PERASPERA ESB». Для просмотра и администрирования сервиса Kubernetes Dashboard необходимо открыть пространство имен для программы «PERASPERA ESB», где будут отображены все относящиеся к программе «PERASPERA ESB» компоненты кластера.

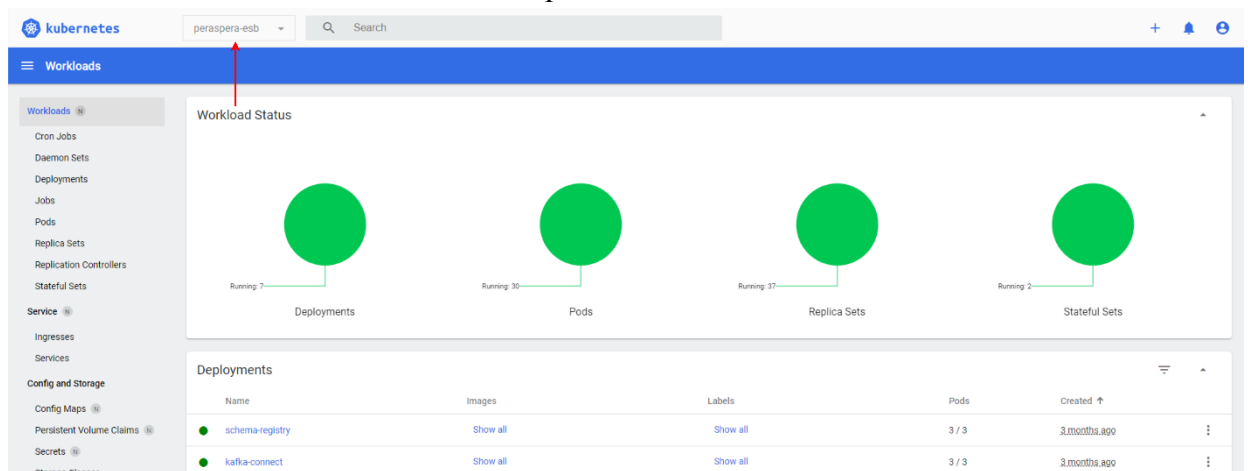


Рисунок 3-13 - Просмотр пространства имен

Состояние подов (pods) кластера можно просмотреть, перейдя на соответствующую закладку.

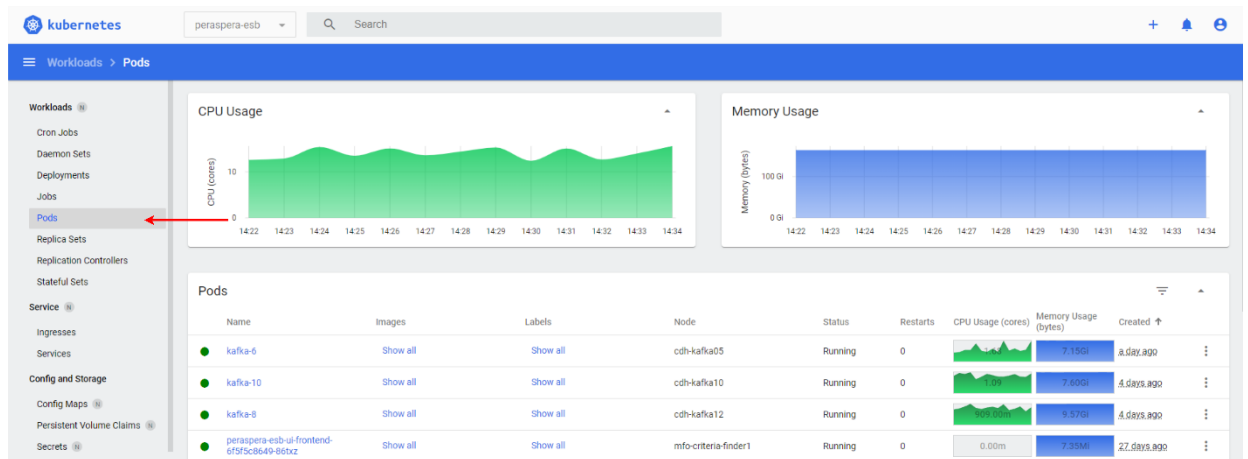


Рисунок 3-14 - Просмотр состояния подов

Здесь отображается использование подами вычислительных ресурсов (процессора и оперативной памяти). Также приводится список подов, работающих в настоящий момент. Пользователь имеет возможность читать журналы работы пода, перезапускать сбойные поды и изменять конфигурацию подов.

Более подробную информацию об администрировании кластера Kubernetes можно получить с официального сайта <https://kubernetes.io/docs>.

3.2.5 Остановка системы

Все консольные команды выполняются на одном master-узле kubernetes, на котором установлена утилита kubectl. На остальных мастер-узлах эти действия не выполняются

Для остановки системы на master-узле kubernetes в консоли следует выполнить следующие команды:

```
kubectl scale --replicas=0 -n peraspera-esb deployment/peraspera-esb-ui-frontend
kubectl scale --replicas=0 -n peraspera-esb deployment/cmak
kubectl scale --replicas=0 -n peraspera-esb deployment/peraspera-esb-ui-backend
kubectl scale --replicas=0 -n peraspera-esb deployment/ksqldb
kubectl scale --replicas=0 -n peraspera-esb deployment/kafka-rest
kubectl scale --replicas=0 -n peraspera-esb deployment/kafka-connect
kubectl scale --replicas=0 -n peraspera-esb deployment/schema-registry
kubectl scale --replicas=0 -n peraspera-esb statefulset/kafka
kubectl scale --replicas=0 -n peraspera-esb statefulset/zookeeper
```

3.2.6 Запуск системы

Для запуска системы на мастер-узле kubernetes в консоли следует выполнить следующие команды:

```
kubectl scale --replicas=1 -n kafka deployment/kafka-ui
```

```
kubectl scale --replicas=1 -n kafka deployment/cmak  
kubectl scale --replicas=1 -n kafka deployment/ksqldb-cli  
kubectl scale --replicas=1 -n kafka deployment/ksqldb  
kubectl scale --replicas=1 -n kafka deployment/kafka-rest  
kubectl scale --replicas=1 -n kafka deployment/kafka-connect  
kubectl scale --replicas=1 -n kafka deployment/schema-registry  
kubectl scale --replicas=3 -n kafka statefulset/kafka  
kubectl scale --replicas=3 -n kafka statefulset/zookeeper
```

3.3 Действия при отключении питания

После отключения питания для повторного включения следует сначала включить master-узлы, а за ними уже worker-узлы.

Проверить работоспособность кластера после включения всех узлов можно, выполнив на мастер узле kubernetes в консоли следующую команду:

```
kubectl get nodes
```

После её выполнения необходимо убедиться, что все узлы кластера находятся в статусе READY.

3.3.1 Просмотр логов

Для просмотра логов на узлах следует воспользоваться следующими командами (выполняются на каждом узле):

```
journalctl -u kubelet -n 10 --no-pager  
journalctl -u docker -n 10 --no-pager
```

Для просмотра логов уровня kubernetes необходимо воспользоваться командой:

```
kubectl logs -n <пространство имен> <имя_пода>
```

3.3.2 Добавление нового узла

3.3.2.1 Установка docker (для ОС CentOS)

Выполнить следующие команды в консоли:

```
sudo yum install -y yum-utils
```

```
sudo yum-config-manager \
```

```
--add-repo \
```

```
https://download.docker.com/linux/centos/docker-ce.repo
```

```
sudo yum install docker-ce docker-ce-cli containerd.io
```

```
sudo systemctl start docker
```

3.3.2.2 Конфигурация docker

- 1) Необходимо создать и прописать daemon.json

```
sudo touch /etc/docker/daemon.json  
sudo bash -c "cat > /etc/docker/daemon.json <<EOF  
{  
  \"exec-opts\": [\"native.cgroupdriver=systemd\"],  
  \"log-driver\": \"json-file\",  
  \"log-opts\": {  
    \"max-size\": \"100m\"  
  },  
  \"storage-driver\": \"overlay2\"  
}  
EOF"
```

- 2) После этого применить изменения

```
sudo systemctl daemon-reload
```

- 3) Затем необходимо перезапустить docker

```
sudo systemctl restart docker
```

3.3.2.3 Выключение swap

Для выключения swap необходимо выполнить следующие команды в консоли:

```
sudo swapoff -a
```

Закомментировать в файле /etc/fstab все строки, где есть swap.

3.3.2.4 Установка kubernetes (для ОС CentOS)

Выполнить следующие команды в консоли:

```
sudo yum install -y apt-transport-https curl  
cat <<EOF > /etc/yum.repos.d/kubernetes.repo  
[kubernetes]  
name=Kubernetes  
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-  
x86_64  
enabled=1
```

```
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
EOF
sudo yum install -y kubelet-1.21.2 kubeadm-1.21.2 kubectl-1.21.2
```

3.3.2.5 Получение команды для присоединения master-узла

На первом мастер-узле kubernetes (ip-адрес 10.0.140.214) в консоли выполнить следующую команду:

```
kubeadm token create --print-join-command --ttl=0
```

3.3.2.6 Подключение master-узла

В консоли master-узла необходимо запустить команду, которая получает certificate-key:

```
sudo kubeadm init phase upload-certs --upload-certs
```

Далее необходимо запустить keepalived:

```
sudo kubeadm init phase upload-certs --upload-certs
```

Для подключения master-узла необходимо запустить команду, которая получена на предыдущем шаге:

```
sudo kubeadm join 172.16.6.162:8443 --token ... --discovery-token-ca-cert-hash sha256:
... --control-plane --certificate-key ...
```

3.3.2.7 Получение команды для присоединения worker-узла

На первом мастер-узле kubernetes (ip-адрес 10.0.140.214) в консоли выполнить следующую команду:

```
kubeadm token create --print-join-command --ttl=0
```

3.3.2.8 Подключение worker-узла

В консоли worker-узла необходимо запустить команду, которая получена на предыдущем шаге:

```
sudo kubeadm join 10.0.140.214:6443 --token ... --discovery-token-
ca-cert-hash sha256:...
```

3.3.2.9 Действия при выходе из строя одного или нескольких узлов

При выходе узла из строя после устранения его неисправности необходимо:

1) Запустить на нем docker и kubelet:

```
systemctl start docker
```

```
systemctl start kubelet
```

2) Проверить, что узел в кubernetes получил статус Ready:

```
kubectl get nodes
```

Если узел не подключился, необходимо отключить узел от кластера:
`kubeadm reset`

Затем повторить действия по добавлению нового узла, описанные в п.3.3.2.

3.3.2.10 Действия при замене вышедшего из строя узла

При замене вышедшего из строя узла необходимо на новом узле выполнить действия по добавлению нового узла, описанные в п.3.3.2.

3.3.3 Управление из консоли (перечень команд)

Для управления из консоли необходимо перейти на мастер-узел `kubernetes`.

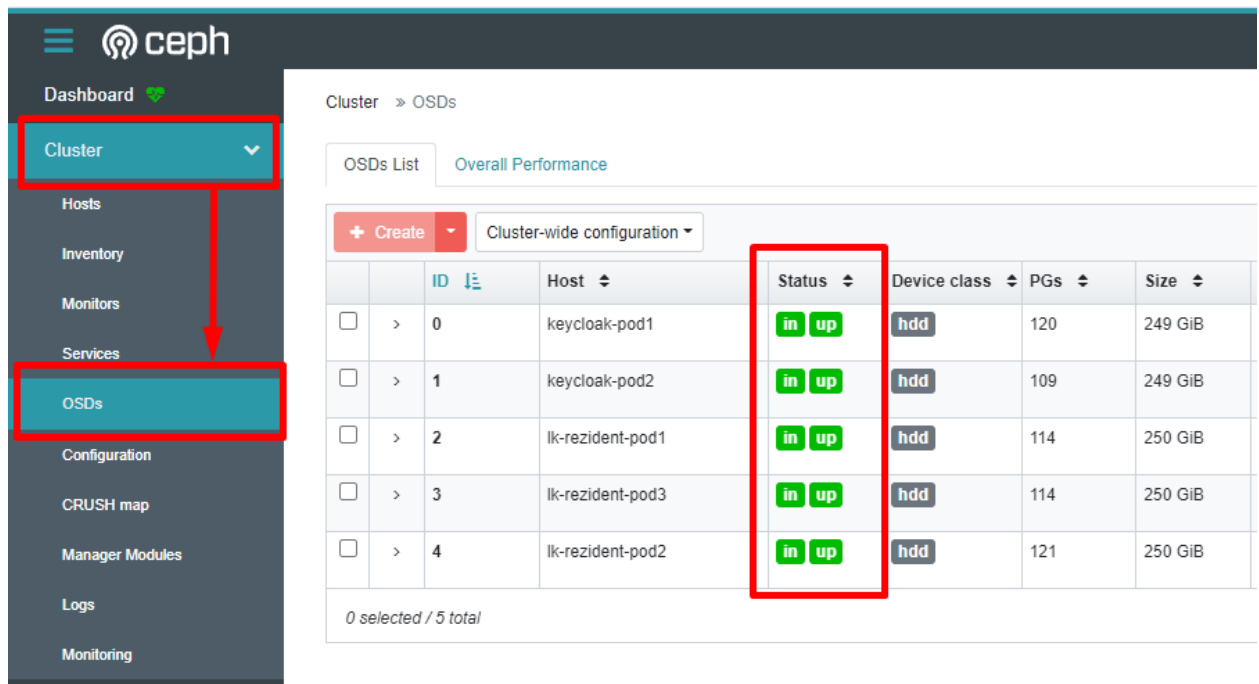
С командами управления можно ознакомиться на официальном сайте `kubernetes` (<https://kubernetes.io/>).

Наиболее часто используемая утилита - `kubectl`. Список самых полезных и часто используемых команд можно также найти на официальном сайте (<https://kubernetes.io/ru/docs/reference/kubectl/cheatsheet/>).

3.3.4 Мониторинг состояния Ceph

Для просмотра состояния Ceph необходимо открыть Ceph Dashbord. Необходимо убедиться, что в блоке Cluster Status отображается статус `HEALTH_OK`.

Для просмотра состояния устройств хранения объектов необходимо перейти на вкладку Cluster->OSDs и убедиться, что все устройства находятся в статусе `IN` и `UP`.



ID	Host	Status	Device class	PGs	Size
0	keycloak-pod1	in up	hdd	120	249 GiB
1	keycloak-pod2	in up	hdd	109	249 GiB
2	lk-rezident-pod1	in up	hdd	114	250 GiB
3	lk-rezident-pod3	in up	hdd	114	250 GiB
4	lk-rezident-pod2	in up	hdd	121	250 GiB

Рисунок 3-15 – Просмотр состояния устройств хранения объектов

В случае наличия ошибок или нештатных состояний для анализа и устранения ошибок следует обратиться к официальной документации Ceph <https://docs.ceph.com/en/pacific/>.

В случае, если недоступен web-интерфейс Ceph, следует открыть `kubernetes dashboard` и перейти в пространство имен `rook-ceph`.

Среди подов в данном пространстве имен необходимо найти под с префиксом имени `rook-ceph-mgr-a`. Если он находится в нештатном состоянии, необходимо прочитать информацию о проблеме и устранить её. После чего необходимо перезапустить под.

Также можно прочитать журнал логов пода из консоли.

Для этого на master-узле `kubernetes` необходимо воспользоваться командой:

```
kubectl logs -n rook-ceph rook-ceph-mgr-a-799c47cf67-xsr48 --tail 100
```

Суффикс имени пода может отличаться.

3.4 Мониторинг состояния программы «PERASPERA ESB » версия 2.0

Мониторинг состояния программы «PERASPERA ESB » версия 2.0 обеспечивается сервисами Kubernetes Dashboard и Peraspera ESB UI.

В сервисе Kubernetes Dashboard можно наблюдать за состоянием вычислительных ресурсов для каждого компонента кластера. При этом для списка компонентов выводится суммарная статистика.



Рисунок 3-16 – Просмотр состояния вычислительных ресурсов компонент кластера

Для каждого пода кластера также можно получить статистику использования им вычислительных ресурсов. Кроме того, статистика отображается и в таблице.

Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
Running	0	2.00m	394.10Mi	4 hours ago
Completed	0	-	-	4 days ago
Running	0	1.00m	19.84Mi	6 days ago

Рисунок 3-17 – Статистика использования ресурсов

В случае возникновения проблемных ситуаций следует обратить внимание на статус пода. Ошибочные статусы разбираются пользователем программы «PERASPERA ESB ». В этом случае следует открыть детальную информацию о сбойном поде и проанализировать сообщение об ошибке.

Журнал работы пода используется для анализа его текущего состояния.

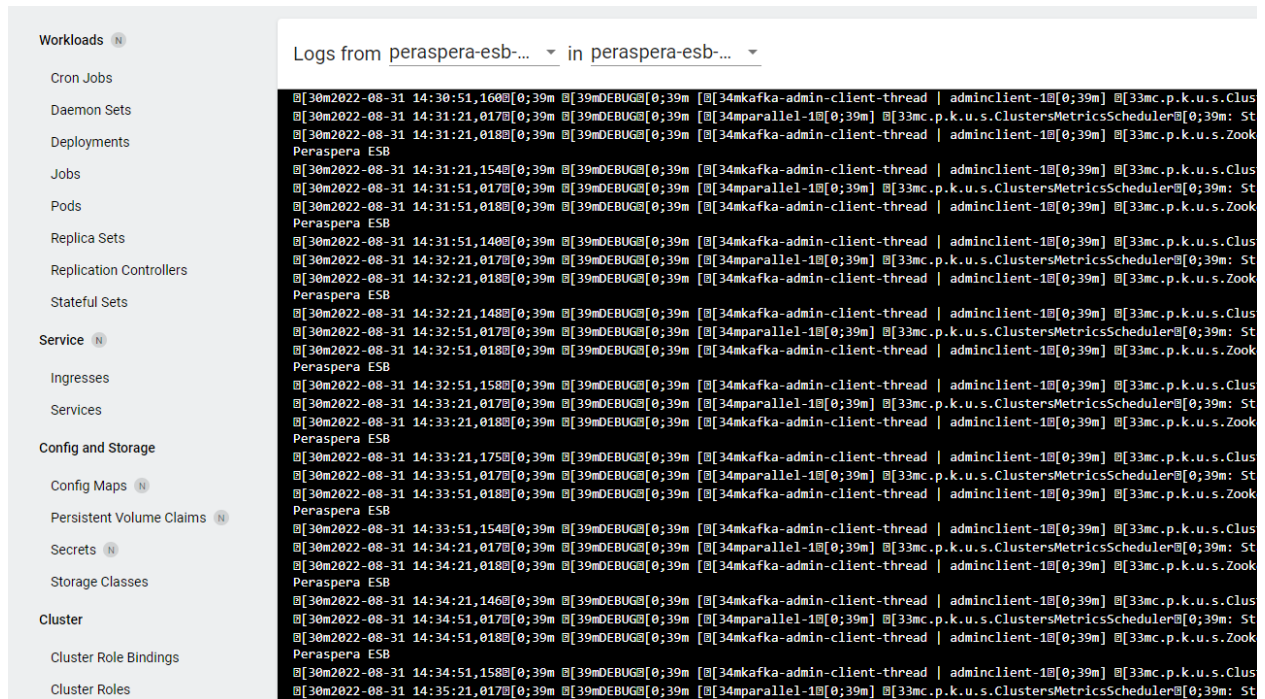


Рисунок 3-18 - Журнал работы пода

Состояние коннекторов можно проверить с помощью Peraspera ESB UI. На закладке «Коннекторы» отображается информация о состоянии каждого коннектора.

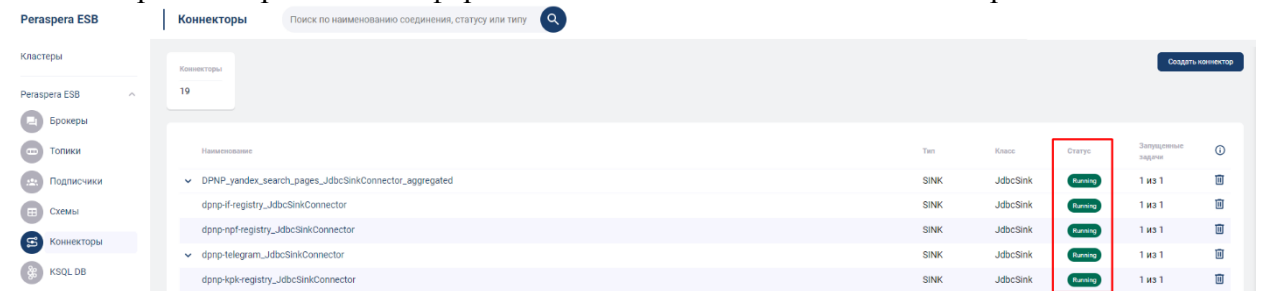


Рисунок 3-19 – Просмотр информации о состоянии коннекторов

Перейдя по ссылке, можно увидеть причину ошибки в случае ошибочного статуса.

3.5 Конфигурирование и диагностика коннекторов

Организация информационного взаимодействия систем-источников данных и систем-приемников данных посредством программы «PERASPORA ESB» осуществляется с использованием коннекторов.

Ниже приведено описание конфигурирования и диагностики коннекторов.

3.5.1 Конфигурирование коннекторов

Конфигурирование коннектора, т.е. создание нового коннектора, удаление коннектора или изменение существующего коннектора, выполняется с использованием интерфейса командной строки ksqlDB (KSQL CLI) или с помощью пользовательского интерфейса Peraspera ESB UI.

ksqlDB – это база данных потоковой передачи событий, построенная по архитектуре клиент-сервер. API ksqlDB позволяет запрашивать и обрабатывать данные, хранящиеся в «топиках» Apache Kafka с помощью типового инструментария SQL-запросов.

3.5.1.1 Создание коннектора

Создание коннектора осуществляется с помощью «Мастера создания коннекторов» в Peraspera ESB UI. Для этого необходимо:

- 1) Во вкладке «Коннекторы» нажать на кнопку «Создать коннектор».

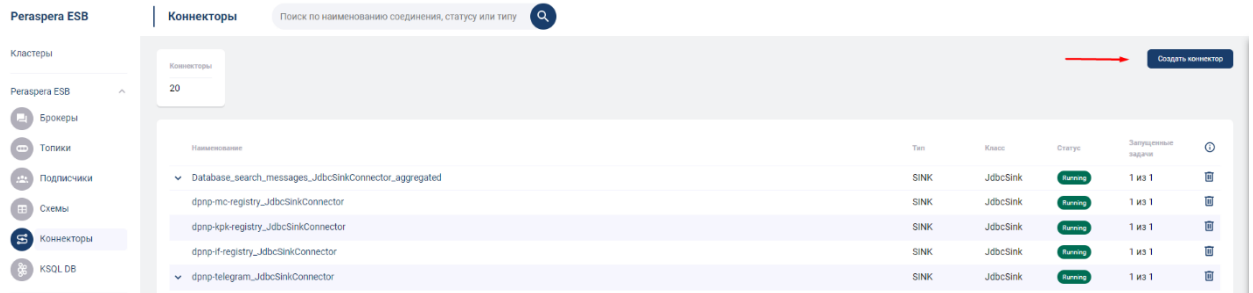


Рисунок 3-20 – Кнопка «Создать коннектор»

- 2) Заполнить все поля для шага «Тип», в частности указать:
 - наименование коннектора;
 - тип коннектора;
 - класс коннектора;
 - максимальное число задач;
 - действие при изменении конфигурации коннектора.

Далее нажать «Сохранить и продолжить»

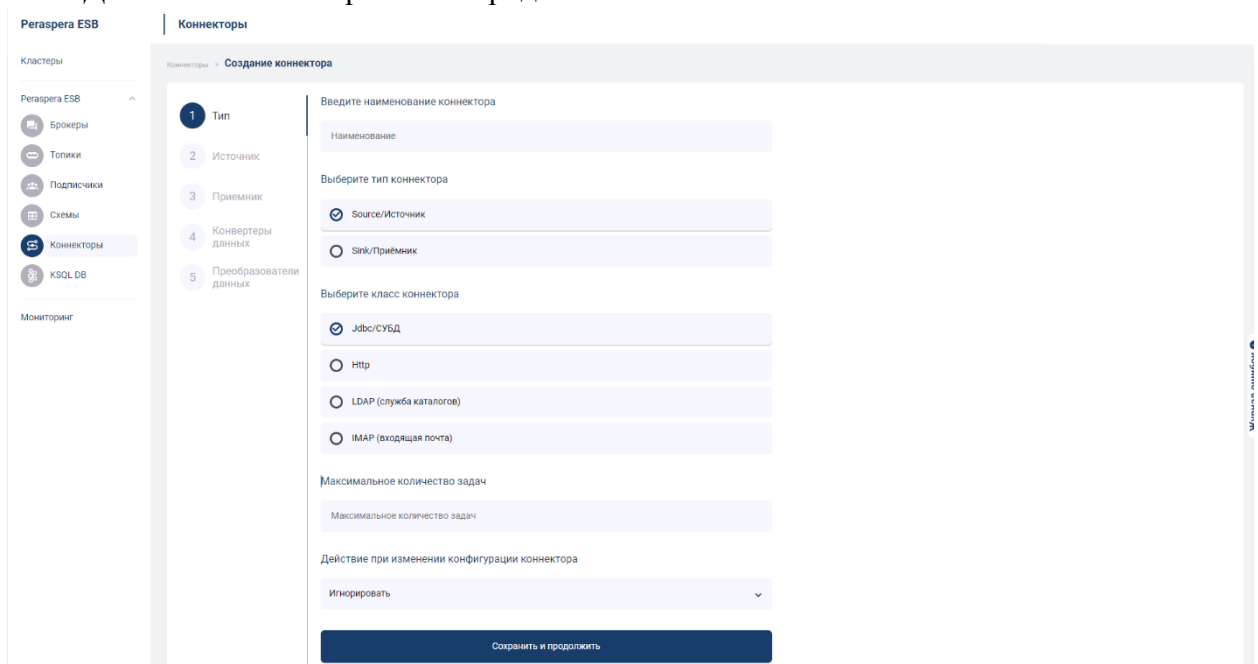


Рисунок 3-21 - Заполнение полей для шага «Тип»

- 3) Заполнить все поля шага «Источник»
Далее нажать «Сохранить и продолжить»

Peraspera ESB | Коннекторы

Классеры

Peraspera ESB

- Брокеры
- Топики
- Подписчики
- Схемы
- Коннекторы**
- KSQL DB

Мониторинг

Коннекторы > Создание коннектора

1 Тип

2 Источник

3 Приемник

4 Конвертеры данных

5 Преобразователи данных

Адрес JDBC подключения

Адрес JDBC подключения

Имя пользователя

Имя пользователя

Пароль подключения

Попытка создания JDBC подключения

3

Время задержки между попытками подключения

Паттерн каталога

Паттерн каталога

Включенные таблицы

Исключенные таблицы

Исключенные таблицы

Паттерн схемы

Округлять значения (устаревшее)

Не округлять значения (устаревшее)

Округлять значения

Округлять значения

Режим проверки новых записей

Имя столбца с временной меткой

Имя столбца с временной меткой

Проверить столбцы на пол null

Не проверять столбцы на пол null

Имя столбца с увеличивающимися значениями

Имя столбца с увеличивающимися значениями

Начальное время

Начальное время

SQL запрос

SQL запрос

Рисунок 3-22 - Заполнение полей для шага «Источник»

4) Заполнить все поля шага «Приемник»
Далее нажать «Сохранить и продолжить»

Peraspera ESB | Коннекторы

Классеры

Peraspera ESB

- Брокеры
- Топики
- Подписчики
- Схемы
- Коннекторы**
- KSQL DB

Мониторинг

Коннекторы > Создание коннектора

1 Тип

2 Источник

3 Приемник

4 Конвертеры данных

5 Преобразователи данных

Auto topic creation - Автоматическое создание топиков

Префикс имен таблиц

Префикс имен таблиц

Коэффициент репликации

Коэффициент репликации

Количество разделов

Количество разделов

Включать топик в группу

Включить топик в группу

Исключать топик из группы

Исключить топик из группы

Политика хранения журнала

delete

Количество сообщений записываемых в журнал до сброса

Количество сообщений записываемых в журнал до сброса

Время, в течение которого журнал может содержать данные, прежде чем они будут сброшены

Время, в течение которого журнал может содержать данные, прежде чем они будут сброшены

Максимальный размер сообщения

Максимальный размер сообщения

Минимальное количество сгруппированных реплик

1

Максимальный размер партиции

Максимальный размер партиции

Максимальное время хранения журнала

Максимальное время хранения журнала

Максимальный размер сегмента

Максимальный размер сегмента

Рисунок 3-23 - Заполнение полей для шага «Приемник»

5) Заполнить все поля шага «Конвертеры данных»
Далее нажать «Сохранить и продолжить»

Рисунок 3-24 - Заполнение полей для шага «Конвертеры данных»

- 6) Заполнить все поля шага «Преобразователи данных»
Далее нажать «Отправить»

Рисунок 3-25 - Заполнение полей для шага «Преобразователи данных»

После нажатия кнопки «Отправить» будет создан новый коннектор в соответствии с указанными в Мастере настройками пользователя.

3.5.1.2 Удаление коннектора

Удаление коннектора осуществляется с помощью пользовательского интерфейса Peraspera ESB UI.

Для удаления коннектора необходимо на вкладке «Коннекторы» нажать на элемент удаления (корзина) справа от имени коннектора



Рисунок 3-26 - Элемент удаления на вкладке «Коннекторы»

3.5.1.3 Изменение существующего коннектора

Изменение ранее созданного коннектора осуществляется с использованием пользовательского интерфейса Peraspera ESB UI.

Во вкладке «Коннекторы» необходимо выбрать коннектор, который следует изменить. Далее в правом верхнем углу пользовательского интерфейса нажать на кнопку «Изменить настройки»

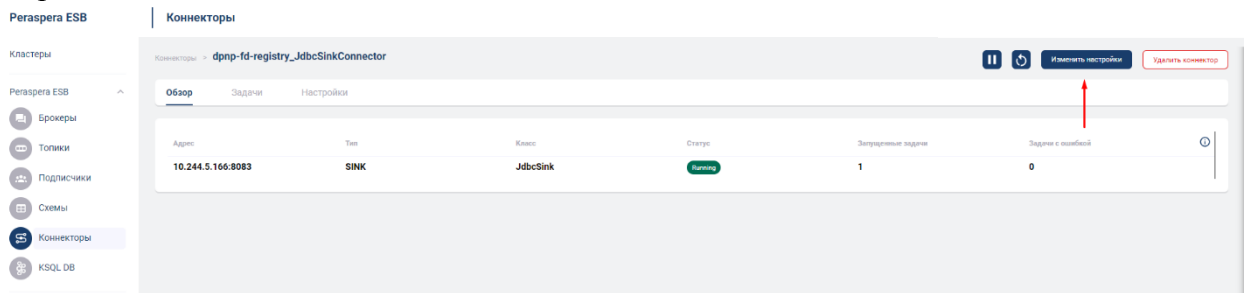


Рисунок 3-27 - Кнопка «Изменить настройки»

С помощью элемента удаления (корзина) можно удалить конкретную настройку коннектора



Рисунок 3-28 - Элемент удаления настройки коннектора

С помощью значка пл «+» можно добавить настройку коннектора

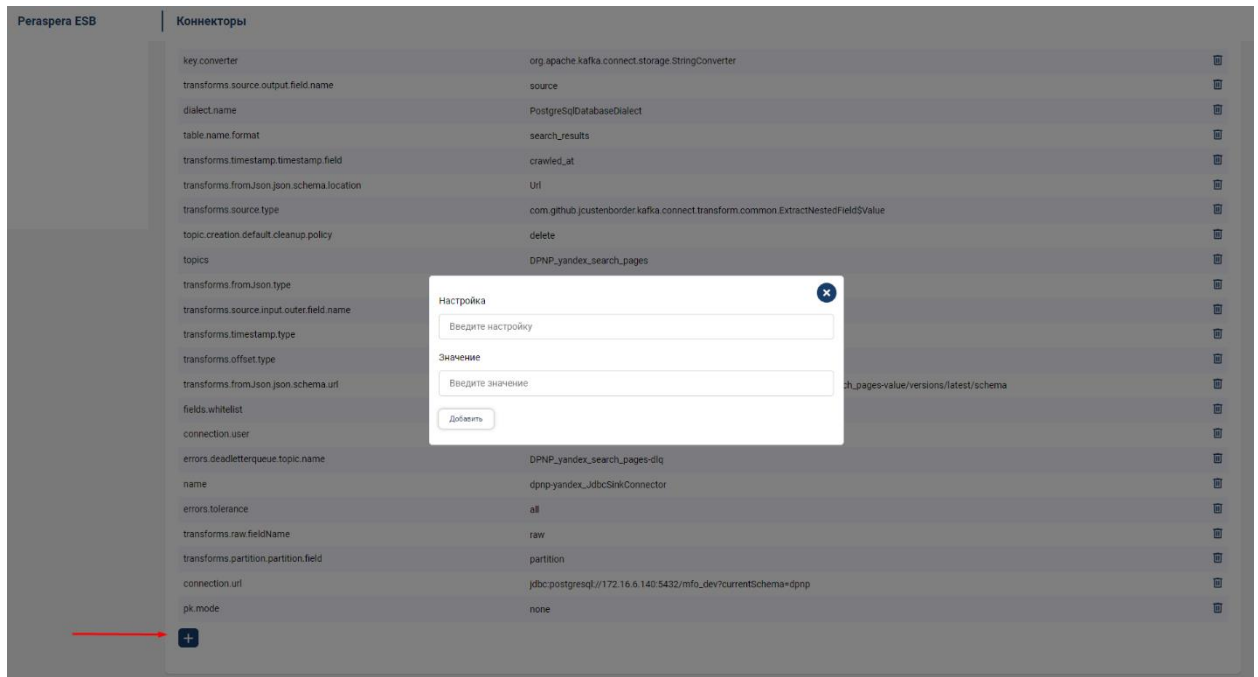


Рисунок 3-29 - Добавление настройки коннектора

По завершению изменений коннектора необходимо нажать на кнопку «Сохранить изменения» внизу окна



Рисунок 3-30 - Кнопка «Сохранить изменения»

Поставить выполнение коннектора на паузу или перезапустить коннектор можно с помощью соответствующих кнопок пользовательского интерфейса

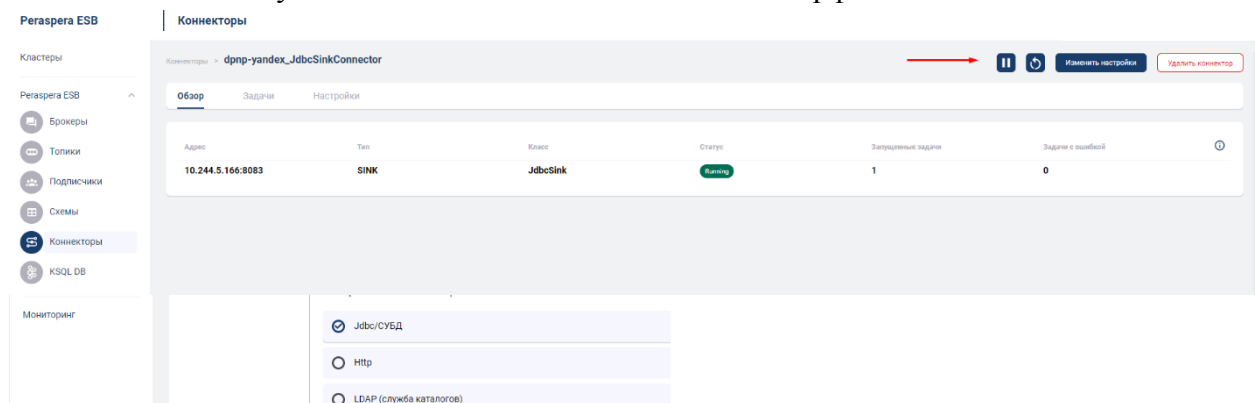


Рисунок 3-31 - Кнопка временной остановки выполнения коннектора (постановки на паузу)

3.5.2 Диагностика коннекторов

Для диагностики работы коннекторов, как самостоятельных единиц, могут использоваться следующие инструменты:

- 1) Пользовательский интерфейс Peraspera ESB UI, в котором на вкладке «Коннекторы» отображаются состояния запущенных коннекторов в столбце «Статус». Возможные состояния: RUNNING, FAILED.

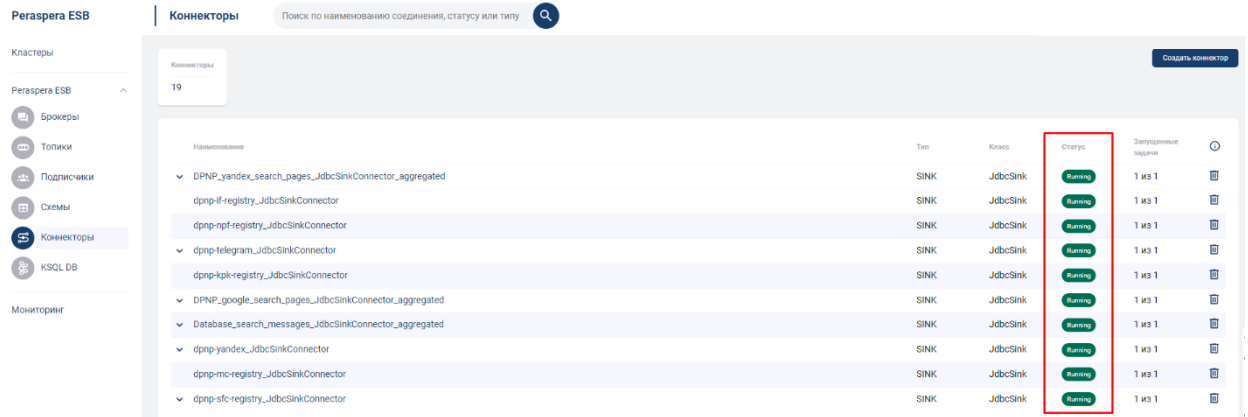


Рисунок 3-32 - Отображение состояния запущенных коннекторов

2) Сервис Kubernetes Dashboard, который предоставляет возможность просматривать лог-сообщения, выводимые запущенными коннекторами.

На основе выводимых лог-сообщений можно делать выводы о ходе работы запущенных коннекторов, отслеживать промежуточные состояния работы запущенных задач, видеть результаты работы завершенных задач и получать детальную информацию об ошибках, в случае их возникновения.

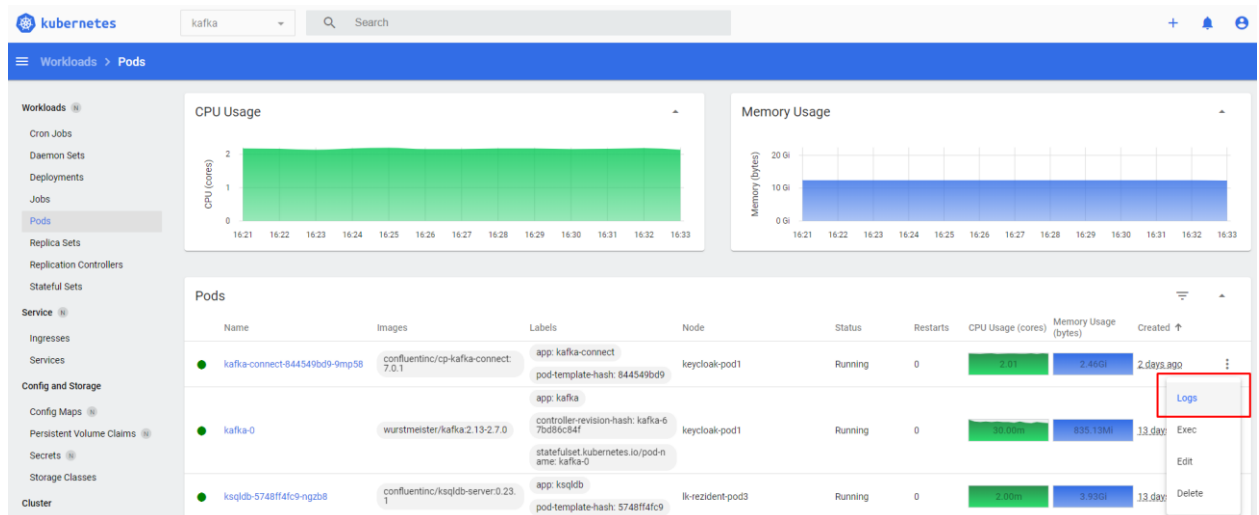


Рисунок 3-33 - Отображение лог-сообщений, выводимых запущенными коннекторами

Также, диагностировать работу коннекторов можно опираясь на результат их работы, в частности:

1) Для Source коннекторов - Увеличивающееся число и объем сообщений, хранимых в «топиках». Значения этого показателя отображается в пользовательском интерфейсе Peraspera ESB UI на вкладке «Топики».

Peraspera ESB | Топики | Поиск по наименованию топика

Кластеры

Peraspera ESB

- Брокеры
- Топики**
- Подписчики
- Схемы
- Коннекторы
- KSQL DB

Мониторинг

Показывать внутренние топики

Наименование топика	Количество разделов	Репликаций не синхронизировано	Количество репликаций	Количество сообщений	Размер
☐ DEV_DPNP_yandex_search_pages	120	240	2	1509	3491661
☐ DPNP_google_search_pages	12	12	1	339684	428601964
☐ DPNP_google_search_pages-dlq	1	3	3	0	0
☐ DPNP_google_search_pages_2206	120	240	2	6687	17666799
☐ DPNP_google_search_pages_2207	120	240	2	2077	6180658
☐ DPNP_queries_google	120	240	2	90	67622
☐ DPNP_queries_google_2206	120	240	2	12	7866
☐ DPNP_queries_google_2207	120	240	2	12	7944
☐ DPNP_queries_yandex_2206	120	240	2	0	0
☐ DPNP_queries_yandex_2207	120	240	2	12	7927
☐ DPNP_telegram_search_messages	12	12	1	509	1604092
☐ DPNP_telegram_search_messages-dlq	1	3	3	0	0
☐ DPNP_yandex_search_pages	12	36	3	539	1855696
☐ DPNP_yandex_search_pages-dlq	1	3	3	133	1775034
☐ DPNP_yandex_search_pages_2206	120	240	2	14862	39041179
☐ DPNP_yandex_search_pages_2207	120	240	2	2253	7327580
☐ TEST_DPNP_google_search_apps	120	240	2	23	7898
☐ TEST_DPNP_google_search_apps_pages	120	240	2	2065	4205408
☐ TEST_DPNP_yandex_search_apps	120	240	2	21	7563
☐ TEST_DPNP_yandex_search_apps_pages	120	240	2	4266	11772924
☐ TEST_MFO_ok	120	240	2	383	1702954
☐ TEST_MFO_process_all	120	240	2	15	10585

Рисунок 3-34 - Отображение числа и объема сообщений

2) Для JDBC Sink коннекторов - Создание таблиц и обновление данных в связанных с коннекторами таблицах базы данных.

3) Для HTTP Sink коннекторов - HTTP-запросы, отправляемые коннекторами, содержащие сообщения из «топиков».

3.5.3 Действия при недоступности web-интерфейса Peraspera ESB UI

В случае, если недоступен web-интерфейс Peraspera ESB UI, следует открыть kubernetes dashboard и перейти в пространство имен kafka.

Среди подов в данном пространстве имен необходимо найти под с префиксом имени kafka-ui. Если он находится в нештатном состоянии, необходимо прочитать информацию о проблеме и устранить её. После чего необходимо перезапустить под.

Также можно прочитать журнал логов пода из консоли.

Для этого на master-узле kubernetes необходимо воспользоваться командой:

```
kubect1 logs -n kafka kafka-ui-64f5b75cbb-t8ng7 kafka-ui --tail 100
```

Суффикс имени пода может отличаться.

3.6 Нештатные ситуации

В случае возникновения нештатной ситуации пользователь программы «PERASPERA ESB »:

- связывается с техническим персоналом, обслуживающим КТС программы «PERASPERA ESB », для устранения технических неисправностей комплекса технических средств;

- после устранения технических неисправностей связывается с администраторами общего программного обеспечения для проверки ими работоспособности соответствующего программного обеспечения;

- проверяет работоспособность программного обеспечения программы «PERASPERA ESB »;

- проверяет сохранность и доступность информационных ресурсов;

- при нарушении целостности информационных ресурсов инициирует восстановление информационных ресурсов с использованием резервных копий.

[illegible][illegible]

