



EVD Platform V Synapse Event Transfer Service

EVTD Сервис передачи сообщений

Описание функциональных характеристик

ОГЛАВЛЕНИЕ

Описание функций	3
Термины и определения	3
Назначение.....	3
Цель создания	3
Основные функции (сущностные)	3
Варианты использования	5
Сценарии использования	5

Описание функций

Термины и определения

Термин/Аббревиатура	Определение
URI (Uniform Resource Identifier)	Унифицированный идентификатор ресурса

Назначение

Цель создания

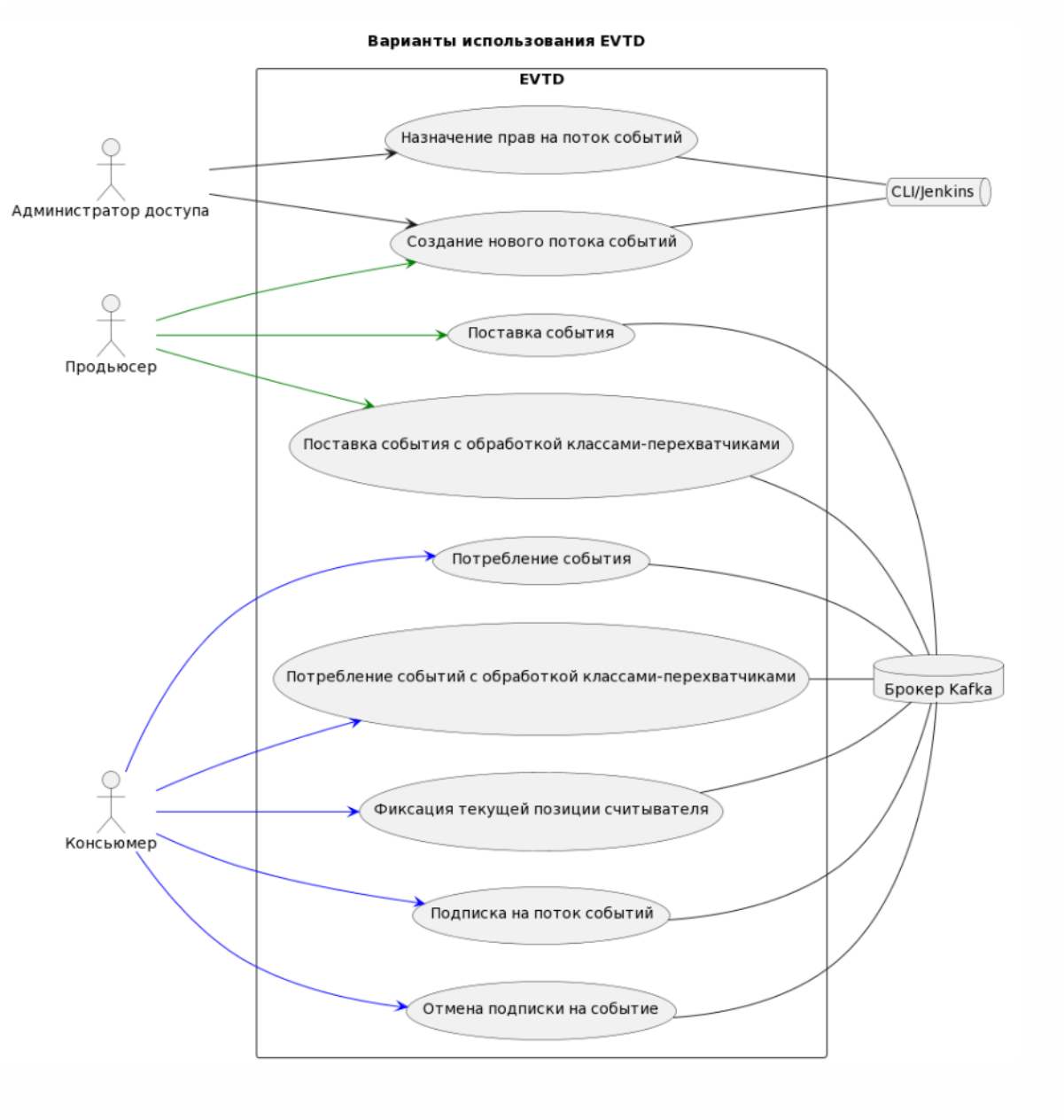
Сервис передачи событий служит для передачи публикуемых поставщиками событий потребителям в рамках одного событийного домена Synapse (EVD).

Основные функции (сущностные)

Название функции	Потребитель функции	Аргументы функции	Результат
Поставка события	Поставщик	URI потока событий, событие	Событие успешно записано в сервис передачи событий; ошибка при записи события
Потребление событий	Потребитель	URI потока событий	Событие успешно получено; пустой список, при отсутствии новых событий; ошибка при получении события

Название функции	Потребитель функции	Аргументы функции	Результат
Подписка на события	Потребитель	URI потока событий	Успешная подписка на поток событий; ошибка при подписке
Отмена подписки на события	Потребитель	URI потока событий	Успешная отмена подписки на поток событий; ошибка при отмене подписки
Фиксация текущей позиции считывания	Потребитель	URI потока событий	Успешная фиксация текущей позиции считывания в потоке; ошибка при фиксации позиции
Поставка события с обработкой классами-перехватчиками	Поставщик с подключенным перехватчиком	URI потока событий, событие, конфигурация перехватчика	Событие успешно записано в сервис передачи событий; ошибка при записи события
Потребление событий с обработкой классами-перехватчиками	Потребитель с подключенным перехватчиком	URI потока событий, конфигурация перехватчика	Событие успешно получено; пустой список, при отсутствии новых событий; ошибка при получении события

Варианты использования



Сценарии использования

1. Поставка события.

Поставщик вызывает метод *Send* клиента Apache Kafka Producer для отправки события.

2. Потребление событий.

Потребитель вызывает метод *poll* клиента Apache Kafka Consumer для потребления событий, на которые он подписан.

3. Подписка на события.

Потребитель вызывает метод *subscribe* клиента Apache Kafka Consumer для установки подписки на необходимые события.

4. Отмена подписки на события.

Потребитель вызывает метод *unsubscribe* клиента Apache Kafka Consumer для отмены подписки.

5. Фиксация текущего позиции считывания.

Потребитель вызывает метод *CommitSync* или *CommitAsync* клиента Apache Kafka Consumer для фиксации текущей позиции чтения.

6. Поставка события с обработкой классами-перехватчиками.

Поставщик вызывает метод *Send* клиента Apache Kafka Producer для отправки события, событие проходит через класс-перехватчик, где, в зависимости от используемого класса, могут быть проведены следующие действия:

- Валидация события по схеме (JSON, XML);
- Подпись события с помощью X.509 сертификата или OTT-токена;
- Обогащение заголовка события временной меткой отправки в ТС передачи событий и идентификатором для трассировки задержки.

7. Потребление событий с обработкой классами-перехватчиками.

Потребитель вызывает метод *poll* клиента Apache Kafka Consumer для потребления событий, на которые он подписан. До передачи в прикладной код события проходят через класс-перехватчик, где, в зависимости от используемого класса могут быть проведены следующие действия:

- Валидация события по схеме (JSON, XML);
- Проверка подписи события с помощью X.509 сертификата или OTT-токена;
- Расчет и трассировка задержки полученного события с записью в журнал (информация ограничена идентификатором события и временными метриками, бизнес-информация не журналируется);
- Фильтрация атрибутивного состава события по JSON-path или XML-path с передачей в прикладной код только атрибутов, попадающих в white-list, указанный в конфигурации;
- Фильтрация атрибутивного состава события на основании DSL-правил.