



УДК 004.03
doi: 10.21685/2587-7704-2023-8-1-5



Open
Access

RESEARCH
ARTICLE

Микросервис исполнения активных правил

Ярослав Сергеевич Шлепнев

Пензенский государственный университет, Россия, г. Пенза, ул. Красная, 40
yaroslav.shlepnev@yandex.ru

Сергей Владимирович Шибанов

Пензенский государственный университет, Россия, г. Пенза, ул. Красная, 40
serega@pnzgu.ru

Александр Сергеевич Гусаров

Пензенский государственный университет, Россия, г. Пенза, ул. Красная, 40
alexandergusarovv@gmail.com

Аннотация. Представлены функциональные возможности микросервиса исполнения активных правил в составе сервиса потоковой обработки событий для реализации информационно-управляющих систем. Приведена архитектура микросервиса. Представлен сценарий работы микросервиса в процессе реагирования на поступающие события с возможностью выявления и устранения коллизий взаимодействия активных правил. Анализ взаимодействия активных правил выполняется с применением раскрашенных сетей Петри. Для управления потоками событий используется платформа Apache Kafka. Для работы с локальным репозиторием метаданных используется Inmemory-СУБД Redis.

Ключевые слова: потоки событий, активные правила, микросервисная архитектура, коллизии исполнения активных правил, Apache Kafka

Для цитирования: Шлепнев Я. С., Шибанов С. В., Гусаров А. С. Микросервис исполнения активных правил // Инжиниринг и технологии. 2023. Т. 8 (1). С. 1–5. doi: 10.21685/2587-7704-2023-8-1-5

Microservice for the execution of active rules

Jaroslav S. Shlepnev

Penza State University, 40 Krasnaya Street, Penza, Russia
yaroslav.shlepnev@yandex.ru

Sergey V. Shibanov

Penza State University, 40 Krasnaya Street, Penza, Russia
serega@pnzgu.ru

Alexander S. Gusarov

Penza State University, 40 Krasnaya Street, Penza, Russia
alexandergusarovv@gmail.com

Abstract. The functional capabilities of the active rules execution microservice as part of the event streaming processing service for the implementation of information management systems are presented. The architecture of the microservice is given. The scenario of the microservice operation in the process of responding to incoming events with the possibility of identifying and eliminating collisions of interaction of active rules is presented. The analysis of the interaction of active rules is performed using colored Petri nets. The Apache Kafka platform is used to manage event flows. To work with the local metadata repository, the In memory DBMS Redis is used.

Keywords: event streams, active rules, micro service architecture, collisions of execution of active rules, Apache Kafka

For citation: Shlepnev Y.S., Shibanov S.V., Gusarov A.S. Microservice for the execution of active rules. *Inzhiniring i tekhnologii = Engineering and Technology*. 2023;8(1):1–5. (In Russ.). doi: 10.21685/2587-7704-2023-8-1-5



Микросервис исполнения активных правил является составной частью сервиса потоковой обработки событий и исполнения активных правил, который, в свою очередь, предназначен для непрерывной обработки атомарных и сложных событий, оперативного реагирования на обнаруженные события с использованием механизма активных правил. В качестве систем наблюдения и управления могут выступать базы и хранилища данных, технические системы, программные приложения.

Сервис представляет собой сложную многофункциональную программную платформу, реализующую не только обработку входных атомарных и вычисляемых сложных событий, исполнение активных правил, но и конструирование и управление метаданными о событиях и активных правил, анализ корректности активных правил в процессе конструирования и исполнения, устранение возможных коллизий, журнализацию событий и правил, анализ данных журналов, аудит и настройку [1].

Сервис построен в событийно-ориентированной микросервисной архитектуре в виде независимых, масштабируемых компонентов, каждый из которых может быть разработан с применением собственного стека технологий [2]. Для получения событий от наблюдаемых систем, передачи событий и данных между микросервисами, реализации управляемых воздействий к системам управления используются механизмы очередей и платформа Apache Kafka с применением фреймворка Kafka Stream [3].

Важнейшими характеристиками информационно-управляющих систем являются непрерывность управления, надежность функционирования, выработка управляющих воздействий в реальном масштабе времени. Непрерывность и надежность обработки событий и данных от систем наблюдения обеспечивается за счет применения технологии и средств потоковой обработки данных. Выработка управляющих воздействий реализуется на основе механизма активных правил. Точно такие же требования непрерывности управления, надежности и поддержка реального времени предъявляются к программным средствам исполнения активных правил [4].

Программные средства исполнения активных правил в составе сервиса реализованы в виде отдельного микросервиса, построенного по единой схеме. Каждый из микросервисов обрабатывает поступающие на вход потоки событий с использованием сервисов очередей Apache Kafka, обеспечивающих надежную связь и оперативный обмен информацией между микросервисами в режиме реального времени. Бизнес-логика микросервиса обрабатывает входные события в соответствии с решаемыми задачами и передает выработанные данные и события другим микросервисам через очереди сообщений. Алгоритмы бизнес-логики микросервиса построены с применением метаданных, хранящихся в локальном репозитории на стороне микросервиса. Локальный репозиторий метаданных реализован в виде key-value базы данных под управлением Inmemory-СУБД Redis [5]. В случае необходимости может выполняться журнализация обрабатываемых данных, событий и правил.

Исходные атомарные события, получаемые от систем наблюдения в виде потоков данных, поступают в микросервис обработки входных событий, который выполняет предварительную обработку, первичный анализ событий и перенаправляет их в соответствующие микросервисы для дальнейшей обработки. Кроме того, микросервис обработки входных событий может сгенерировать новые события за счет агрегации событий из входного потока с помощью непрерывных запросов [6].

Если с исходными или сгенерированными событиями ассоциированы активные правила, то события передаются в микросервис исполнения активных правил. Если события являются элементом составного события, то они передаются в микросервис обнаружения составных событий. Если составное событие обнаруживается в микросервисе составных событий, и с ним ассоциированы активные правила, то оно также передается в микросервис исполнения активных правил.

К основным функциональным возможностям микросервиса исполнения активных правил относятся:

- обработка исходных потоков событий, поступающих от микросервиса обработки входных событий от систем наблюдения и микросервиса обработки составных событий;
- активация управляющих воздействий за счет исполнения активных правил, связанных с поступившими событиями;
- динамический анализ активных правил в процессе их исполнения с целью выявления и устранения возможных коллизий;
- передача управляющих воздействий в виде исполняемых команд в выходную очередь.

Для поддержки основных функций реализован ряд вспомогательных функций:

- чтение метаданных из локального репозитория в память процесса;
- передача информации о поступивших и обработанных событиях, исполненных активных правил, о нештатных ситуациях в микросервис журнализации.

На рис. 1 представлена архитектура микросервиса исполнения активных правил.

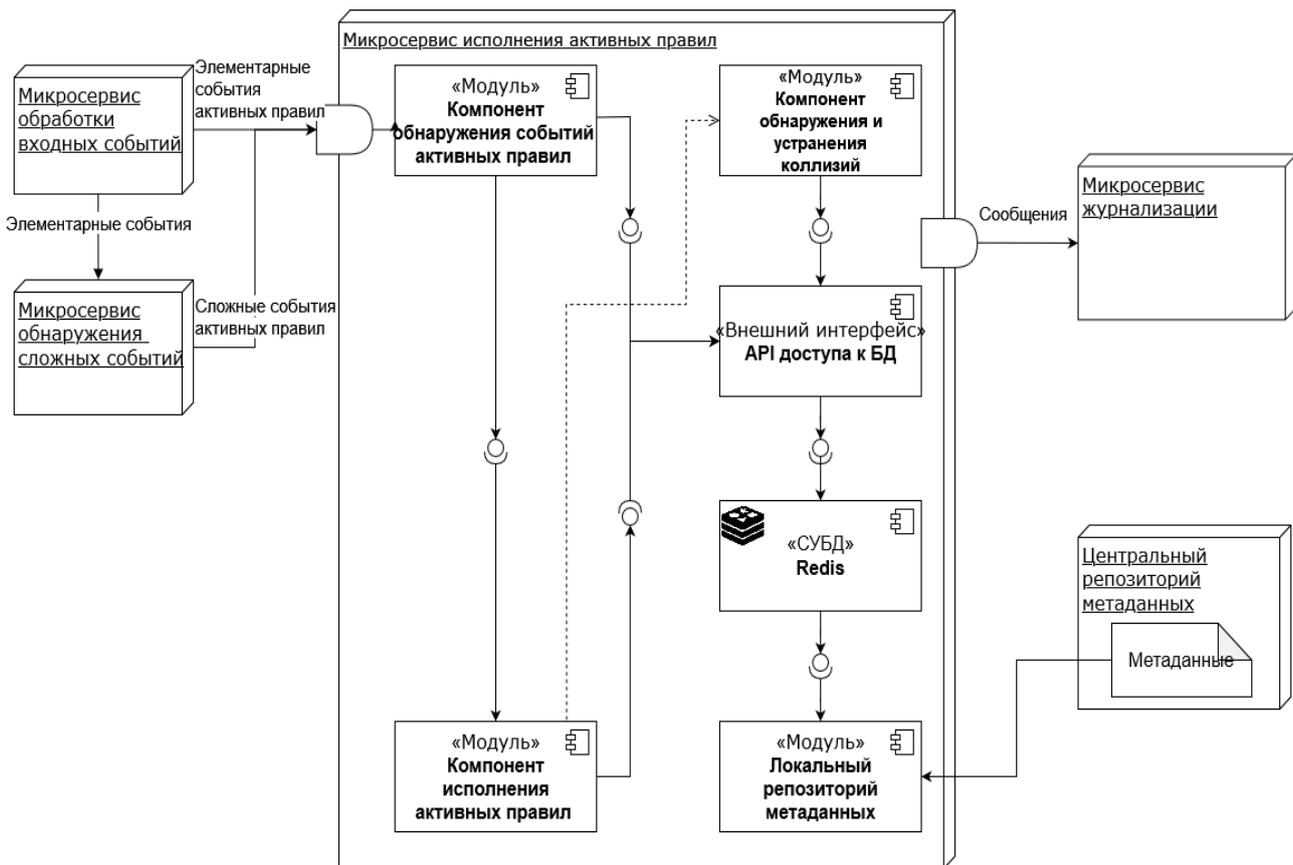


Рис. 1. Архитектура микросервиса исполнения активных правил

Микросервис исполнения активных правил принимает входной поток событий, производит поиск активных правил, ассоциированных с конкретным типом события, в локальном репозитории метаданных и запускает программные потоки исполнения правил и анализа возможных коллизий. Если коллизии обнаруживаются, предпринимается попытка их устранения с наименьшими допустимыми издержками. Если устранить коллизии из-за их критичности невозможно, исполнение правил отменяется, цепочка правил откатывается.

Перед непосредственным исполнением правила микросервис выполняет подготовительные действия, заключающиеся в определении контекстов поступивших событий, значений параметров, а также выполняет создание программных структур в памяти процесса. Модуль «исполнения активных правил» используется для выработки воздействий в зависимости от набора связанных с типом события управляющих правил.

В процессе исполнения активных правил в режиме динамического анализа модулем обнаружения и устранения коллизий запускается параллельный поток анализа [7]. В этот момент в память процесса уже загружена информация об активных правилах из репозитория метаданных, а также сформированы вспомогательные для анализа структуры. Исходя из концептуальных особенностей активных правил в качестве формальной модели представления взаимодействующих правил использовать условно раскрашенные сети Петри (ССРН) [8]. С целью сохранения информации о событии в маркерах и для создания новых маркеров, представляющих собой данные об активизируемом действии активного правила, условно раскрашенная сеть Петри использует концепцию «окраски», заимствованную от раскрашенных сетей Петри. Структура сети создается на этапе статического анализа, а микросервис исполнения активных правил использует данную модель в процессе динамического анализа.

Сценарий работы микросервиса исполнения активных правил описывается последовательностью действий (рис. 2).

1. Загрузка информации репозитория в память процесса.
2. Валидация программной структуры раскрашенной сети Петри.
3. Для каждого события из потока поступающих событий:
 - 1) идентификация типов событий, значений параметров контекста, учет сложных событий;
 - 2) запуск параллельного потока анализа (выявления коллизий);



- 3) запуск потока исполнения с отправкой запросов к потоку анализа;
- 4) выполнение управляющих воздействий;
- 5) аудит всех событий потоков исполнения и анализа;
- 6) параллельная отправка запроса к репозиторию для обновления программных структур;
- 7) сохранение результатов анализа в журнал событий.

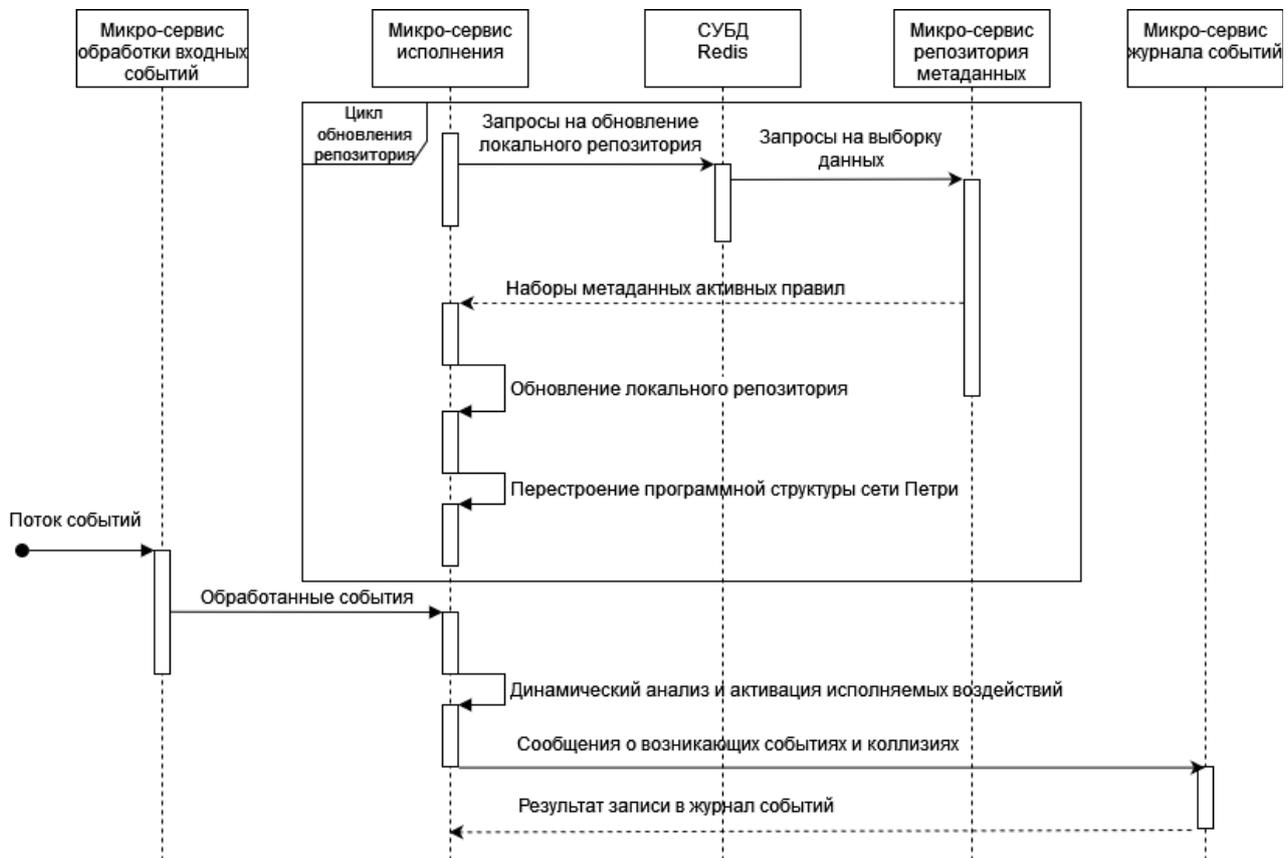


Рис. 2. Сценарий работы микросервиса исполнения активных правил

Для поддержки программных структур данных, используемых при анализе, в актуальном для сервиса состоянии происходит загрузка метаданных из центрального репозитория. Для того, чтобы не возникало задержек при обработке потоков событий, непосредственное обновление происходит в момент отсутствия новых событий в очереди.

В настоящее время разработан начальный прототип программных средств микросервиса исполнения активных правил, выполняется его доработка и совершенствование. При этом продолжаются теоретические исследования и разработки единой методики, моделей и алгоритмов, ориентированных на повышение эффективности и надежности исполнения активных правил, анализа, выявления устранения возможных коллизий.

Список литературы

1. Шибанов С. В., Шлепнев Я. С. Сервис потоковой обработки событий и исполнения активных правил // Математическое моделирование и суперкомпьютерные технологии: тр. XXI Междунар. конф., Нижний Новгород, 22–26 ноября 2021 года. Н. Новгород : Национальный исследовательский Нижегородский государственный университет им. Н. И. Лобачевского, 2021. С. 403–407.
2. Pattern: Database per service // Microservice Architecture [Электронный ресурс]. URL: <https://microservices.io/patterns/data/database-per-service.html> (дата обращения: 05.03.2022).
3. Стопфорд Б. Проектирование событийно-ориентированных систем: Концепции и шаблоны проектирования сервисов потоковой обработки данных с использованием Apache Kafka : пер. с англ. 2-е изд., испр. Иркутск : ITSumma Press, 2019. 175 с.
4. Macehiter N., Ward-Dutton N. The business value of event processing: preparing the path. 2009. April. 24 p.
5. Carlson J. L. Redis in Action. NY : MANNING Shelter Island, 2013. 322 p.
6. Macia H., Valero V., Diaz G. [et. al.]. Complex Event Processing Modeling by Prioritized Colored Petri Nets // IEEE Access. 2016. Vol. 4. P. 7425–7439. doi: 10.1109/ACCESS.2016.2621718



7. Шлепнев Я. С., Шибанов С. В. Стратегии решения проблем исполнения активных правил в системах потоковой обработки событий // Информационные технологии в науке и образовании. Проблемы и перспективы : сб. ст. по материалам IX Всерос. науч.-практ. конф., Пенза, 16 марта 2022 года. Пенза : Изд-во ПГУ, 2022. С. 99–103.
8. Jensen K., Kristensen L. M. Coloured Petri Nets: Modelling and Validation of Concurrent Systems. Cham : Springer, 2009.

References

1. Shibanov S.V., Shlepnev Ya.S. Service for streaming event processing and execution of active rules. *Matematicheskoe modelirovanie i superkomp'yuternye tekhnologii: tr. XXI Mezhdunar. konf. (Nizhniy Novgorod, 22–26 noyabrya 2021 g.) = Mathematical Modeling and Supercomputer Technologies: works of XXI International Conference, Nizhny Novgorod, November 22–26, 2021*. Nizhniy Novgorod: Natsional'nyy issledovatel'skiy Nizhegorodskiy gosudarstvennyy universitet im. N.I. Lobachevskogo, 2021:403–407. (In Russ.)
2. Pattern: Database per service. *Microservice Architecture*. Available at: <https://microservices.io/patterns/data/database-per-service.html> (accessed 05.03.2022).
3. Stopford B. *Proektirovanie sobytiyno-orientirovannykh sistem: Kontseptsii i shablony proektirovaniya servisov potokovoy obrabotki dannykh s ispol'zovaniem Apache Kafka: per. s angl. 2-e izd., ispr. = Designing event-oriented systems: Concepts and design patterns of streaming data processing services using Apache Kafka: transl. from English; 2nd ed., corrected*. Irkutsk: ITSumma Press, 2019:175. (In Russ.)
4. Macehiter N., Ward-Dutton N. *The business value of event processing: preparing the path*. 2009;April:24.
5. Carlson J.L. *Redis in Action*. New York: MANNING Shelter Island, 2013:322.
6. Macia H., Valero V., Diaz G. et. al. Complex Event Processing Modeling by Prioritized Colored Petri Nets. *IEEE Access*. 2016;4:7425–7439. doi: 10.1109/ACCESS.2016.2621718
7. Shlepnev Ya.S., Shibanov S.V. Strategies for solving problems of execution of active rules in event streaming systems. *Informatsionnye tekhnologii v nauke i obrazovanii. Problemy i perspektivy: sb. st. po materialam IX Vseros. nauch.-prakt. konf. (Penza, 16 marta 2022 g.) = Informative technologies in science and education. Problems and prospects: art. collection based on the materials of the IX All-Russian scientific - practical conference, Penza, March 16, 2022*. Penza: Izd-vo PGU, 2022:99–103. (In Russ.)
8. Jensen K., Kristensen L.M. *Coloured Petri Nets: Modelling and Validation of Concurrent Systems*. Cham: Springer, 2009.

Поступила в редакцию / Received 09.04.2023

Поступила после рецензирования и доработки / Revised 07.05.2023

Принята к публикации / Accepted 26.05.2023