



ООО «ПУЛЬС АЙТИМ»
141402, Московская область, г. Химки
ул. Ленинградская, 29, эт. 2, каб. 159
pulsiteam@pulsiteam.ru

ОГРН: 1225000031068
ИНН: 5047262307
КПП: 504701001
+7 499 648-03-04 pulsiteam.ru

ОПИСАНИЕ СИСТЕМЫ АЙФАЙЛ

Оглавление

Введение	3
Alarm service.....	5
Download service	5
Converter service	7
Notification service.....	8
Аналитика.....	8

Введение

Система «АйФайл» состоит из четырех функциональных компонентов (микросервисов):

- Alarm service;
- Download service;
- Converter service;
- Notification service.

Описание и назначение каждого из компонентов указано в таблице ниже:

Таблица 1 – описание и назначение компонентов «АйФайл»

Компонент	Подробно
Alarm service	Java-приложение. Своевременно инициирует начало работы системы.
Download service	Java-приложение. Отвечает за скачивание и размещение файла в целевом хранилище формата S3.
Converter service	Java-приложение. Отвечает за конвертацию скачанного файла из исходного формата в целевой.
Notification service	Java-приложение. Отвечает за уведомление системы-потребителя о наличии новых файлов.

Для полноценного функционирования системы «АйФайл» применяется ряд вспомогательных и общепринятых решений. Описание вспомогательных решений указано в таблицах ниже.

Таблица 2 – описание и примеры вспомогательных решений

Направление решения:	Пояснение
Управление сообщениями	Решение для обмена сообщениями между Java-приложениями «АйФайл», а также для фиксации служебных сообщений с данными по работе «АйФайл» (логов). Пример решения: Apache Kafka.
Генерация имен файлов	Решение для генерации обезличенных имен для загружаемых файлов. Пример решения: Redis.
Дедубликация файлов	Решение для проверки содержимого загружаемого файла. Пример решения: Redis.
Хранение служебной информации	Решение для долговременного хранения служебной информации (логов) по работе «АйФайл». Пример решения: ClickHouse.
Хранилище файлов	Решение для целевого хранения загружаемых файлов. Формат хранилища – S3. Система-потребитель (клиент) скачивают файлы из целевого хранилища. Пример решения: Yandex Object Store.

Логика работы «АйФайл» описана ниже в разрезе функциональных компонентов (Java-приложений) и их взаимодействия со вспомогательными решениями и решениями пользователя (клиента) «АйФайл».

Alarm service

Ниже на схеме изображена работа Java-приложения Alarm service:

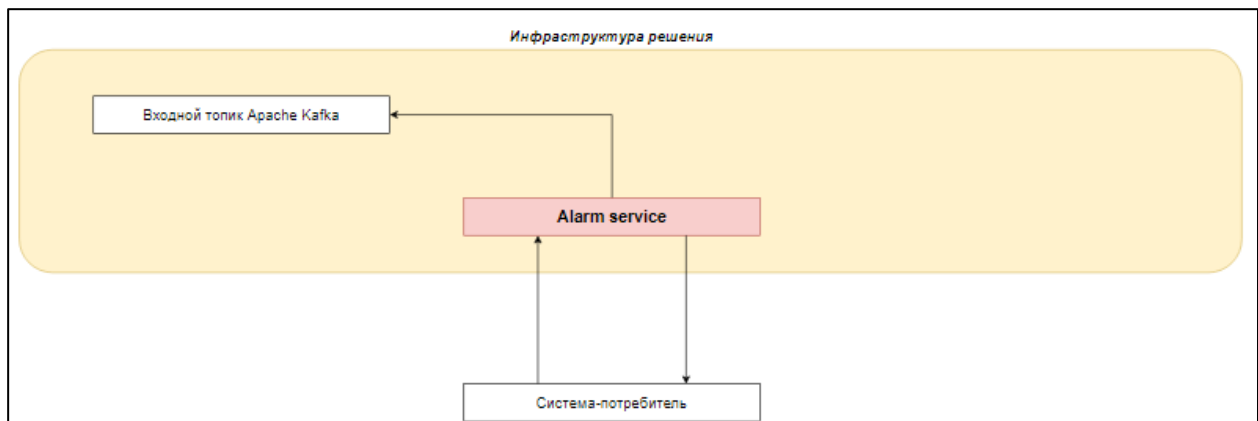


Рисунок 1 – Схема работы Java-приложения Alarm service.

Java-приложение Alarm service выступает в роли инструмента для оркестровки процессов, своевременно инициируя работу системы в целом. Alarm service запрашивает у системы-потребителя данные по источникам файлов для дальнейшего опроса и передает эту информацию на входной топик Apache Kafka Java-приложения Download service.

Download service

Ниже на схеме изображена работа Java-приложения Download service:

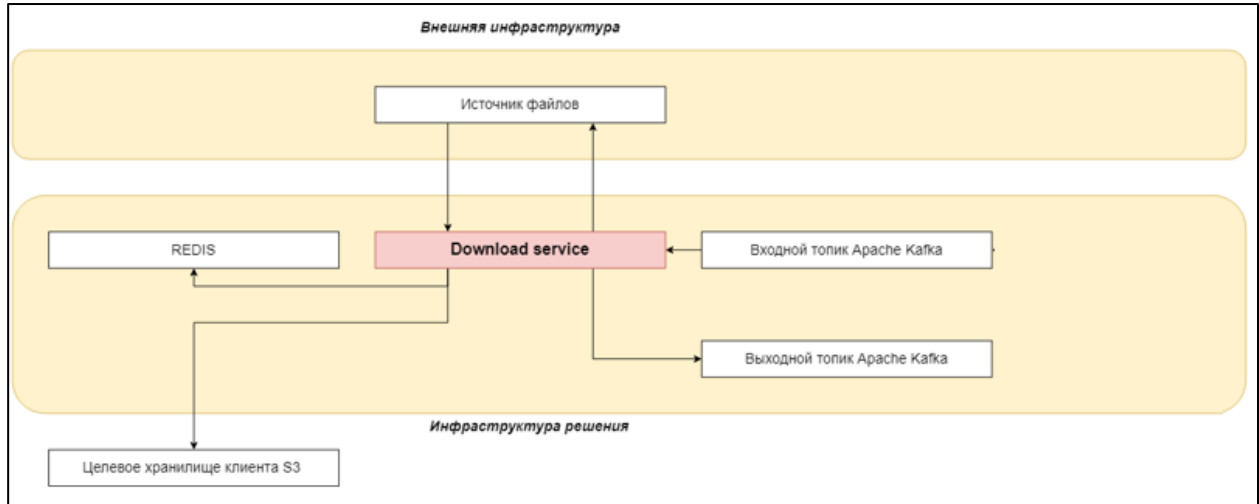


Рисунок 1 – Схема работы Java-приложения Download service.

Download сервис опрашивает источник и загружает необходимый файл в целевое хранилище формата S3. Для механизма дедупликации и генерации имен файлам используется такое решение, как Redis.

После загрузки файла в целевое хранилище Download service отправляет сообщение в выходной топик ApacheKafka, который одновременно является входным для Java-приложения Converter service.

Converter service

Ниже на схеме изображена работа Java-приложения Converter service:

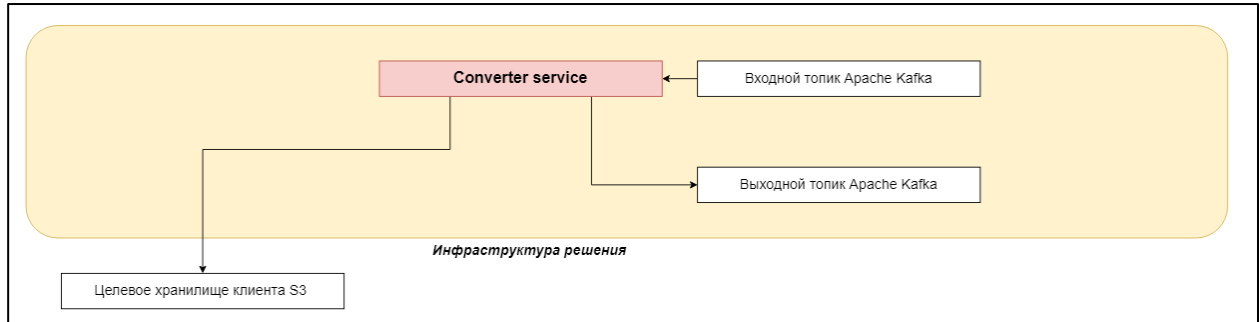


Рисунок 2 – Схема работы Java-приложения Converter service.

Converter service из входного топика получает информацию от Download service о загрузке необходимого файла в целевое хранилище. Converter service выполняет попытку конвертации необходимого файла из исходного формата в целевой формат JSON. При успешной конвертации JSON-файл сохраняется обратно в целевое хранилище.

После выполнения работы Converter service направляет в выходной топик Apache Kafka сообщение с результатами проведенной работы. Выходной топик Converter service одновременно является входным для Java-приложения Notification service.

Notification service

Ниже на схеме изображена работа Java-приложения Notification service:

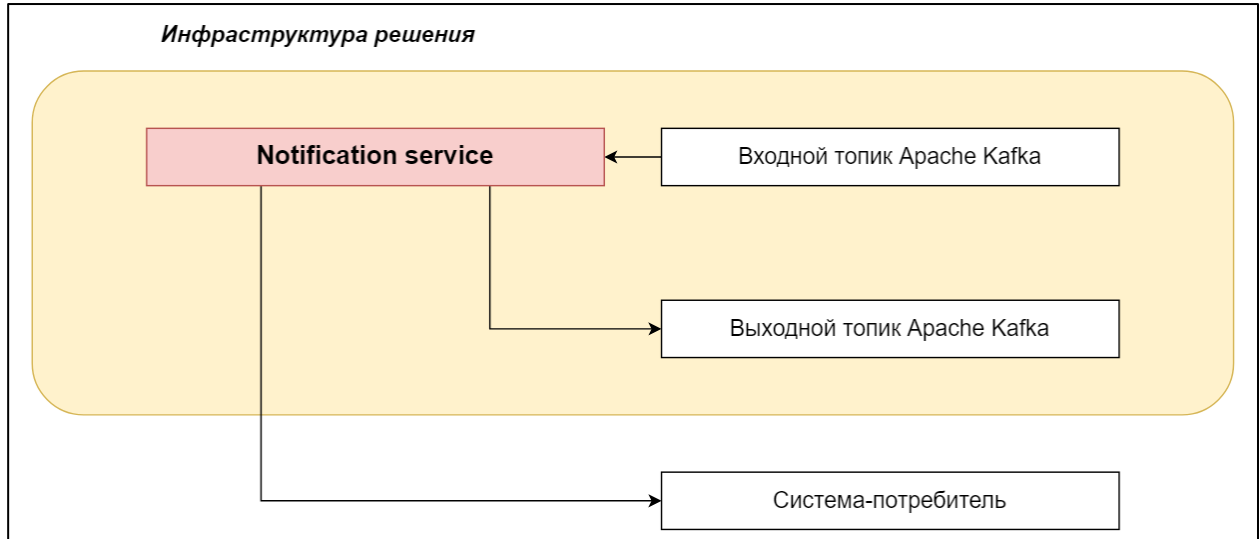


Рисунок 2 – Схема работы Java-приложения Notification service.

Notification service из входного топика получает информацию от Converter service и пытается проинформировать систему-потребителя о наличии готового файла для загрузки.

При определенном количестве неуспешных попыток (настраиваемый параметр) Notification service помещает сообщение в очередь своего входного топика Apache Kafka, чтобы запустить новый цикл оповещения. Система-потребитель должна гарантированно получить информацию о наличии необходимого для загрузки файла в целевом хранилище.

После успешной попытки Notification service отправляет сообщение о факте уведомления системы-потребителя в свой выходной топик

Аналитика

Информация о работе «АйФайл» размещается в аналитическую СУБД ClickHouse для дальнейшей аналитики и мониторинга показателей.