

УДК 004.05

МОДИФИКАЦИЯ ИНФРАСТРУКТУРЫ СБОРА ДАННЫХ ДЛЯ ОБЕСПЕЧЕНИЯ УНИВЕРСАЛЬНОГО МЕХАНИЗМА ИНТЕГРАЦИИ НОВЫХ ИСТОЧНИКОВ

**Ефимов Т.А. (федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»)
Научный руководитель – кандидат педагогических наук, доцент А. В. Маятин
(федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»)**

Модифицируемая система берет данные из сторонних источников и предоставляет конечным пользователям ряд возможностей по обработке и управлению этими данными. В эту систему необходимо внедрить новый источник данных, взамен существующего.

Введение.

В качестве данных выступает информация об однотипных объектах, которые характеризуются числовыми параметрами. Каждый параметр, для модифицируемой системы, представляет собой случайную величину, которая может изменяться с течением времени. Данные поступают периодически, при наличии изменения в них. Причем возможны пики в интенсивности получаемых данных.

В данный момент в системе используется один источник данных. В рамках данной задачи необходимо интегрировать новый источник данных, который является новой, сильно переработанной, версией используемого в данный момент.

В связи с неудовлетворяющей некоторым требованиям архитектурой существующих компонент, отвечающих за импорт и предоставление данных, и сильным изменением API существующего источника данных (по сравнению с новой версией), было решено отказаться от данных компонент в пользу разработки новых.

Постановленной задачей является: разработать компоненты для импорта данных из нового источника в систему, а также компонента, предоставляющего релевантные для клиентских устройств данные. Причем:

- последний компонент не должен быть затронут при переходе на другой источник полностью аналогичных данных, когда это будет необходимо в следующий раз;
- скорость предоставления данных клиентским устройствам не должна зависеть от процесса импорта данных;
- после перезапуска компонента, предоставляющего данные клиентским устройствам, данные должны продолжить предоставляться без дополнительных затрат на ожидание по восстановлению данных в компоненте;
- разрабатываемый компонент, импортирующий данные, должен справляться с незначительными пиками в объеме получаемых данных.

Основная часть.

Было решено подготовленные импортирующим компонентом данные хранить в Redis. А компонент, предоставляющий данные клиентским устройствам, будет брать данные из него. И таким образом, при перезапуске компонента, он сразу же готов предоставлять данные - без

затрат времени на восстановление данных. При этом время на получение данных клиентскими устройствами никак не зависит от работы импортирующего компонента.

Оповещение компонента о наличии обновлений в Redis осуществляется с помощью Kafka. В соответствии с сообщением от Kafka, компонент берет обновленные данные из Redis и присылает их клиентским устройствам с помощью SockJS.

Описанное выше решение распределяет обязанности между элементами системы позволяет при замене импортирующего компонента, или добавления нового с аналогичными данными, не изменять работу этих элементов. Он будет работать с Kafka и Redis.

Импортирующий компонент было решено реализовать таким образом, чтобы перед обработкой сообщений от источника данных, они складывались в очередь, что позволит бороться с пиками в получаемых данных.

Выводы. Была спроектирована системная архитектура решения, а также программная архитектура разрабатываемых компонент.

Ефимов Т. А. (автор)

Подпись

Маятин А. В. (научный руководитель)

Подпись