

РАЗВЕРТЫВАНИЕ МИКРОСЕРВИСНОГО ПРИЛОЖЕНИЯ НА ОБЛАЧНОЙ ПЛАТФОРМЕ HEROKU

Рафеенко Е. Д., Кондратьева О. М., Соболева Т. В.

*Белорусский государственный университет, Минск, Беларусь,
e-mail: rafeenko@bsu.by*

Heroku – облачная PaaS (platform as a service)-платформа, поддерживающая ряд языков программирования, в том числе Java [1]. Она позволяет выложить Java приложение на бесплатный хостинг. Heroku имеет бесплатный тарифный план, который позволяет загружать не более 5 приложений на один аккаунт и, если приложение не активно в течение 30 минут, оно переводится в спящее состояние. Список поддерживаемых языков также включает Node.js, Scala, Clojure, Python, Go, Ruby, PHP. В данной работе рассматриваются особенности развертывания приложения с микросервисной архитектурой на платформе Heroku. В Heroku все приложения выполняются в легковесных Linux контейнерах, которые называются dynos. Каждый dynos принадлежит к одному из объявленных типов процессов и выполняет команду запуска, связанную с этим типом процесса. Эта команда описывается в специальном файле (Procfile). В частности, Web dynos принадлежат типу Web- процесс. Они получают весь HTTP трафик от маршрутизаторов. Также существуют рабочие dynos, предназначенные для выполнения длительных фоновых задач.

Развертываемое микросервисное приложение разработано с помощью фреймворка Vert.x. Eclipse. Vert.x – это проект с открытым исходным кодом от Eclipse Foundation, который предоставляет удобную модель для построения микросервисов [2]. Vert.x имеет API для разработки асинхронных сетевых приложений, можно подобрать необходимые модули для приложения: взаимодействия с БД, мониторинга, аутентификации, логгирования, обнаружения сервисов и т.д. Vert.x – платформа полиглот, поддерживает ряд языков, использующих JVM.

Основные понятия, которые использует Vert.x – это verticle (вертикаль) и event bus (шина событий), позволяющая вертикалям взаимодействовать между собой. Вертикаль – единица развертывания в Vert.x, она обрабатывает входящие события, приходящие в event loop, такие как получение входящих сетевых буферов или сообщений от других вертикалей.

Для того чтобы развернуть микросервисное vert.x приложение в облачном сервисе Heroku необходимо выполнить следующие шаги.

1. Зарегистрироваться на сайте heroku.com.
2. В личном кабинете создать новое приложение: New -> Create new app, указать имя приложения и выбрать регион (Europe).
3. Т.к. приложение будет использовать базу данных, то ее нужно создать в Heroku. Бесплатно и без верификации cloud-провайдер поддерживает Postgres базу данных. Heroku также предоставляет возможность бесплатного использования других СУБД, например, MySQL, но для того чтобы их подключить необходимо ввести данные платежной карты для верификации (в том числе и для бесплатного тарифного плана).

Для создания базы данных нужно выбрать вкладку Resources созданного приложения, нажать кнопку Find more add ons и найти нужное дополнение. После создания базы данных имя пользователя, пароль и имя базы данных можно найти на вкладке Resources.

4. Heroku при развертывании приложения использует Maven, поэтому микросервисное vert.x приложение лучше создавать как Maven проект, добавив в pom.xml файл все необходимые зависимости.

5. Heroku при развертывании приложения использует Procfile. Его необходимо добавить в корень приложения (на одном уровне с pom.xml). Для vert.x микросервиса Procfile выглядит следующим образом:

```
web: java $JAVA_OPTS -Dhttp.port=$PORT -jar target/*-fat.jar
```

6. Heroku поддерживает развертывание приложений через GitHub. Поэтому развертываемое приложение необходимо выложить на GitHub, а на вкладке Deploy в личном кабинете Heroku подключиться к GitHub.

После успешного развертывания приложение будет доступно по url:

```
http://your-app-name.herokuapp.com/
```

Для удобства работы рекомендуется установить на свой компьютер консоль Heroku. В частности, с помощью CLI можно просматривать поток вывода логгера приложения: `heroku logs -a your-app-name -t`

Таким образом, в работе рассмотрена общая информация о разработке микросервисного приложения с помощью фреймворка Vert.x и его развертывании на облачной платформе Heroku. Поскольку приложение состоит из нескольких микросервисов, то представляет интерес организация их взаимодействия между собой. Полноценный поиск сервисов (Service Discovery) в Heroku можно организовать в Heroku Private Space. Private Space – это выделенный контекст для работы dynos - процессов и определенных дополнений (add ons) в изолированной сети. Но доступна такая возможность только на платной платформе Heroku Enterprise. В рамках стандартных возможностей vert.x микросервисы могут отсылать друг другу запросы, используя RESTful API. Также удобным средством взаимодействия может быть брокер сообщений. В частности, в Heroku доступно дополнение RabbitMQ – это программный брокер сообщений на основе стандарта AMQP. После подключения данного дополнения к каждому микросервису, становится возможен обмен сообщениями между ними.

Литература

1. Heroku DevCenter [Electronic resource]. – Mode of access: <https://devcenter.heroku.com>. – Date of access: 21.01.2021.

2. Vert.x documentation [Electronic resource]. – Mode of access: <http://vertx.io/docs/>. – Date of access: 14.03.2021.