



ТЕХНОЛОГИЯ МИКРОСЕРВИСНОЙ АРХИТЕКТУРЫ ИНФОРМАЦИОННЫХ СИСТЕМ ДЛЯ ИСПОЛЬЗОВАНИЯ В ОБЛАЧНОЙ СРЕДЕ

Алексеев Д. И.

Харьковский национальный университет радиоэлектроники

На текущий момент наблюдается устойчивый рост использования облачных технологий на предприятиях и организациях. При этом облачные технологии являются средством, которое можно применить для улучшения бизнес процессов на предприятиях и в организациях. Таким образом, можно сказать, что создание облачных информационных систем является актуальной задачей.

Для решения задачи создания облачных ИС возможно использовать сервис-ориентированный подход. При таком подходе информационная система представляется как набор сервисов. Комбинация различных сервисов предоставляет функционал полноценной информационной системы [1]. Развитие виртуализации ресурсов положило начало новой архитектуре построения информационных систем. Прежде всего, это касается обеспечения независимости от имеющихся физических ресурсов и обеспечения динамической реконфигурации текущих функциональных возможностей информационной системы. Динамическая реконфигурация необходима для адаптации свойств системы к изменяемым условиям эксплуатации и растущим потребностям пользователей.

Предлагается технология проектирования облачных информационных систем, основанная на облачных решениях и положениях сервис-ориентированных структур. Эта технология заключается в том, что на этапе проектирования, согласно подходу SOA, информационная система представляется как набор функциональных сервисов с микросервисной архитектурой. Каждый функциональный сервис реализуется в виде отдельного модуля. Модуль представляет собой самодостаточную реализацию функциональной задачи, а также имеет возможность принимать и передавать данные во внешнюю относительно себя среду [2].

Для реализации такой технологии предлагается создать структуру информационной системы, которая представляет собой каркас, состоящий из контейнеров. Каждый контейнер содержит микросервисы, реализующие отдельные функциональные задачи в рамках информационной системы.

Чтобы построить описанную структуру необходимо решить несколько задач: выбор среды для разворачивания контейнеров, выбор микросервисов и формирование связей между сервисами и контейнерами [3].

Таким образом, предлагается технология микросервисной архитектуры информационных систем, состоящая из трех этапов.

Первый этап заключается в выполнении работ по декомпозиции бизнес-задач информационной системы на микросервисы, выполняющие отдельные функции.



Секция 1. Современные информационные системы и технологии: проблемы, методы, модели. Управление проектами и программами.

Второй этап технологии заключается в реализации полученных микросервисов в виде контейнеров, содержащих программные модули функций информационной системы.

Управление контейнерами осуществляется с помощью средства управления контейнерами Docker. Для работы микросервисов предлагается использовать контейнеры с установленными в них средами выполнения.

При этом в облачной ИС предполагается использование множества контейнеров, соответствующих функциям информационной системы. Как следствие, необходимо решать задачи взаимодействия всех контейнеров между собой, масштабировать набор контейнеров, запускать их и выполнять балансировку нагрузки между ними.

Для решения поставленных задач предлагается использовать инструмент Kubernetes, в котором реализованы все функции, необходимые для запуска приложений на основе Docker в конфигурации с высокой доступностью (кластеры более 1000 узлов, с multi-availability и multi-region зонами): управление кластером, планирование, обнаружение сервисов, мониторинг, управление учетными данными и многое другое.

Процесс разворачивания среды выполнения в каждом контейнере предлагается реализовать с помощью системы управления конфигурациями Chef. Использование такой системы позволит создать шаблоны настроек контейнеров и автоматизировать процесс разворачивания контейнеров.

На третьем этапе необходимо реализовать процесс взаимодействия между микросервисами. Для организации такого взаимодействия предлагается использовать платформу RabbitMQ, которая реализует систему обмена сообщениями между компонентами программной системы на основе протокола AMQP (Advanced Message Queuing Protocol), т.е. протокола для передачи сообщений между компонентами системы. Отдельные подсистемы (или независимые приложения) могут обмениваться произвольным образом сообщениями через AMQP-брокер, который осуществляет маршрутизацию, возможно гарантирует доставку, распределение потоков данных, подписку на нужные типы сообщений.

Предложенная технология позволит создать структуру облачной ИС, что упростит процесс ее реконфигурации и последующего сопровождения за счет того, что модули ИС между собой независимы и отказ или изменение одного из них не влияет на остальные. Кроме того, размещение ИС в облаке увеличит ее производительность и скорость работы.

1. SOA Blueprint: A Toolbox for Architects by Jürgen Kress, Berthold Maier, Hajo Normann, Danilo Schmeidel, Guido Schmutz, Bernd Trops, Clemens Utschig-Utschig, Torsten Winterberg [Электронный ресурс] Режим доступа: <http://www.oracle.com/technetwork/articles/soa/ind-soa-1-blueprint-1934607.html>. 2. S. Newman, Building Microservices – Designing Fine-Grained Systems, O’Reilly, 2015 3. Balalaie, A. (2016-05-01). «Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture». IEEE Software 33 (3): 42–52.