

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 004.65:[004.771+004.383.2]

Пац
Егор Александрович

**СИСТЕМА РАСПОЗНАВАНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ НА
ОСНОВЕ ТЕХНОЛОГИЙ ВИРТУАЛИЗАЦИИ И ОБЛАЧНЫХ
ВЫЧИСЛЕНИЙ**

АВТОРЕФЕРАТ

на соискание степени магистра информатики и вычислительной техники

по специальности 1- 40 81 02 «Технологии виртуализации и облачных
вычислений»

Научный руководитель
Насуро Екатерина Валериевна
кандидат технических наук,
доцент кафедры ЭВМ, БГУИР

Минск 2020

ВВЕДЕНИЕ

В современном мире наблюдается очень быстрый рост технологий. На сегодняшний день одним из перспективных направлений информационных технологий являются облачные вычисления и предоставляемые ими современные сервисы создания, хранения, обработки и поиска данных.

Современные облачные технологии способны существенно сократить расходы, и сегодня многие компании все чаще переносят свои корпоративные системы и бизнес-приложения в облако. Данные системы позволяют в значительной мере снизить нагрузку на инфраструктуру, оптимизировать утилизацию вычислительных ресурсов, повысить доступность критически важных данных и сервисов, улучшить информационную безопасность компонентов и систем. Перечисленные преимущества делают очевидным выбор облачных технологий большинством компаний.

Одним из наиболее удачных примеров использования облачных технологий является использование микросервисной архитектуры программного обеспечения. Данная архитектура подразумевает использование отдельных сервисов для решения определенных задач. Таким образом убирается единая точка отказа приложения, что увеличивает доступность сервиса даже при отсутствии части функционала.

Актуальность исследования обусловлена необходимостью оптимизации разработки инновационных облачных систем. Разработанная система предоставляет ряд услуг для конечного пользователя. Мониторинг объектов не тривиальная задача, которая, на данный момент, не имеет общепринятых решений. Часть детектирования объектов в видеопотоке передать автономным системам. Основой данных систем выступает искусственный интеллект. При расширении количества обрабатываемых потоков данных необходимо позаботиться о распределении нагрузки, масштабируемости и отказоустойчивости всех ключевых компонентов сервиса, данные задачи позволяет решить разработанная облачная система. Зачастую, при развертывании программного обеспечения, отсутствует возможность запуска каждого экземпляра на отдельном, изолированном сервере. Одной из особенностей сложных систем выступает утилизация ресурсов. На данный момент, разработка подобных систем очень актуальна. Большинство компаний, которые предоставляют программное обеспечение как сервис, хотят получить максимальную доступность и минимальное время отклика предоставляемого сервиса. Таким образом, система, разрабатываемая в рамках диссертации, находится на пике популярности, так как отвечает всем основным принципам высоконагруженных, отказоустойчивых, масштабируемых систем.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Магистерская диссертация представлена следующими материалами. Пояснительная записка 103 страниц, 32 рисунка, 3 таблицы, 2 формулы, 9 блоков кода, 31 литературный источник, 1 приложение.

Цель и задачи исследования

Объект исследования: Система обработки данных, как сложная система, использующая методы и концепции облачных вычислений.

Предмет исследования: Принципы построения высоконагруженных, отказоустойчивых, масштабируемых, распределенных систем, поддерживающих микросервисную архитектуру.

Цель исследования: Разработка архитектуры и интеграционных модулей для развертывания системы на облачных платформах.

Задачи исследования:

1. Обзор принципов построения облачных систем.
2. Сравнение методов обработки видеопотока.
3. Анализ принципов разработки программного обеспечения на основе микросервисной архитектуры.
4. Поиск и анализ способов виртуализации для оптимизации утилизации ресурсов инфраструктуры.
5. Анализ особенностей построения высоконагруженных, отказоустойчивых, масштабируемых систем.
6. Разработка архитектуры инфраструктурной части системы.
7. Разработка архитектуры микросервисной части системы.
8. Разработка модулей непрерывной интеграции и непрерывной доставки программного обеспечения для развертывания и установки системы на различные облачные платформы.

Научная и практическая значимость исследования: создание и внедрение разработанного продукта в рамках требований Республиканской системы мониторинга общественной безопасности.

Произведено сравнение методов обработки видеопотока. По результатам сравнения был выбран наиболее актуальный метод по требуемым характеристикам. После изучения всех актуальных возможностей было принято решение использовать для дальнейшей разработки библиотеку компьютерного зрения OpenCV.

Были изучены алгоритмы поиска объектов на изображениях. По выбранным для сравнения параметрам, было принято решение применять в разрабатываемой системе алгоритм поиска объектов Faster R-CNN. Так как данный алгоритм хорошо оптимизирован, то появляется возможность применения данного алгоритма в высоконагруженных системах обработки данных в режиме реального времени.

Для ускорения и оптимизации вычислений было принято решения использовать библиотеку глубинного обучения TensorFlow. TensorFlow – это библиотека для обработки чисел в виде графов потоков данных с сохранением состояния.

Выполнен анализ принципов разработки программного обеспечения на основе микросервисной архитектуры. Микросервисная архитектура – вариант сервис-ориентированной архитектуры программного обеспечения, ориентированный на взаимодействие несколько это возможно небольших, слабо связанных и легко изменяемых модулей – микросервисов, получивший распространение в связи с развитием практик гибкой разработки и DevOps методологии.

В процессе исследований было определено понятие контейнеризации приложений, выделены основные преимущества запуска программного обеспечения именно в контейнере, произведено сравнение с виртуальной машиной и аппаратным сервером.

Определена необходимость планирования зависимостей и программных библиотек, требующихся для обработки видеопотока внутри контейнера. Рассмотрены основные рекомендации и составлению программного стека контейнера.

В результате поиска способов виртуализации для оптимизации утилизации ресурсов инфраструктуры, было найдено большое количество решений. Анализ показал, что наиболее оптимальным является контейнеризация, так как данный подход снижает прочие нагрузки на вычислительные ресурсы до минимальных значений.

Исходя из понятия облачных вычислений следует понятие виртуальная машина. Виртуальная машина – программная и/или аппаратная система, эмулирующая аппаратное обеспечение некоторой системы и исполняющая программы для target-платформы на host-платформе или виртуализирующая

некоторую платформу и создающая на ней среды, изолирующие друг от друга программы и даже операционные системы.

Так же для конкретного понимания облачных сред дано определение понятию виртуализация. Виртуализация – предоставление набора вычислительных ресурсов или их логического объединения, абстрагированное от аппаратной реализации, и обеспечивающее при этом логическую изоляцию друг от друга вычислительных процессов, выполняемых на одном физическом ресурсе.

Рассмотрены основные тренды построения облачных систем. К наиболее актуальному относится явление контейнеризации. Контейнеризация – метод виртуализации, при котором ядро операционной системы поддерживает несколько изолированных экземпляров пространства пользователя вместо одного. Данное решение уменьшает накладные расходы на инфраструктуру, позволяя оптимизировать вычислительные ресурсы под полезные нагрузки.

Были рассмотрены основные особенности построения высоконагруженных, отказоустойчивых, масштабируемых систем. Использование IaaS по сравнению с PaaS и SaaS, значительно снижает расходы на инфраструктуру, расширяет возможности конфигурирования и использования вычислительных ресурсов. Данная модель предоставления услуг делает инфраструктуру максимально независимой от поставщика облачных технологий. Из недостатков использования IaaS: требуемая высокая квалификация системных инженеров и большая часть ответственности лежит на заказчике инфраструктурного решения.

Исходя из данных, полученных в результате экспериментальных установок платформ оркестрации контейнеров, было принято важное архитектурное решение. Использование Docker Swarm не предоставляет необходимой функциональности, не способна выдерживать большие нагрузки. Mesos система со сложной конфигурацией и администрированием, так как развертывания в данной системе описываются в формате JSON. Решение было сделано в пользу Kubernetes, так как данная платформа является наиболее гибкой, имеет большое количество ресурсов. Платформа Kubernetes готова к высоким нагрузкам, она достаточно отказоустойчива и продумана для исключения ситуаций отказа от обслуживания.

Платформа Kubernetes промышленного уровня обычно включает мониторинг, ведение журнала, трассировку, управление выпусками для развертываний, автоматическое тестирование в конвейере и автоматическое развертывание. Все это затраты на управление, делая Kubernetes дешевым и простым решением для развертывания программного обеспечения в производственной среде.

ЗАКЛЮЧЕНИЕ

Для проведения исследований применялся вычислительный кластер БГУИР на базе Белорусского государственного университета информатики и радиоэлектроники. Интегрированная среда разработки Visual Studio Code и Unix система MacOS, использовались для разработки интеграционных модулей развертывания. В качестве операционной системы виртуальных машин использовался Linux дистрибутив CentOS версии 7.7.1908.

По результатам исследования были получены актуальные и точные знания по основным принципам построения облачных систем. Дано точное определение понятию облачные вычисления. Облачные вычисления – модель обеспечения удобного сетевого доступа по требованию к некоторому общему фонду конфигурируемых вычислительных ресурсов, которые могут быть оперативно предоставлены и освобождены с минимальными эксплуатационными затратами или обращениями к провайдеру.

Исследования доказали преимущества применения облачных вычислений и технологий виртуализации для построения система обработки данных. По результатам исследования спроектированная система имеет высокую доступность, так как все ключевые ресурсы системы реплицируются и управляются оркестратором, что позволяет решать проблему создания кворума. Спроектированная система имеет возможность масштабирования, благодаря правильным архитектурным решениям, возможно расширение системы до 10000 вычислительных узлов, что значительно увеличивает доступные ресурсы центрального процессора, оперативной памяти и дискового пространства. Масштабирование также возможно на уровне микросервисной части системы, что позволяет оптимизировать нагрузку на вычислительные ресурсы и их утилизацию. Данная возможность позволяет динамическое управление инфраструктурой для обработки пиковых нагрузок. Архитектура и интеграционные модули разработаны так образом, что не имеют жесткой привязки к определенному провайдеру облачных технологий. Это позволяет использовать систему в любом центре обработки данных.

Разработана архитектура инфраструктурной части системы (Рисунок 1). Данная архитектура позволяет объединить вычислительные мощности виртуальных машин, запущенных в облачной среде. Таким образом предоставляется возможность обращение к кластеру для выделения необходимых ресурсов для запуска определенного контейнера, либо процесса. Исходя из этого возможно гибкое распределение ресурсов по узлам кластера. Так же решается проблема утилизации ресурсов, позволяя распределить нагрузку.

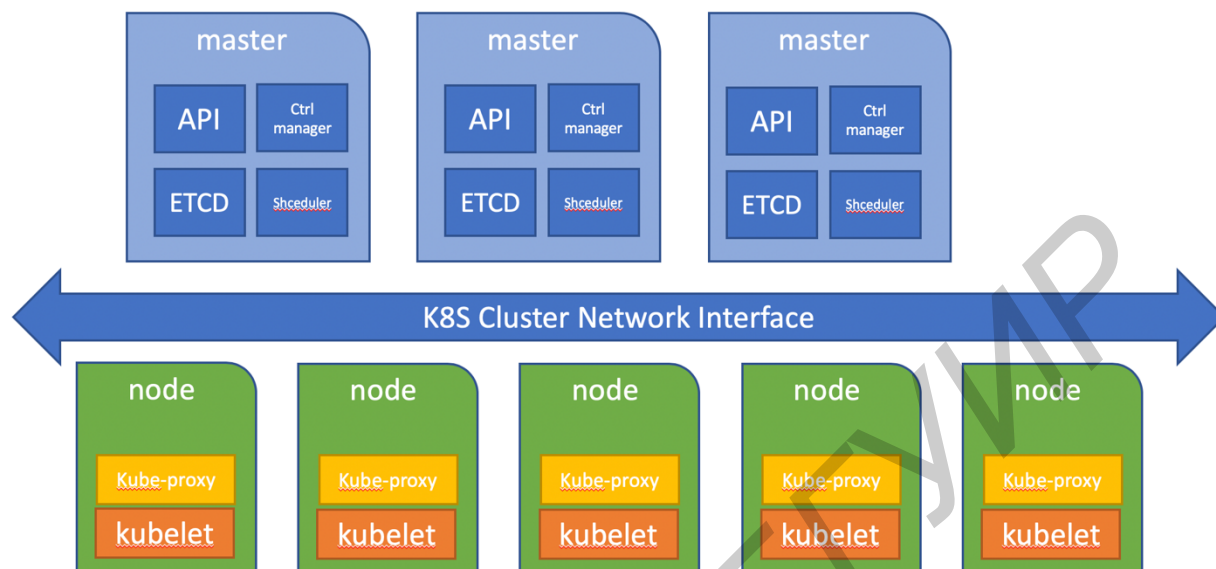


Рисунок 1 – Архитектура инфраструктурной части системы

Разработана архитектуры микросервисной части системы (Рисунок 2). Микросервисы являются слабосвязанными, небольшими и независимыми частями большой сети сервисов, которые составляют приложение. Все эти службы могут быть развернуты и управляться независимо и динамично.

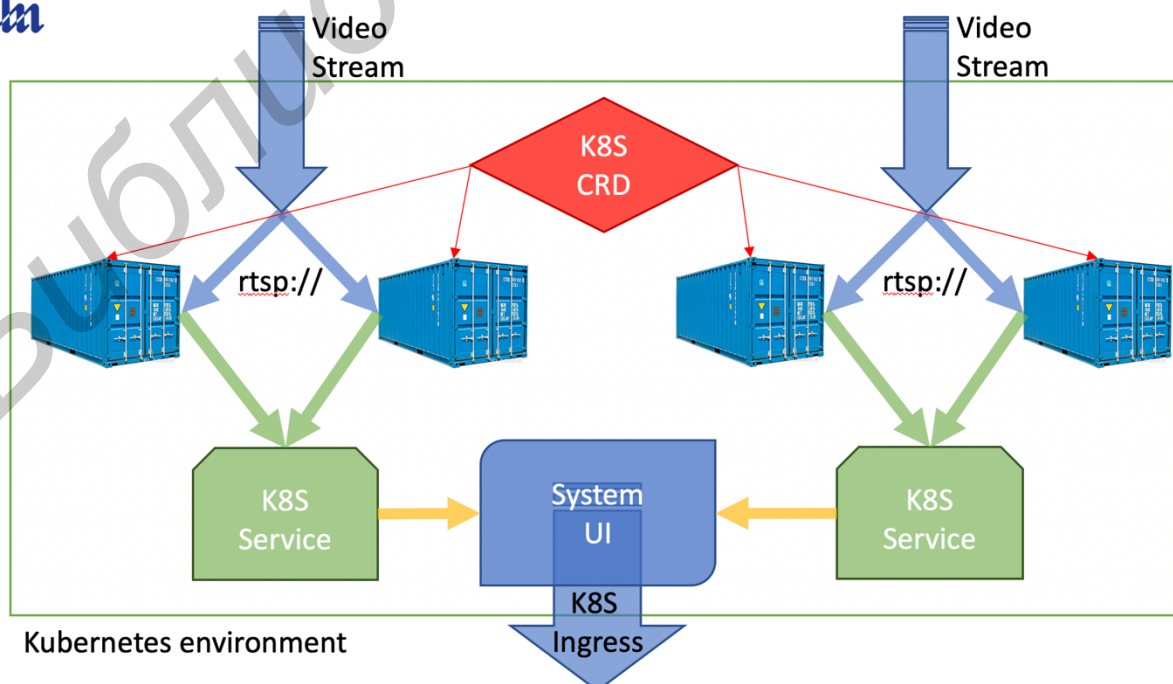


Рисунок 2 – Архитектура микросервисной части системы

На высоком уровне среда Kubernetes состоит из плоскости управления, распределенной системы хранения для поддержания согласованного состояния кластера и нескольких узлов кластера. Плоскость управления – это система, которая ведет учет всех объектов Kubernetes. Узлы кластера – это машины, которые запускают контейнеры и управляются главными узлами. Kubelet является основным и наиболее важным из контроллеров Kubernetes. Он отвечает за управление уровнем выполнения контейнеров.

Произведен научный поиск и выбор среды запуска контейнеров. По результатам сравнения по ряду ключевых параметров, принято важное архитектурное решение в выборе Docker в качестве Container runtime.

Разобраны и применены лучшие практики создания образов docker-контейнеров, по результатам был разработан docker-образ включающий все зависимости и требуемое программное обеспечение необходимое для корректной обработки видеопотока в режиме реального времени.

Разработаны модули непрерывной интеграции и непрерывной доставки программного обеспечения для развертывания и установки системы на различные облачные платформы.

Terraform является одним из ключевых инструментов для эффективного развертывания и обновления инфраструктурных решений. Таким образом, Terraform отлично подходит для реализации DevOps методологий.

Для выбора инструмента управления конфигурациями был проведен анализ самых популярных систем. Исходя из таблицы 3.1 можно сделать вывод, что оптимальным инструментом, для решения поставленной задачи, является Ansible. Данная система управления конфигурациями имеет открытый исходный код, что означает возможность ее бесплатного использования. Ansible имеет большое сообщество, это означает, что большое количество тривиальных задач уже имеют оттестированные решения. Одним из ключевых преимуществ использования Ansible является отсутствие программ-агентов. Для конфигурации узлов достаточно иметь доступ к ним. Написание кода для конфигурации системы не вызывает затруднений, так как формат `yaml` интуитивно понятен.

Диссертация является завершенной. Программным комплексом успешно решаются задачи распознавания и классификации. Система может использоваться как в качестве конечного продукта, так и выступать в роли платформы для других программных средств.

Публикации результатов исследования

«ОБНАРУЖЕНИЕ И РАСПОЗНАВАНИЕ ОБЪЕКТОВ В ВИДЕОПОТОКЕ НА ОСНОВЕ R-CNN ДЕТЕКТОРОВ» Компьютерные системы и сети: материалы 55-й научной конференции аспирантов, магистрантов и студентов (Минск, 22-26 апреля 2019 года). – Минск: БГУИР, 2019. – С. 55-56

«СИСТЕМА ОБНАРУЖЕНИЯ И РАСПОЗНАВАНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ» Международная научно-техническая конференция «Мониторинг техногенных и природных объектов» (Минск, 28-29 ноября 2019). – Минск: БГУИР, 2019.

«АРХИТЕКТУРА РАСПРЕДЕЛЕННОЙ СИСТЕМЫ ОБРАБОТКИ ВИДЕО» Компьютерные системы и сети: материалы 56-й научной конференции аспирантов, магистрантов и студентов (Минск, 18-20 мая 2020 года). – Минск: БГУИР, 2020.

«СИСТЕМА РАСПОЗНАВАНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ НА ОСНОВЕ ТЕХНОЛОГИЙ ВИРТУАЛИЗАЦИИ И ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ» Шестая международная научно-практическая конференция Big Data and Advanced Analytics (Минск, 20-21 мая 2020 года). – Минск: БГУИР, 2020.

Библиотека БГУИР