АНАЛИЗ И РАЗРАБОТКА РАСПРЕДЕЛЁННОЙ АРХИТЕКТУРЫ ЭКСПЛУАТАЦИИ НЕЙРОННЫХ СЕТЕЙ

Дипломная работа

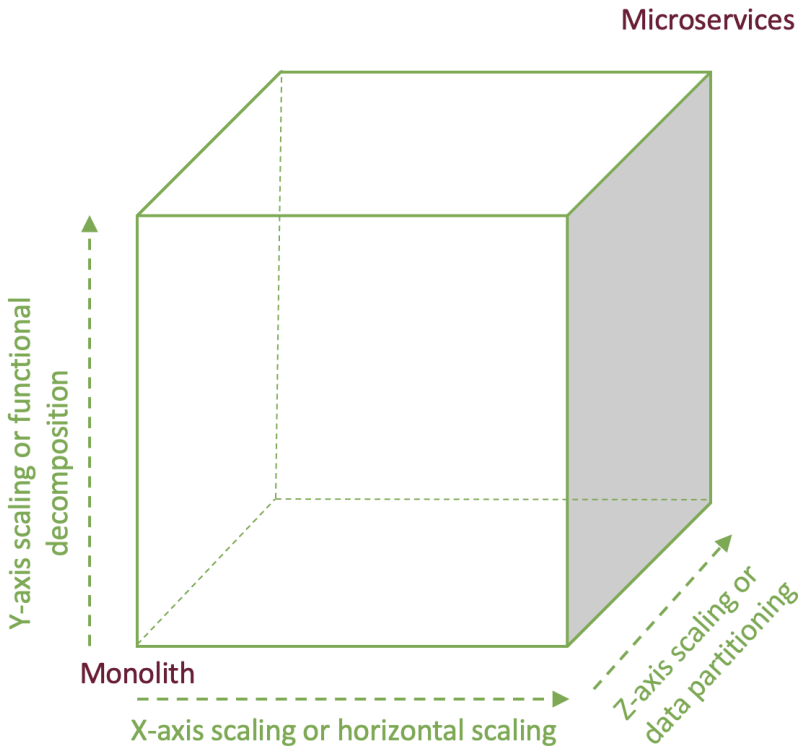
Ларин Егор Сергеевич

Белорусский государственный университет
ФПМИ, КТС, 4 курс
руководитель: старший преподаватель Шолтанюк С. В.

Минск, 2024

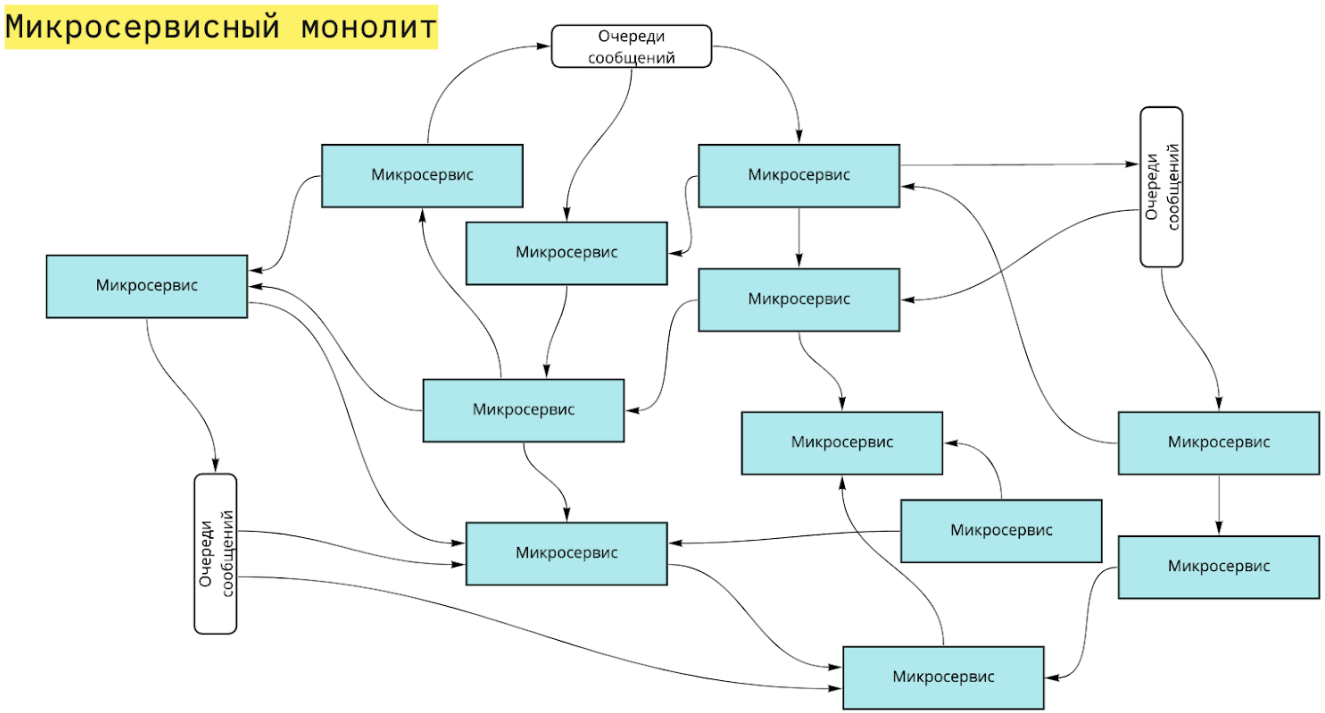
Микросервисы

- В последние годы микросервисная архитектура значительно приобретает популярность в области разработки ПО.
- Использование микросервисной помогает решить вопросы масштабируемости.



Микросервисный монолит

- Внедрение микросервисов порождает другие проблемы, которые необходимо решать с помощью современных инструментов.

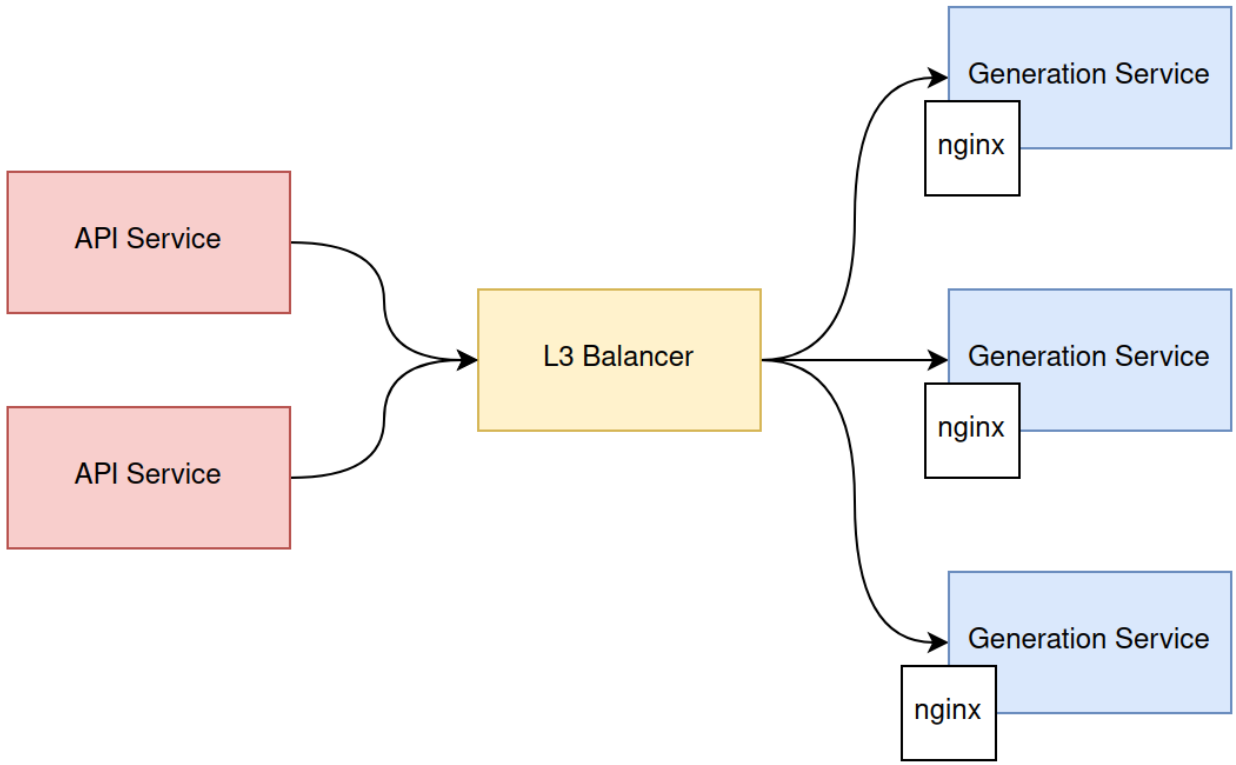


Задача сервиса генерации изображений

1. Обработка входного запроса с параметрами генерации изображения
2. Обработка запросов о статусе генерации изображения
3. Обработка запроса отмены генерации
4. Обработка запроса получения сгенерированного изображения

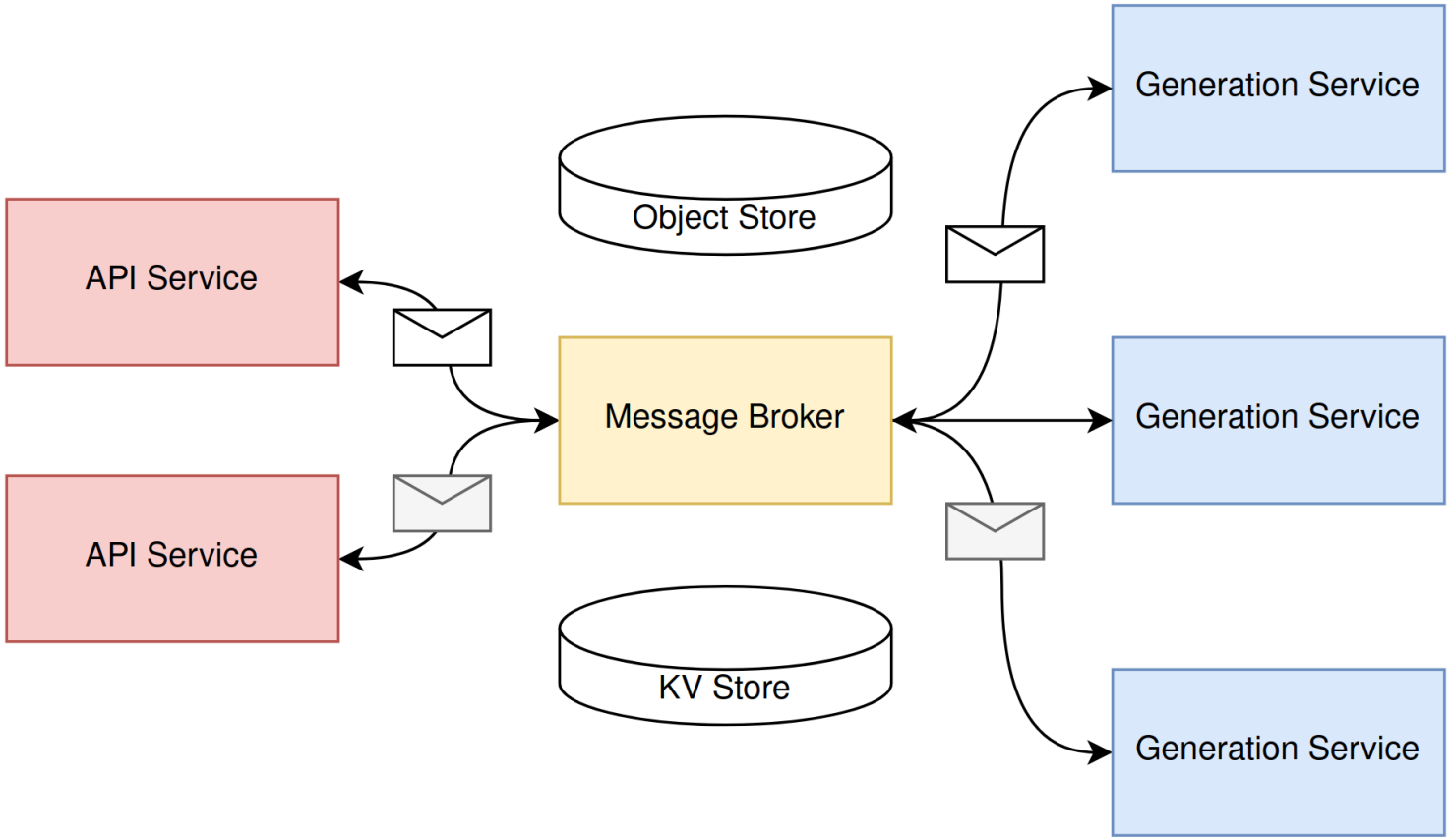
Общий балансер и проксирование трафика

- Легче всего в реализации
- Плохо масштабируется и предоставляет слабые гарантии обработки сообщений



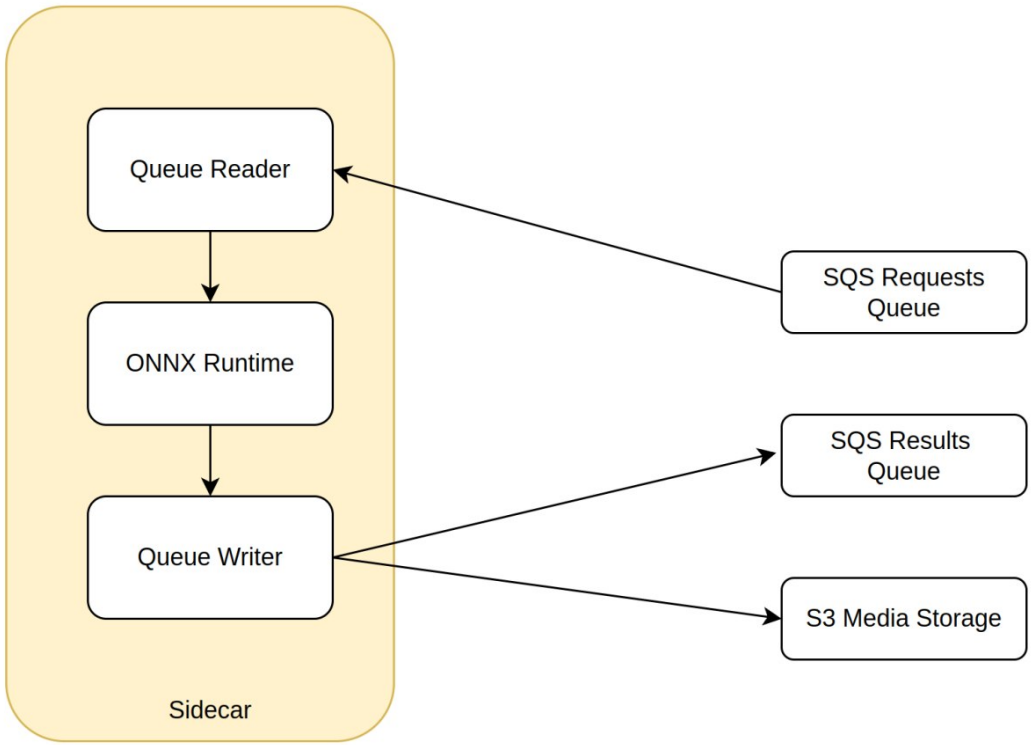
Брокер сообщений и асинхронная обработка

- Устойчива к перегрузкам
- Гарантии обработки и порядка



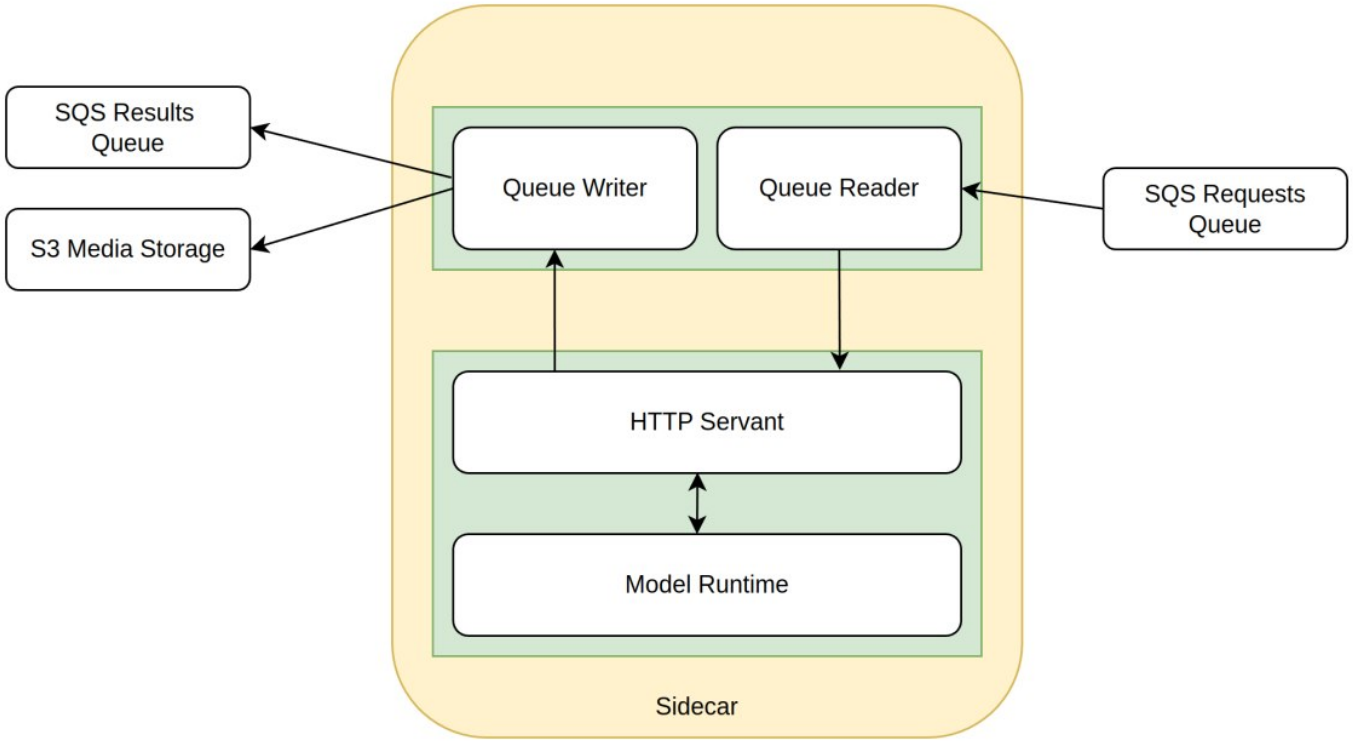
Нативная реализация

- Минимальные тайминги и высокая надежность
- Требует дополнительной разработки для поддержки другой среды исполнения



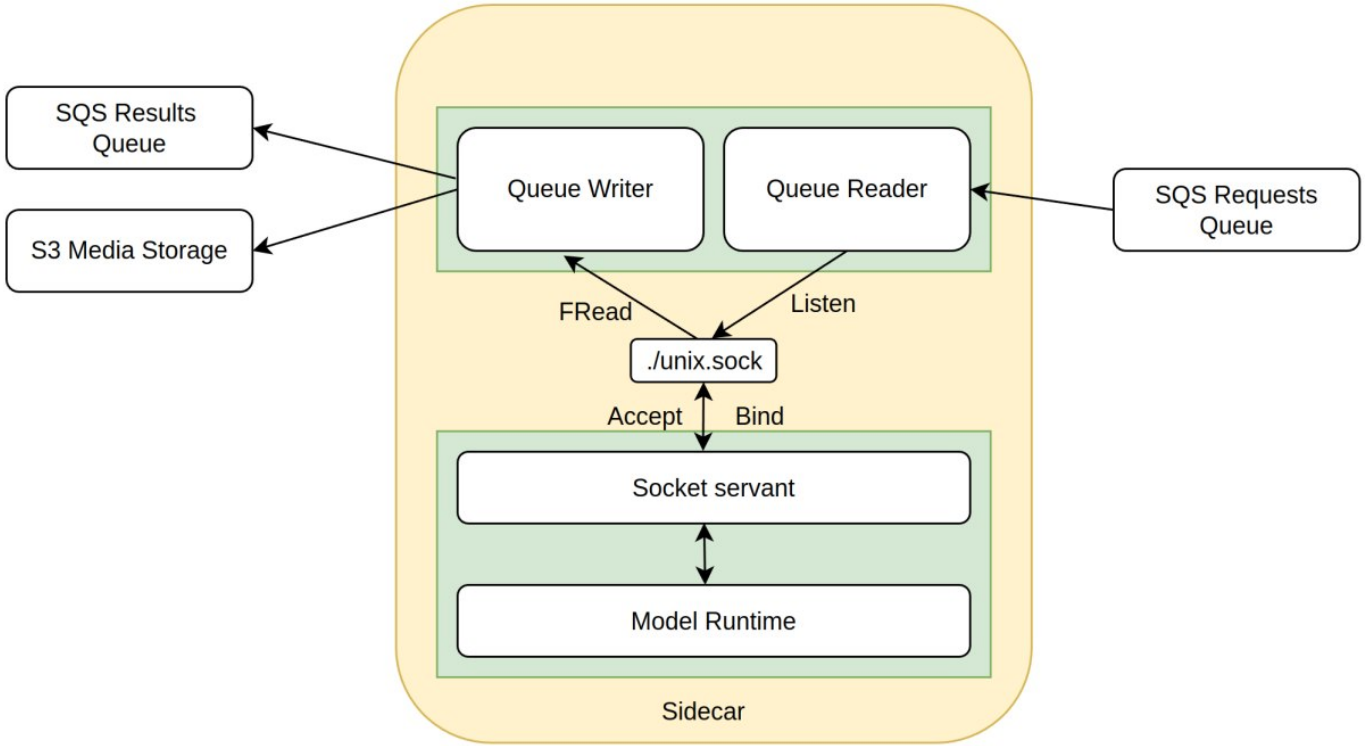
HTTP сайдкар

- Самый простой подход в реализации
- Издержки на передачу данных по сети и сериализацию



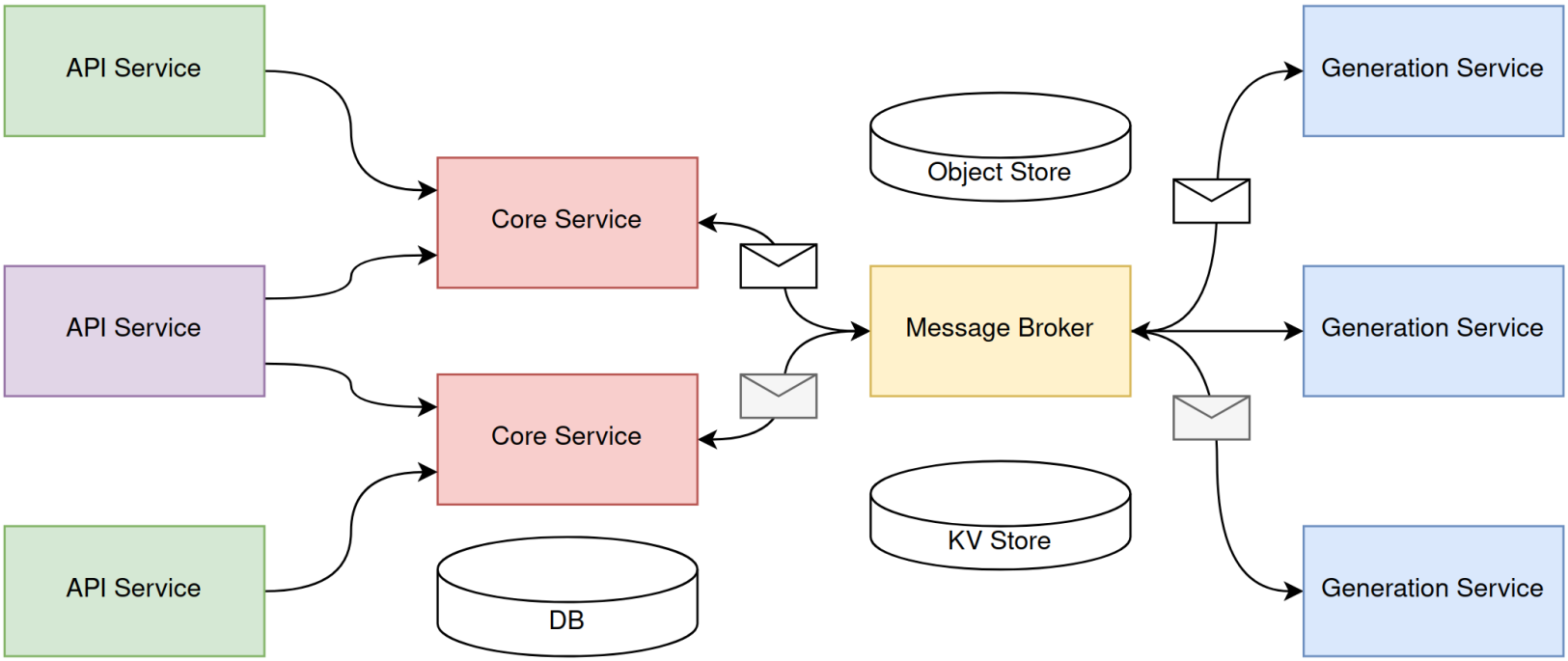
UDS сайдкар

- Более высокая производительность по сравнению с HTTP
- Требует разработки и поддержки сетевого протокола



HTTP гейтвей

- Отдельный релизный цикл
- Инкапсуляция бизнес-логики



Заключение

- В ходе работы было рассмотрено понятие микросервисной архитектуры и произведен обзор имеющихся средств и методологий разработки, применяющихся для коммуникации веб-сервисов.
- Результатом работы стала разработка программного обеспечения для генерации изображений с помощью нейронной сети с сетевым интерфейсом.
- Рассмотрены подходы общения разных процессов и проанализированы достоинства и недостатки каждого из методов.

Источники

1. Микросервисы: паттерны разработки и рефакторинга / Крис Ричардсон. - Санкт-Петербург [и др.] : Питер, Прогресс книга, 2020. - 542 с. - (Библиотека программиста).
2. Высоконагруженные приложения: программирование, масштабирование, поддержка: [перевод с английского] / Мартин Клеппман. - Санкт-Петербург [и др.] : Питер, Прогресс книга, 2018. - 637 с. - (Бестселлеры O'Reilly).
3. Marek Bolanowski, Kamil Zak, Andrzej Paszkiewicz, Maria Ganzha, Marcin Paprzycki, Piotr Sowinski, Ignacio Lacalle, and Carlos E. Palau. Eficiency of REST and gRPC Realizing Communication Tasks in Microservice-Based Ecosystems. IOS Press, September 2022..

Справочная информация

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

```
Latency Comparison Numbers (~2012)
----------------------------------
L1 cache reference                           0.5 ns
Branch mispredict                            5   ns
L2 cache reference                           7   ns                      14x L1 cache
Mutex lock/unlock                           25   ns
Main memory reference                      100   ns                      20x L2 cache, 200x L1 cache
Compress 1K bytes with Zippy             3,000   ns        3 us
Send 1K bytes over 1 Gbps network       10,000   ns       10 us
Read 4K randomly from SSD*             150,000   ns      150 us          ~1GB/sec SSD
Read 1 MB sequentially from memory     250,000   ns      250 us
Round trip within same datacenter      500,000   ns      500 us
Read 1 MB sequentially from SSD*     1,000,000   ns    1,000 us    1 ms  ~1GB/sec SSD, 4X memory
Disk seek                           10,000,000   ns   10,000 us   10 ms  20x datacenter roundtrip
Read 1 MB sequentially from disk    20,000,000   ns   20,000 us   20 ms  80x memory, 20X SSD
Send packet CA→Netherlands→CA      150,000,000   ns  150,000 us  150 ms
```