

Сервис-ориентированная архитектура (SOA)

Определение: SOA - это архитектурный стиль, который подразумевает создание приложений на основе использования различных слабо связанных, взаимодействующих между собой сервисов. Каждый сервис включает в себя часть бизнес-логики и обеспечивает конкретную функциональность.

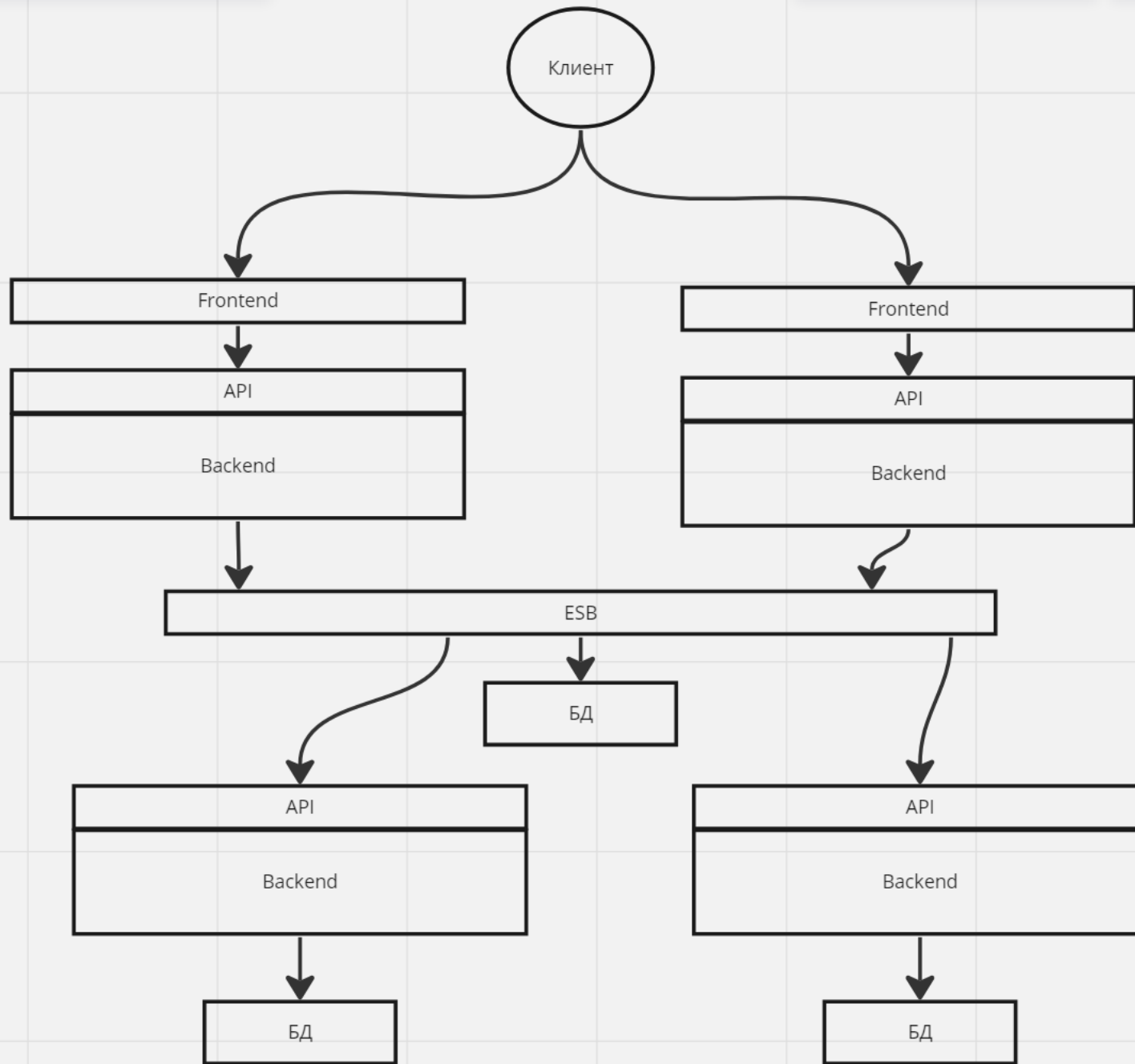
Структура: В SOA приложения разрабатываются как набор сервисов, каждый из которых может быть развернут на разных машинах или в разных облачных средах. Это позволяет достичь высокой масштабируемости и эффективности распределения ресурсов.

Взаимодействие компонентов: В сервис-ориентированной архитектуре сервисы взаимодействуют друг с другом посредством обмена сообщениями. Обычно это происходит с помощью таких протоколов, как HTTP, REST или SOAP. Сообщения могут быть синхронными или асинхронными, в зависимости от требований к приложению.

На практике SOA часто сопровождается использованием ESB (Enterprise Service Bus). ESB (или “шина”) — это набор принципов, направленных на сокращение прямых связей между сервисами и системами. Вместо прямой интеграции используется отдельный инструмент (асинхронный), который управляет всеми интерфейсами системы и логикой их маршрутизации и преобразования данных.

Преимущества: Главное преимущество SOA - это возможность повторного использования сервисов, что приводит к уменьшению дублирования кода и упрощает разработку и поддержку приложений. Сервис-ориентированная архитектура также обеспечивает высокую масштабируемость и эффективное распределение ресурсов.

Недостатки: Недостатками могут являться сложность управления такой архитектурой, необходимость решать вопросы координации действий между сервисами, а также потенциальные проблемы с производительностью из-за необходимости обработки сообщений. Создание ESB-команды, которая становится узким местом. Это усложняет процесс проектирования, поскольку возникают вопросы о необходимости новых сервисов и о разделении системы на отдельные сервисы.



Выводы:

Сервисы представлены как набор отдельно развертываемых единиц.

Эти сервисы общаются между собой по строго стандартизированным протоколам.

Каждый из них выполняет ограниченный набор функций, специализированных для повторного использования в разных процессах.

Часто для обработки бизнес-процессов используются промежуточные инструменты, такие как ESB-шины, BPM-оркестраторы.

Когда можно использовать?

- Средний и высокий уровни зрелости: Если ваш проект или организация достигла среднего или высокого уровня зрелости, в котором существуют сложные и взаимосвязанные бизнес-процессы, то использование SOA может упростить эти процессы. Сервисы, созданные в рамках SOA, могут быть переиспользованы в различных процессах, что повышает эффективность и уменьшает затраты на разработку.
- Цель - гибкость и масштабируемость: Если главной целью является создание гибкой и легко масштабируемой системы, то SOA может быть хорошим выбором. В рамках SOA, отдельные сервисы могут быть развернуты и обновлены независимо, что позволяет быстро адаптироваться к изменяющимся требованиям бизнеса.
- Цель - стандартизация и переиспользование: Если основная цель - стандартизация и повторное использование кода в разных частях системы, то SOA может облегчить эту задачу. Сервисы в SOA могут быть спроектированы таким образом, чтобы их можно было легко переиспользовать, что сокращает время разработки и повышает качество кода.