

Варианты архитектуры клиент-сервер (клиент-сервер- архитектура, клиент-сервер-архитектура) ? клиент-сервер клиент-сервер клиент-сервер ? клиент-сервер.

Клиент-сервер-архитектура (2-клиент-сервер)

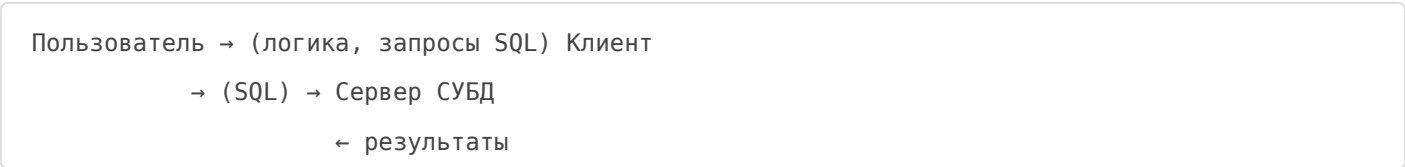
Клиент напрямую обращается к СУБД, формирует SQL и получает данные. Может быть толстый или тонкий клиент.

Клиент-сервер ? клиент-сервер ? клиент-сервер

Тип клиента	Характер	Где логика	Обработка запросов к СУБД
Толстый клиент	Fat Client	Бизнес-логика + UI на клиенте	Клиент формирует SQL и напрямую обращается к СУБД
Тонкий клиент	Thin Client	Логика минимальна	Клиент формирует простые SQL-запросы и напрямую обращается к СУБД

Клиент-сервер

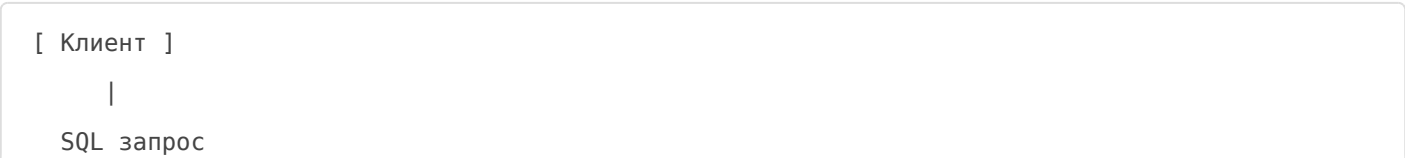
Клиент-сервер (2-клиент-сервер)



Клиент-сервер (2-клиент-сервер)



Клиент-сервер ? клиент-сервер





□

??????????? ???????????? (3-tier)

Это расширение клиент-серверной модели. Вводится промежуточный уровень приложения (API), который обрабатывает бизнес-логику и взаимодействует с СУБД вместо клиента. Клиент общается только с API.

????? (3 ??????)



Уровень	Что делает	К СУБД подключается?
Клиент	Интерфейс пользователя	Нет прямого доступа
Сервер приложений / API	Бизнес-логика и обработка данных	Да
Сервер СУБД	Хранит данные и отвечает на запросы	—

“ Клиент никогда не общается напрямую с СУБД — только через API сервера приложений.

Преимущества:

- Лучшая безопасность
- Обновления логики на сервере без изменения клиентов
- Меньшая нагрузка на сеть
- Масштабируемость
- Разделение ответственности

??????????? ????????????

Характеристика	2-звенная	3-звенная
Кол-во уровней	2	3
К клиенту можно подключиться к БД?	Да	Нет
Бизнес-логика	На клиенте	На сервере приложений
Обновление UI	Труднее	Легче
Масштабируемость	Ниже	Выше
Безопасность доступа к данным	Ниже	Выше
СУБД-API	Прямой	Через сервер приложений

Revision #1
Created 2026-01-17 18:43:23 UTC by Fedmog1lnkv
Updated 2026-01-17 18:54:47 UTC by Fedmog1lnkv