

Разновидности клиент-серверных архитектур.

Клиент-серверные архитектуры обычно разделяют на два вида :

1. Двухуровневая архитектура. Так сказать классический клиент-сервер. Один сервер - много клиентов.
2. Трехуровневая архитектура. Клиент - Сервер приложений - Сервер баз данных.

Двухуровневая клиент-серверная архитектура :

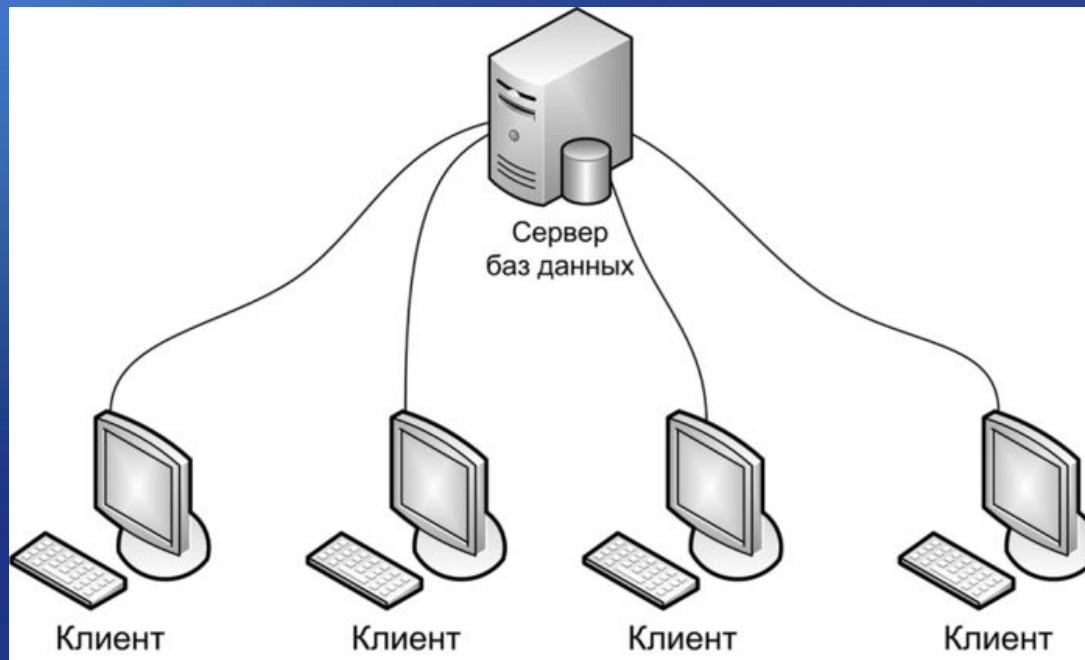
В любой сети присутствуют элементы клиент-серверного взаимодействия, чаще всего на основе двухуровневой архитектуры. Двухуровневой (two-tier, 2-tier) она называется из-за необходимости распределения трех базовых компонентов между двумя узлами (клиентом и сервером).

Двухуровневая архитектура используется в клиент-серверных системах, где сервер отвечает на клиентские запросы напрямую и в полном объеме, при этом используя только собственные ресурсы. Т.е. сервер не вызывает сторонние сетевые приложения и не обращается к сторонним ресурсам для выполнения какой-либо части запроса

Преимущества архитектуры:

1. Пониженные требования к машинам клиентов

2. Гибкость
архитектуры



Недостатки

архитектуры:

1. Стоимость серверного оборудования
2. Сервер должен обслуживать специально обученный человек
3. Ляжет сервер - клиенты не смогут работать

Трехуровневая клиент-серверная архитектура:

На нижнем уровне на компьютерах пользователей расположены приложения клиентов. Приложение нижнего уровня называют «тонким» или «облегченным» клиентом. В качестве клиента может выступать обычный web-браузер.

На среднем уровне расположен сервер приложений, этот уровень обеспечивает обмен данными между пользователями и базами данных. Сервер приложений размещается в узле сети доступно всем клиентам.

На третьем, верхнем, уровне расположен удаленный специализированный сервер базы данных, принимающий информацию от сервера приложений. Сервер баз данных выделен для услуг обработки данных и файловых операций.

Преимущества

архитектуры.

- 1.Разгрузка сервера баз данных от выполнения части
- 2.уменьшение размера транзакционных
- 3.упрощение структуры
- 4.еще больше сбалансирована нагрузка на сеть.



Недостатки

архитектуры:

1. Более высокая сложность создания приложений;
2. Сложнее в разворачивании и администрировании;
3. Высокие требования к производительности серверов приложений и сервера базы данных.
4. Высокие требования к скорости канала (сети) между сервером базы данных и серверами приложений.