АРХИТЕКТУРА ИНФОРМАЦИОННЫХ СИСТЕМ

КЛАССИЧЕСКОЕ ОПРЕДЕЛЕНИЕ АРХИТЕКТУРЫ

- ***Архитектура*** (лат. *architectural* — искусство проектировать и строить здания и другие сооружения (комплексы), создающие материально организованную среду, необходимую людям для их жизни и деятельности, в соответствии с современными техническими возможностями и эстетическими воззрениями общества.

ОПРЕДЕЛЕНИЯ АРХИТЕКТУРЫ ИНФОРМАЦИОННЫХ СИСТЕМ

- **архитектура** — организационная структура системы;
- **архитектура информационной системы** — концепция, определяющая модель, структуру, выполняемые функции и взаимосвязь компонентов информационной системы;
- **архитектура** — базовая организация системы, воплощенная в ее компонентах, их отношениях между собой и окружением, а также принципы, определяющие проектирование и развитие системы;

ОПРЕДЕЛЕНИЯ АРХИТЕКТУРЫ ИНФОРМАЦИОННЫХ СИСТЕМ

- **архитектура** — набор значимых решений по поводу организации системы программного обеспечения, набор структурных элементов и их интерфейсов, при помощи которых компонуется система вместе с их поведением, определяемым во взаимодействии между этими элементами, компоновка элементов в постепенно укрупняющиеся подсистемы, а также стиль архитектуры, который направляет эту организацию (элементы и их интерфейсы, взаимодействия и компоновку);
- **архитектура программы или компьютерной системы** — структура или структуры системы, которые включают элементы программы, видимые извне свойства этих элементов и связи между ними;
- и т.д.

- На сайте SEI (Software Engineering Institute) имеется специальный раздел, посвященный определениям архитектуры программного обеспечения http://www.sei.cmu.edu/architecture/start/glossary/community.cfm

КАКИЕ ЗАДАЧИ РЕШАЮТСЯ В РАМКАХ АРХИТЕКТУРЫ ИС?

- Конструктивно архитектура обычно определяется как набор ответов на следующие вопросы:
  - что делает система?;
  - на какие части она разделяется?;
  - как эти части взаимодействуют?;
  - где эти части размещены?.

СПОСОБЫ РАЗДЕЛЕНИЯ РАСПРЕДЕЛЕННЫХ ПРИЛОЖЕНИЙ НА ЧАСТИ

Приложения условно можно разделить на следующие функциональные части:

- Средства представления данных на экране;
- Логика представления данных на экране (описывает правила и сценарии взаимодействия пользователя с приложениями);
- Прикладная логика (правила для принятия решений, вычислительные процедуры и т.п.);
- Логика данных – операции с данными, хранящимися в некоторой базе;
- Внутренние операции БД – действия СУБД, вызываемые в ответ на выполнение запросов логики данных;
- Файловые операции – стандартные операции над файлами и файловой системой.

ТИПОВЫЕ ФУНКЦИОНАЛЬНЫЕ КОМПОНЕНТЫ ИС

ТИПОВЫЕ ФУНКЦИОНАЛЬНЫЕ КОМПОНЕНТЫ ИС

- Пользовательский интерфейс
  - Средства представления (Presentation Services (PS))
  - Логика представления (Presentation Logic (PL))

ТИПОВЫЕ ФУНКЦИОНАЛЬНЫЕ КОМПОНЕНТЫ ИС

- Бизнес-логика
  - Прикладная логика (Business or Application Logic (BL))
  - Логика данных (Data Logic (DL))

ТИПОВЫЕ ФУНКЦИОНАЛЬНЫЕ КОМПОНЕНТЫ ИС

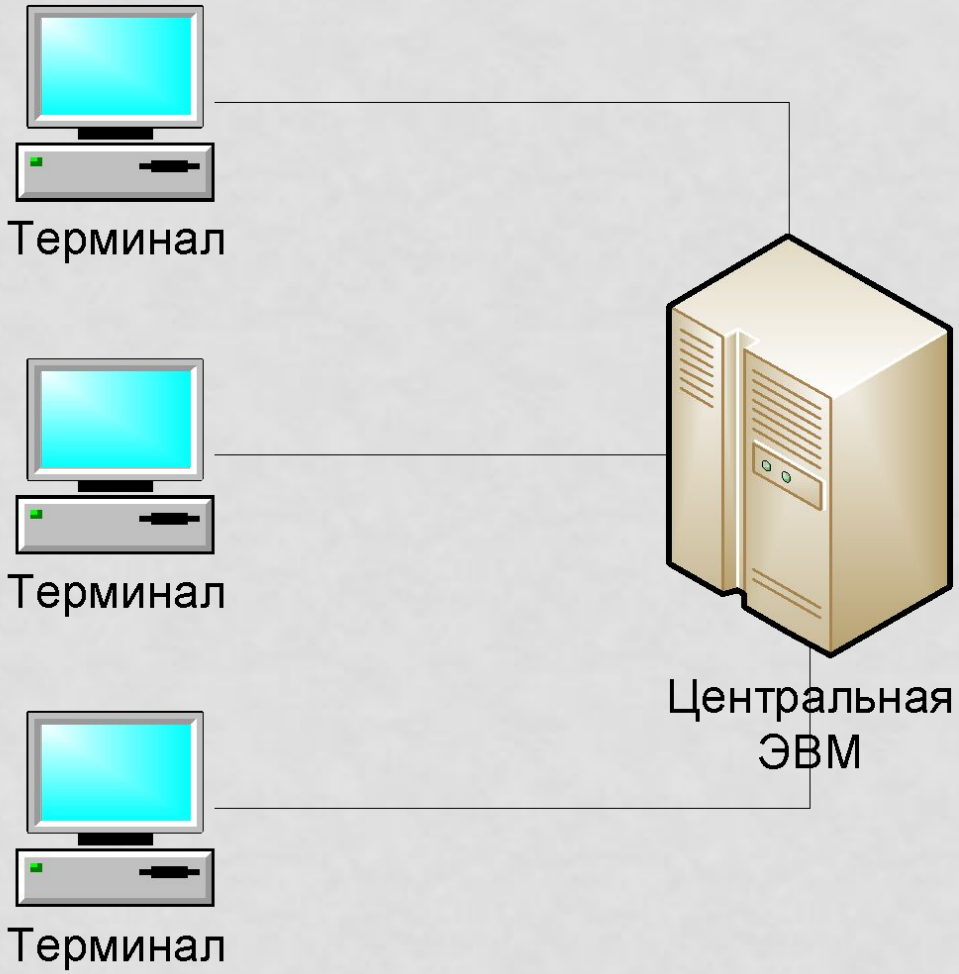
- Управление данными
  - Средства управления БД (Data Services (DS))
  - Средства управления файлами (File Services (FS))

ДВУХЗВЕННЫЕ АРХИТЕКТУРЫ РАСПРЕДЕЛЕННЫХ ИС

Двухзвенные архитектуры описывают разделение функций приложения между двумя компьютерами:

- Централизованная обработка данных;
- Архитектура «файл-сервер»
- Архитектура «клиент-сервер»

ЦЕНТРАЛИЗОВАННАЯ АРХИТЕКТУРА



ЦЕНТРАЛИЗОВАННАЯ АРХИТЕКТУРА

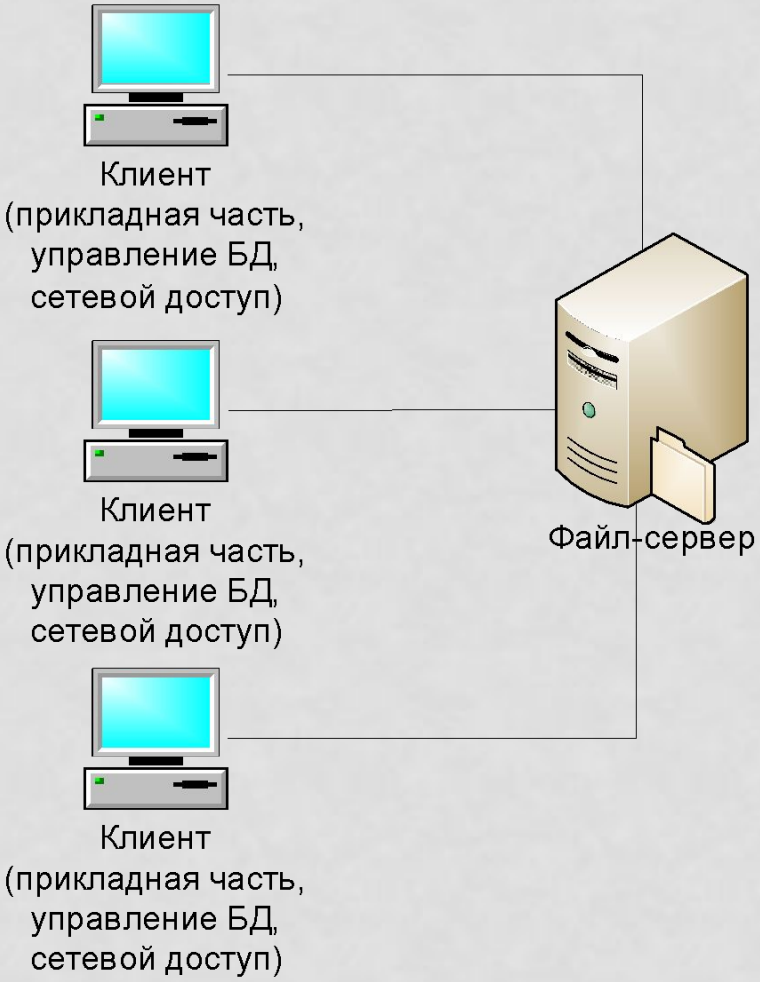
Достоинства:

- пользователи совместно используют дорогие ресурсы ЭВМ и дорогие периферийные устройства
- централизация ресурсов и оборудования облегчает обслуживание и эксплуатацию вычислительной системы
- отсутствует необходимость администрирования рабочих мест

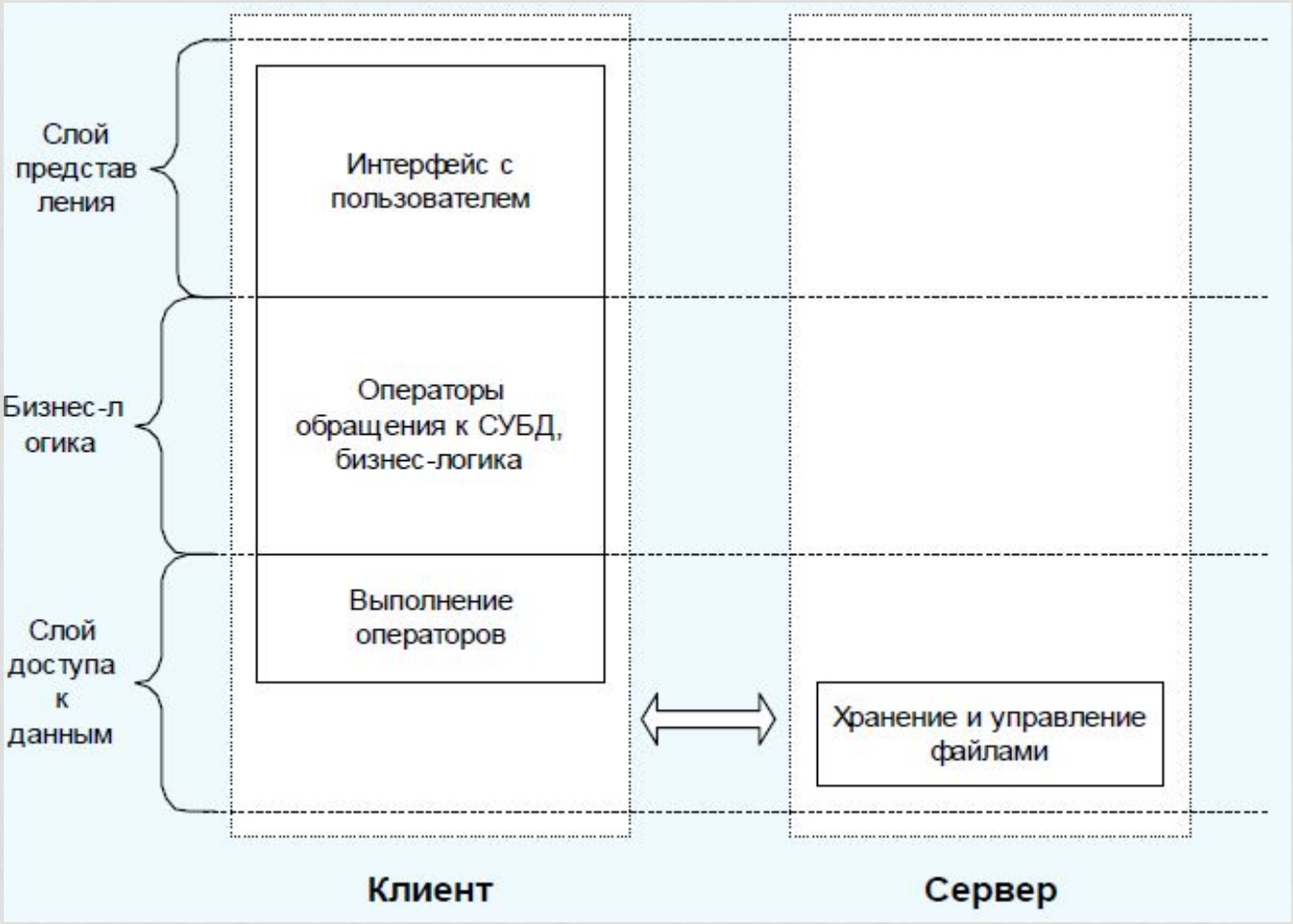
Главный недостаток:

- пользователи полностью зависят от администратора хост-ЭВМ

АРХИТЕКТУРА «ФАЙЛ-СЕРВЕР»



АРХИТЕКТУРА ФАЙЛ-СЕРВЕР



АРХИТЕКТУРА «ФАЙЛ-СЕРВЕР»

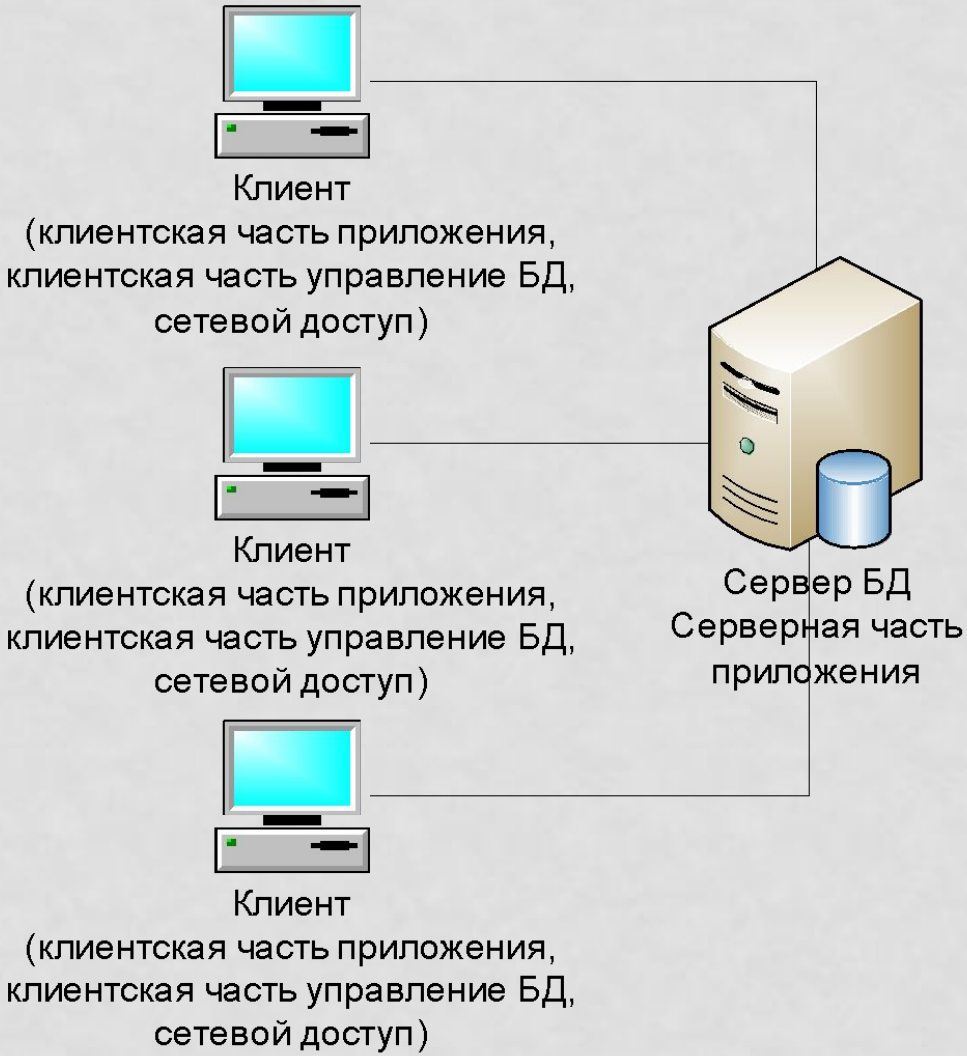
Достоинства:

- многопользовательский режим работы с данными
- удобство централизованного управления доступом
- низкая стоимость разработки
- высокая скорость разработки
- невысокая стоимость обновления и изменения ПО

Недостатки:

- проблемы многопользовательской работы с данными
- низкая производительность
- плохая возможность подключения новых клиентов
- ненадежность системы

ДВУХУРОВНЕВАЯ АРХИТЕКТУРА «КЛИЕНТ-СЕРВЕР»



АРХИТЕКТУРА КЛИЕНТ-СЕРВЕР

ДВУХУРОВНЕВАЯ АРХИТЕКТУРА «КЛИЕНТ-СЕРВЕР»

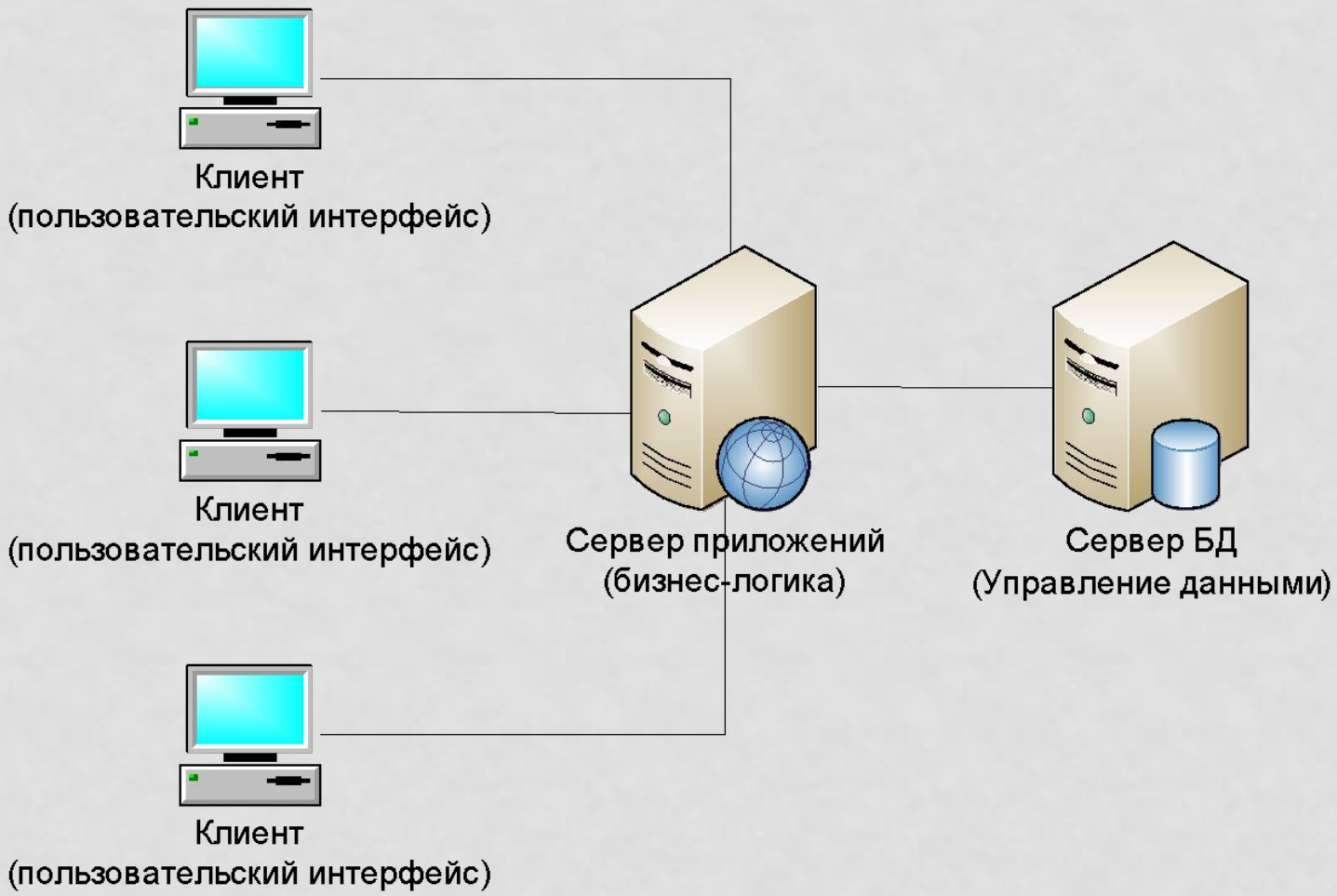
Достоинства:

- возможность распределить функции вычислительной системы между несколькими независимыми компьютерами
- все данные хранятся на защищенном сервере
- поддержка многопользовательской работы
- гарантия целостности данных

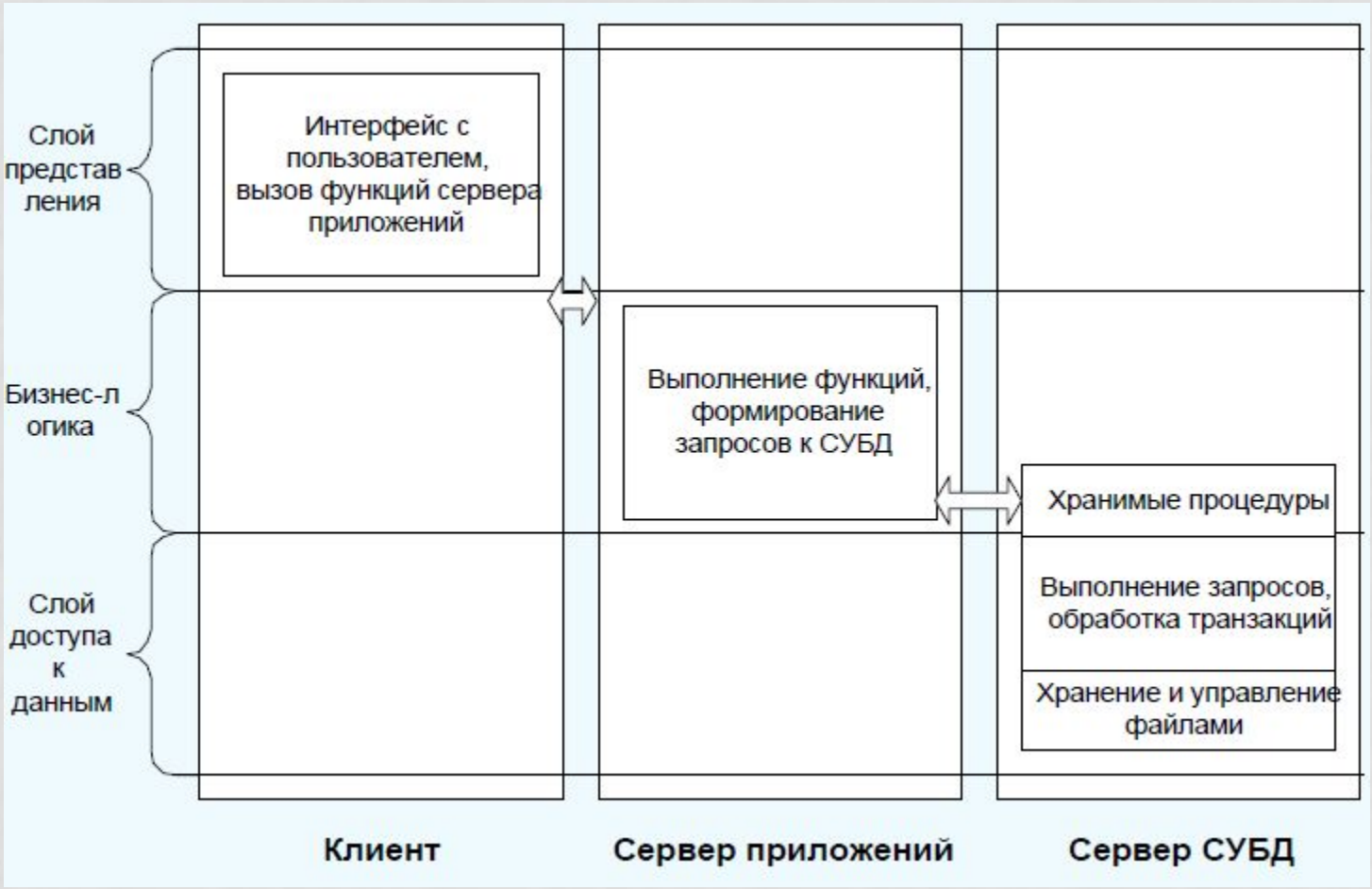
Недостатки:

- неработоспособность сервера может сделать неработоспособной всю вычислительную сеть
- сложное администрирование
- высокая стоимость оборудования
- бизнес логика приложений осталась в клиентском ПО

МНОГОУРОВНЕВАЯ АРХИТЕКТУРА «КЛИЕНТ-СЕРВЕР»



МНОГОУРОВНЕВАЯ АРХИТЕКТУРА КЛИЕНТ-СЕРВЕР



МНОГОУРОВНЕВАЯ АРХИТЕКТУРА «КЛИЕНТ-СЕРВЕР»

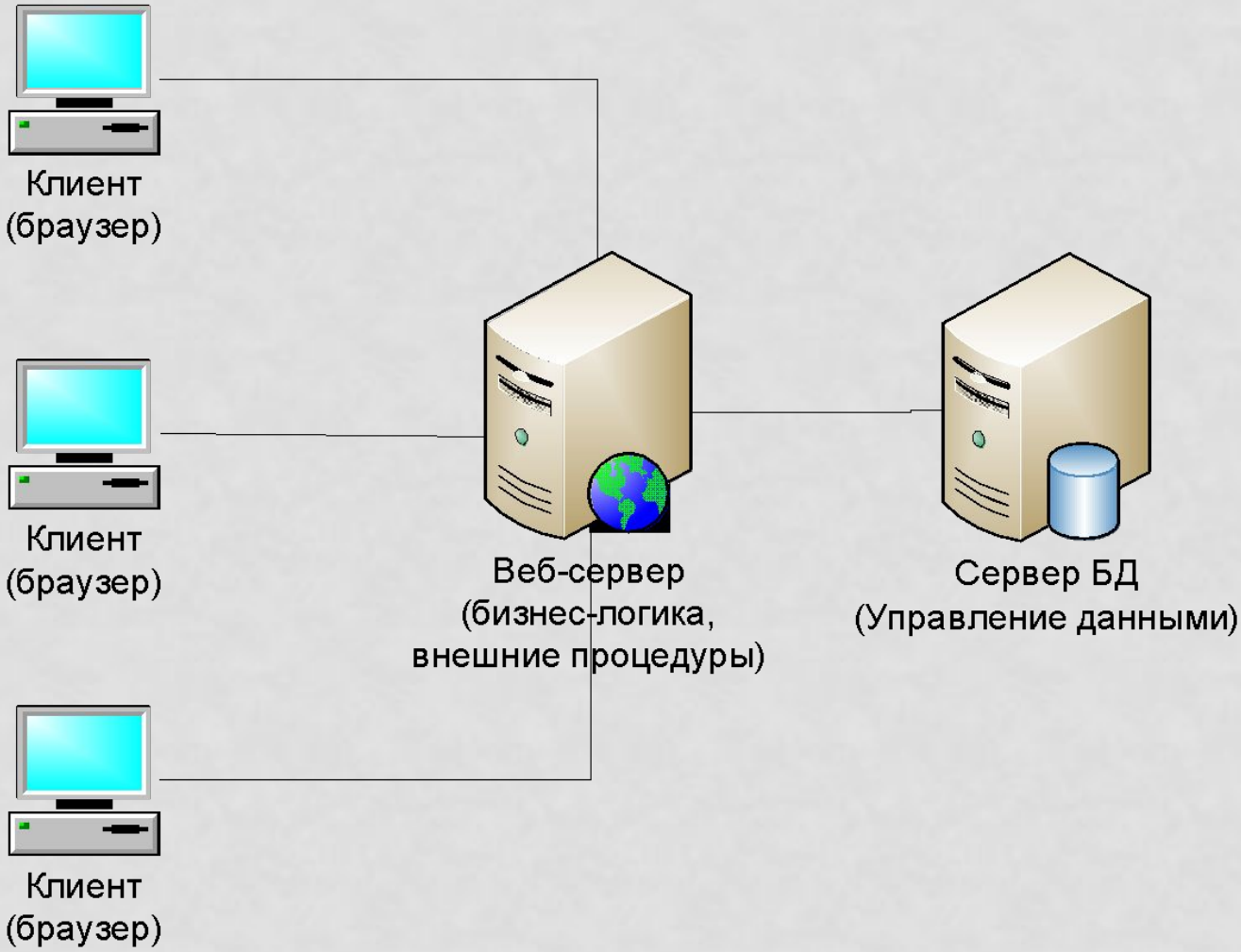
Достоинства:

- клиентское ПО не нуждается в администрировании
- масштабируемость
- конфигурируемость
- высокая безопасность и надежность
- низкие требования к скорости канала между терминалами и

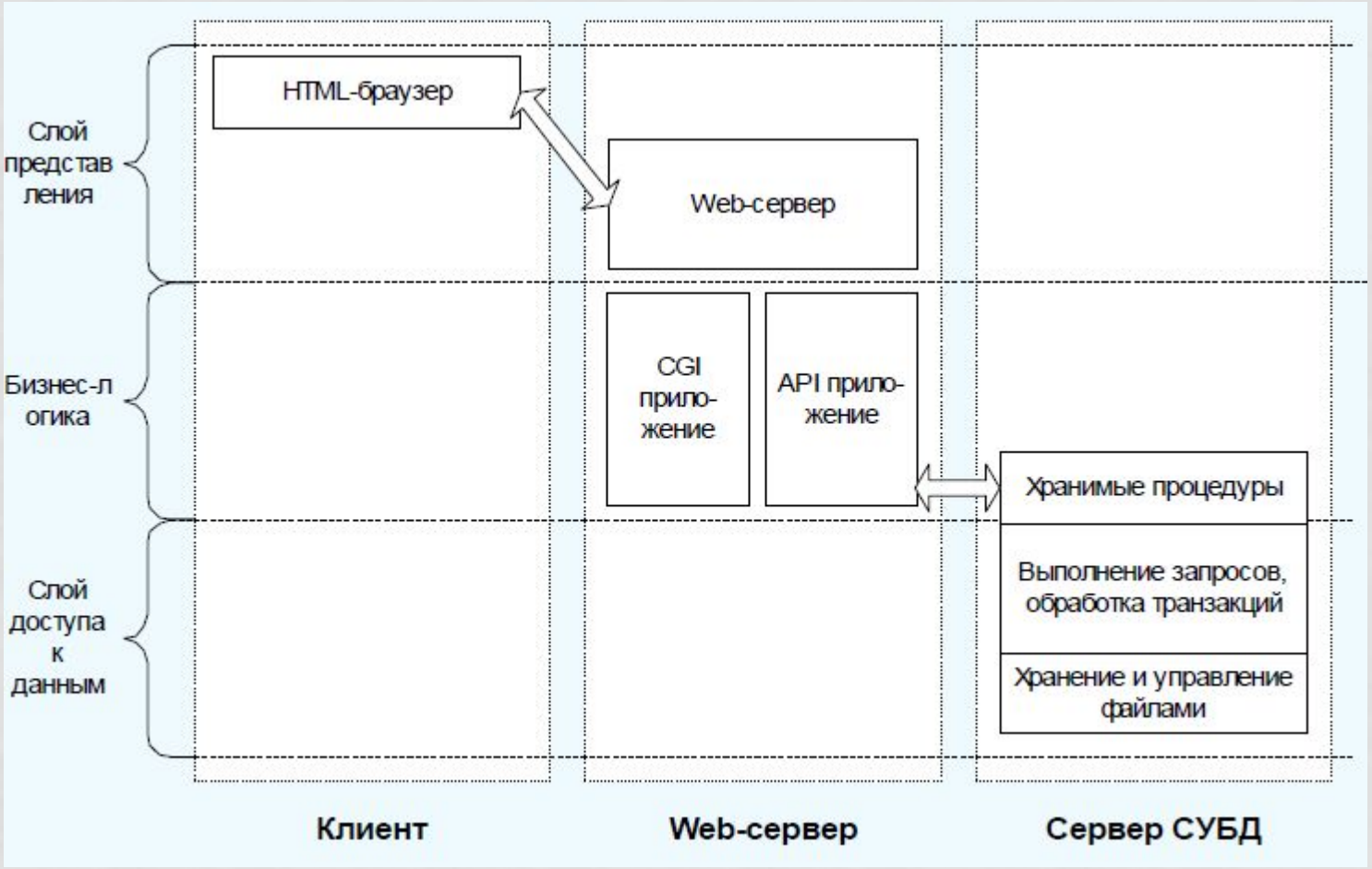
Недостатки:

- сложность администрирования и обслуживания
- более высокая сложность создания приложений
- высокие требования к производительности серверов приложений и сервера базы данных
- высокие требования к скорости канала (сети) между сервером базы данных и серверами приложений

АРХИТЕКТУРА ВЕБ-ПРИЛОЖЕНИЙ



АРХИТЕКТУРА ВЕБ-СИСТЕМ



ОСОБЕННОСТИ АРХИТЕКТУРЫ ВЕБ-ПРИЛОЖЕНИЙ

Отсутствие необходимости использовать дополнительное программное обеспечение на стороне клиента

Возможность подключения к централизованному хранилищу данных

Недоступность при отсутствии клиентов

Достаточно низкая скорость работы веб-сервера

Способность работать с большим количеством клиентов