

обмениваясь данными по сети. По такой схеме построены системы обработки данных на основе СУБД и другие системы.

Выделяют клиентскую и серверную стороны приложения. Клиентская сторона приложения функционирует на рабочем месте пользователя. Серверная сторона функционирует на специализированном комплексе, требуемый набор стандартного программного обеспечения, систему управления базами данных и собственно структуры данных.

Концепция активного сервера, который использует механизм хранимых процедур. Часть прикладного компонента переносится на сервер. Процедуры хранятся в словаре базы данных, разделяются между несколькими клиентами и выполняются на том же компьютере, что и SQL-сервер.

Преимущества такого подхода:

- возможно централизованное администрирование прикладных функций;

- значительно снижается сетевой трафик.

Недостаток — ограниченность средств разработки хранимых процедур по сравнению с языками общего назначения.

Обычно используется смешанный подход:

- простейшие прикладные функции выполняются хранимыми процедурами на сервере;

- более сложные прикладные функции реализуются на клиенте непосредственно в прикладной программе.

Клиентская часть приложения взаимодействует с клиентской частью программного обеспечения управления базами данных, которая, является индивидуальным представителем СУБД для приложения. Интерфейс между клиентской частью приложения и клиентской частью сервера баз данных, основан на использовании языка SQL. Клиентская часть сервера баз данных, используя средства сетевого доступа, обращается к серверу баз данных, передавая ему текст оператора языка SQL.

К.С. Копертехов (ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **С.А. Лукашевич**, ст. преподаватель

ТОЛСТЫЙ И ТОНКИЙ КЛИЕНТЫ В КЛИЕНТ-СЕРВЕРНОЙ АРХИТЕКТУРЕ

Тонкий клиент – терминал сети без жестких дисков, вычислительная мощность которого и объем памяти определяются задачами пользователя. Все программы и приложения, хранящиеся на сервере, становятся доступными для пользователя при включении его устройства и выполнении процедуры регистрации на сервере.

Клиент называется тонким, если он не содержит вовсе или содержит лишь малую часть бизнес-логики. К толстым относятся клиенты со значительной долей бизнес-логики. Преимуществ тонких клиентов представлены в таблице 1.

Таблица 1 — Преимущества тонких клиентов

Централизация администрирования настольных устройств	За счет централизованного оперирования приложениями и их модификациями, выполняемыми на сервере, они становятся доступными для всех пользователей сразу, не требуется контакт с отдельными пользователями.
Возможность контроля за действиями пользователя	Благодаря отсутствию накопителей на рабочем месте, пользователь не может привносить в конфигурацию программного обеспечения что-то свое, устанавливая собственные программы.
Мобильность пользователей	Пользователь, не привязанный к конкретному рабочему месту, может произвольно перемещаться в пределах локальной сети, применять устройства дистанционного доступа.
Снижение стоимости эксплуатации оборудования	При не слишком большом различии в стоимости оборудования тонкий клиент заметно дешевле в эксплуатации.

Технология «тонкий клиент-сервер» базируется на трех основных составляющих:

- стопроцентное выполнение прикладных задач на сервере;
- многопользовательская операционная система;
- технология распределенного отображения пользовательского интерфейса приложений.

Пользователи имеют возможность одновременно заходить в систему и выполнять приложения на сервере в разных, защищенных друг от друга сессиях сервера.

В системе с использованием тонкого клиента по сети на сервер передаются сигналы, отражающие движение мыши. А сервер производит соответствующие действия и формирует изменения экрана пользователя и передает эти изменения тонкому клиенту.

Технология тонких клиентов обеспечивает высокую производительность даже на низкопроизводительных рабочих местах, за счёт использования вычислительных ресурсов терминального сервера.

Толстый клиент — это приложение, обеспечивающее расширенную функциональность независимо от центрального сервера.

Достоинства:

- режим многопользовательской работы;

– предоставляет возможность работы даже при обрывах связи с сервером;

– имеет возможность подключения к банкам без использования сети Интернет.

Недостатки:

– большой размер дистрибутива;

– многое в работе клиента зависит от того, для какой платформы он разрабатывался;

– при работе с ним возникают проблемы с удаленным доступом к данным;

– довольно сложный процесс установки и настройки;

– сложность обновления и связанная с ней неактуальность данных.

Большинство современных средств быстрой разработки приложений (RAD), которые работают с различными базами данных, реализует стратегию: «толстый» клиент обеспечивает интерфейс с сервером базы данных через встроенный SQL. Такой вариант реализации системы с «толстым» клиентом, обычно обеспечивает недопустимо низкий уровень безопасности. Кроме того, данную систему почти невозможно перевести на Web-технологии, так как для доступа к серверу базы данных используется специализированное клиентское ПО.

Д.А. Костюченко (ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Е.А. Дей**, канд. физ.-мат. наук, доцент

РАЗРАБОТКА УНИВЕРСАЛЬНОЙ СИСТЕМЫ УПРАВЛЕНИЯ СВЕТОВЫМИ ПРИБОРАМИ ПО ПРОТОКОЛУ ART-NET

Идея проекта. В основе данного проекта лежит идея создания аппаратной платформы, позволяющей управлять световым оборудованием при помощи протокола Art-Net с возможностью также работать в автономном режиме.

Постановка задачи. При разработке устройства были поставлены следующие задачи:

- поддержка протокола Art-Net;
- возможность работать в автономном режиме;
- поддержка использования нескольких устройств и расширения проекта новыми устройствами (расширяемость и масштабируемость);
- обеспечение низкой себестоимости сборки.

Общая характеристика Art-Net. Сетевой протокол Art-Net представляет собой реализацию протокола управления освещением DMX512-A (*Асинхронный DMX 512 протокол*) через User Datagram