

на более высоком уровне. Пользователю необходимо предоставить доступ к файлу MS Access и ограничить его возможности работы с ним.

Повышение производительности является другой, иногда не менее важной причиной перехода к клиент-серверной архитектуре. Требования к производительности приложения зависят от типа задач, которые оно должно выполнять. В таблице 1 указаны некоторые задачи, для которых, вероятно, лучше перейти к клиент-серверной архитектуре.

Таблица 1 – Преимущества использования сервера для некоторых задач

| Тип обработки   | Преимущества использования сервера   |
|---|--|
| Большой объем транзакций, выполняемых над многими таблицами                 | Сетевой трафик уменьшается за счет передачи по сети только запросов клиентов и результатов обработки. Вследствие этого уменьшается время ожидания обработки запроса при той же стоимости аппаратуры. |
| Выполнение нетривиальных запросов над совместно используемыми данными       | Улучшенное управление приоритетами доступа. Сетевой трафик уменьшается, поскольку по сети передаются только результаты обработки данных  |
| Изменение небольшого числа полей (даже одного) в большом количестве записей | Сетевой трафик значительно уменьшается, поскольку по сети передаются лишь измененные поля  |
| Массивная обработка данных над большими таблицами                           | Мощность серверной СУБД в сочетании с аппаратными возможностями самого сервера позволяет значительно повысить производительность   |

Благодаря возможности разрабатывать клиентские приложения для доступа к данным сервера, Access можно активно применять не только на малых, но и на средних и крупных предприятиях для создания клиент-серверных приложений.

#### Список цитированных источников

1. Бекаревич, Ю.Б. Самоучитель Access 2010 / Ю.Б. Бекаревич – СПб.: БХВ-Петербург, 2011.
2. Гурвиц, Г.А. Microsoft Access 2007. Разработка приложений на реальном примере / Г.А. Гурвиц – СПб.: БХВ-Петербург, 2007.

УДК 51-74

*Махунова М.А.*

*Научный руководитель: к.т.н., доцент Кофанов В.А., доцент Аверина И.Н.*

### ИНТЕРФЕЙС СЕТЕВОГО ПРИЛОЖЕНИЯ ACCESS

В современном мире уже разработано достаточно большое количество специализированных программ для ведения управленческого учета, например, «1С: Управление небольшой фирмой». Но приобретение готовой программы не всегда целесообразно и материально оправдано, так как не гарантирует отражения специфики деятельности предприятия.

В большинстве случаев в каждой фирме, использующей компьютерную технику, уже установлено программное обеспечение. Как правило, это операционная система Windows и пакет программ Microsoft Office, где присутствуют такие программы как Word, Excel, а также система управления базами данных Access. Access позволяет обеспечить контроль над большим объемом информации и совместную работу пользователей с базой данных в сети. По сравнению с

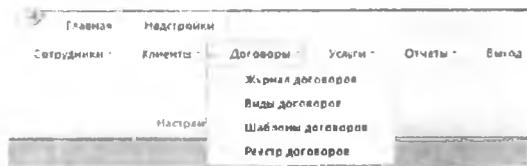
другими аналогичными программами Access, с одной стороны, имеет удобный набор средств разработчика и высокую совместимость и интеграцию ее баз данных с другими программами и форматами файлов.

Создать базу данных (запросы, отчеты и др.) в Access для сотрудника фирмы на много сложнее, чем использовать ее. Поэтому целью нашей работы является разработка такого приложения Access, которое способствует оптимизации бизнес-процессов, позволяя сотруднику фирмы:

- пополнять базу данных;
- идентифицироваться в базе данных;
- работать с базой данных в сети;
- экспортировать данные в Word и Excel.

В подобном приложении необходимо ограничить пользователя от большинства инструментов управления базой данных, для того чтобы его действия не привели к необоснованным изменениям базы данных. В настройках базы данных пункт «отключить» приведет к скрытию всех вкладок кроме вкладки «главная», при этом на самой вкладке «главная» станут неактивными большинство элементов редактирования.

Взамен скрытых элементов мы создали с помощью VBA дополнительную вкладку, на которой расположили элементы главного меню. В меню объединяют последовательности и группы команд, одну из которых может выбрать работник для совершения очередного действия. Как правило, названия команд в меню достаточно информативны, так что пользователь может легко найти нужную ему команду. Команды для решения близких задач всегда объединяются в группу. В одном меню команды объединяют на основе одного из двух принципов: либо это различные действия над одним объектом, либо однотипные действия над различными объектами. Эти элементы позволяют быстро сориентировать внимание пользователя на возможных действиях с открытой БД (рис. 1).



*Рисунок 1 – Главное меню*

Работать с БД может любой пользователь, имеющий права доступа к пунктам главного меню. Доступ предоставляет администратор БД. В зависимости от полученных прав у каждого пользователя свои элементы главного меню. Элемент главного меню, который предоставляет право добавлять в БД нового пользователя, есть только у администратора. Добавляя нового пользователя, администратор обязательно должен указать его имя, фамилию, отчество, дату рождения, должность и права доступа к определенным пунктам меню (рисунок 2).

Определенные права доступа получает только этот пользователь, который прошел идентификацию в этой БД. Идентификация пользователя выполняется в момент открытия БД с помощью диалогового окна, показанного на рисунке 3. В этом окне пользователь должен выбрать свое имя, фамилию и отчество и ука-

зять пароль. В случае отсутствия в БД введенного пароля для выбранного пользователя система укажет на возможные ошибки при вводе данных, либо предложит завершить работу с открываемой БД.



**Рисунок 2 – Диалоговое окно для разграничения прав пользователей**



**Рисунок 3 – Пользовательское диалоговое окно входа в приложение**

Поскольку диалоговое окно, требует выбор (ввод) пользователя, исходим из того, что требуется дополнить базу данных таблицей Сотрудники, которая будет фиксировать основные данные сотрудников и пароль для входа в приложение, а также определять права доступа пользователей. Реляционные базы данных позволяют выполнить эти изменения в ее структуре в любой момент без какого-либо ущерба для ранее разработанных объектов приложения. Для автоматической идентификации пользователя определено ключевое поле Префикс (Prefix) с типом данных Счетчик (AutoNumber).

При вводе пароля, приложение (специальный макрос) кодирует его и сверяет с заданным паролем (пароль хранится в виде программного кода) в таблице Сотрудники.

Если пароль отсутствует, то работа программного комплекса прекращается. "Подсматривать" пароль через ODBC не имеет смысла, так как в таблице он зашифрован, а вводить его в форму требуется в раскодированном виде. Пароль, указанный пользователем, зашифровывается и сравнивается с хранящимся в

таблице. В случае совпадения все глобальные переменные, управляющие доступом к пунктам меню, формам и отчетам, получают значения, соответствующие полям Access01—Access14 таблицы Сотрудники. Рассмотрим содержание полей таблицы на рисунке 4.

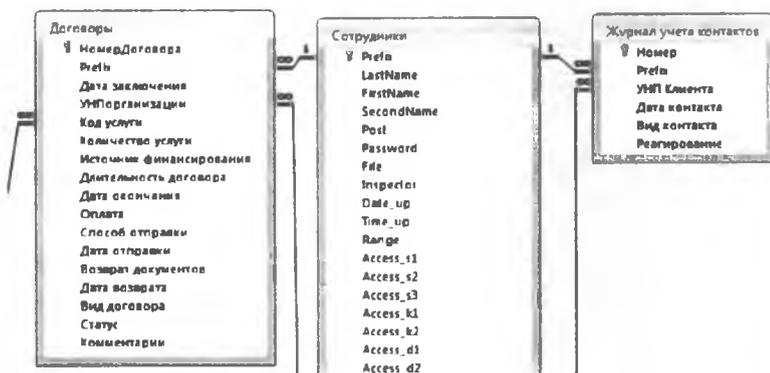


Рисунок 4 – Схема таблицы Сотрудники

LastName – Фамилия,  
 FirstName – Имя,  
 SecondName – Отчество,  
 Post – Должность,  
 Password – Пароль,  
 File – ссылка на фотографию сотрудника,  
 Inspector – логическое поле (да / нет), если стоит значение истина, то это означает, что данный сотрудник наделен правами администратора.

Date\_up – дата последнего изменения пароля,  
 Time\_up – время последнего изменения пароля,  
 Значения полей Access01 – Access07 присваиваются глобальным переменным в результате работы формы Login, и на основании того, False это или True, обеспечивается доступ к пунктам меню, формам и расчетам программного комплекса (права доступа к пунктам меню).

Прошедшие идентификацию пользователи могут вносить изменения только в ту часть БД, к которой им предоставлен доступ. Для этих целей было разработано соответствующее количество диалоговых окон. В основном структура этих окон была одинаковой с точки зрения интерфейса. Все отличие было визуальным, т.е. в каждом окне располагался определенный набор элементов для изменения соответствующих полей таблиц БД. Типичный пример такого диалогового окна приведен на рисунке 5.

Одним из главных объектов диалогового окна, показанного на рисунке 5, является объект «вкладка», позволяющий создавать двухстраничную форму. На первой странице, как правило, расположен объект список, отображающий все записи соответствующей таблицы БД. На второй странице расположены объекты, позволяющие редактировать старые или добавлять новые записи. В режиме

редактирования вторая страница открывается в результате выбора из списка на первой странице соответствующей записи, а в режиме добавления записи – путем нажатия кнопки «добавить».



*Рисунок 5 – Диалоговое окно «журнал договоров»*

Код процедур событий объектов (список, вкладка и кнопка) позволяет следить за изменением данных, совершаемых пользователем и предупреждать его о необратимых последствиях, таких как удаление.

Для удобства работы в сети мы разделили базу данных с помощью инструмента «база данных Access» на два файла: файл с данными и файл с компонентами интерфейса. Файл с данными расположили на сервере, а файл с компонентами интерфейса – на рабочих станциях. Этот прием позволил нам снизить нагрузку на сеть, повысить скорость работы интерфейса, упростить установку новых версий приложений и повысить сохранность данных.

Разделение базы данных может перевести к тому, что файл с данными, хранящийся на сервере, останется не защищен от внешнего доступа. Поэтому при разделении базы данных файл с данными обязательно необходимо защитить паролем. В этом случае пользователь может получить доступ к данным только через файл с компонентами интерфейса.

Результаты выборки передавались либо на форму, либо в заранее сформированный шаблон отчета Access или Word. В Access удобно формировать отчет на основе нескольких записей, в Word – на основе одной (заявление, договор, акт).

Передача данных из Access в Word происходит с помощью инструмента «слияние» в автоматическом режиме. Для реализации такого трюка в документе Word, представляющего собой шаблон договора, предварительно вручную было выполнено слияние. Код таких действий записывался с помощью макрорекордера в VBA в виде исполняющей процедуры.

Запуск процедуры слияния происходил после нажатия на кнопку Word диалогового окна «журнал договоров» (рисунок 5). Для того чтобы оба приложения не конфликтовали друг с другом, в приложении Word была подключена библиотека объектов Access, а в приложении Access подключена библиотека объектов Word. Такой подход помог значительно сократить объем кодирования процесса слияния.

Разработанное в такой форме приложение Access очень удобно использовать в локальной вычислительной сети при работе нескольких человек одновременно.

#### Список цитированных источников

1. Гурвиц, Г.А. Microsoft Access 2007. Разработка приложений на реальном примере / Г.А. Гурвиц. – СПб.: БХВ-Петербург, 2007. – 672 с.
2. Бекаревич, Ю.Б. Самоучитель Access 2010 / Ю.Б. Бекаревич. – СПб.: БХВ-Петербург, 2011. – 432 с.

УДК 338.001.36

*Махунова М.А.*

*Научный руководитель к.э.н., доцент Потапова Н.В.*

### **СТОИМОСТНО-ОРИЕНТИРОВАННАЯ КОНЦЕПЦИЯ РЕСТРУКТУРИЗАЦИОННОГО КОНТРОЛЛИНГА**

В современных быстроменяющихся условиях рыночной экономики становится очевидным, что любое предприятие для выживания и сохранения конкурентоспособности должно постоянно корректировать свою деятельность с учетом требований внешней среды. Это вызвано тем, что изменения факторов внешней среды может привести к возникновению определенного дисбаланса между предприятием и средой и повлечь необходимость адаптации предприятия. В этой связи, предприятие должно обладать способностью своевременно осуществлять адекватные изменения. Такие изменения на предприятии могут быть осуществлены при помощи реструктуризации.

Реструктуризация – инструмент быстрого и гибкого реагирования бизнес-единиц (предприятий) на изменения внешней среды.

Реструктуризация предприятия – это один из методов его реформирования посредством проведения преобразований в организационной, финансово-экономической, производственно-технической и социальной подсистемах с целью адаптации к быстро меняющимся условиям поведения внешней среды для повышения эффективности его функционирования и конкурентоспособности.

Таким образом, если реформирование предприятия подразумевает любые возможные преобразования и изменения, то реструктуризация является одним из конкретных методов реформирования предприятия путем структурной перестройки его основных подсистем.

Реструктуризация, зависящая от целевых установок и стратегии предприятия, может быть оперативной или стратегической. В зависимости от количества структурных изменений существует комплексная и частичная реструктуризация.

Оперативная реструктуризация означает существенное изменение структуры компании, преследующее такие цели как возможность финансового оздоровления или улучшения платежеспособности компании.

Стратегическая реструктуризация также приводит к изменениям в структуре компании, но в первую очередь направлена на улучшение привлекательности компании для инвесторов, расширение её внешнего финансирования, а также повышение стоимости самой компании.

Комплексная реструктуризация в основном проводится в несколько этапов, постепенно затрагивая все элементы компании.

Частичная реструктуризация вносит изменения лишь в один или несколько элементов предприятия.