

**ПРОГРАММИРОВАНИЕ БАЗ ДАННЫХ В DELPHI-7.
СРАВНЕНИЕ МЕХАНИЗМОВ ДОСТУПА К ДАННЫМ BDE И ADO**

Т.Н.РУБЛЕВА

E.mail. ksucta@elcat.kg

Бул макалада Delphi-7 программалоо чөйрөсүндө маалыматтарга жетүү механизмдеринин салыштыруу көрсөтүлөт.

В статье описывается сравнение механизмов доступа данных в среде программирования Delphi-7.

This article explains comparison of mechanisms for accessing data in a programming environment Delphi-7.

Без баз данных сегодня не обойтись, так как в любой организации, фирме имеется масса данных, которые нужно не только сохранить, но и обработать, получить требуемые запросы и комплексные отчеты. Надо отметить, что программирование баз данных – очень серьезный раздел практического программирования. Многие программисты большую часть своего времени тратят именно на проектирование баз данных и разработку приложений, работающих с ними. Недостаточно просто написать программу, взаимодействующую с БД, нужно уметь правильно спроектировать эту базу данных. И как показывает практика, для студентов самым сложными этапами становятся проектирование базы данных и выбор программного обеспечения для разработки проекта. Рассмотрим более подробно эти этапы.

1. Проектирование базы данных. Нужно отметить, что проектирование БД является очень важным этапом, от которого зависят последующие этапы разработки СУБД. Время, затраченное разработчиком на проектирование БД, обычно окупается высокой скоростью реализации проекта.

Перед созданием базы данных необходимо располагать описанием выбранной предметной области, которое должно охватывать реальные объекты и процессы, иметь всю необходимую информацию для удовлетворения предполагаемых запросов пользователя и определить потребности в обработке данных.

На основе такого описания на этапе проектирования базы данных осуществляется определение состава и структуры данных предметной области, которые должны находиться в базе данных и обеспечивать выполнение необходимых запросов и задач пользователя. Структура данных предметной области может отображаться информационно-логической моделью. На основе этой модели легко создается реляционная база данных /1/.

Основные этапы проектирования базы данных показаны на рис. 1.



Рис.1. Этапы проектирование базы данных

Таким образом, при проектировании базы данных студенту необходимо выполнить следующие действия:

- 1) в имеющихся документах необходимо выявить реквизиты, подлежащие хранению в БД;
- 2) определить функциональную зависимость между ними;
- 3) выделить ключевые и описательные реквизиты и сгруппировать реквизиты, зависимые от выделенных ключевых реквизитов;
- 4) описать полученные информационные объекты, представив их в виде таблиц;
- 5) создать таблицы в выбранной системе программирования.

2. Выбор системы программирования. В настоящее время существует достаточно большое количество программных систем, позволяющих создавать и использовать локальные (dBASE, FoxPro, Access, Paradox) и удаленные (Interbase, Oracle, SysBase, Infomix, Microsoft SQL Server) базы данных.

На практических занятиях студентам предлагается разработать базу данных в системе программирования Delphi-7 с использованием технологий доступа к данным: BDE и ADO. Рассмотрим эти технологии.

Как известно, в состав Delphi входят компоненты, позволяющие создавать программы работы с файлами данных, созданными различными системами: от dBASE до Infomix и Oracle. Кроме того, Delphi-7 позволяет программисту, используя утилиту Borland Database Desktop, создавать файлы баз данных в различных форматах.

Надо отметить, что характерной особенностью программ, созданных с помощью Delphi и предназначенных для работы с базами данных, является их зависимость от специальной библиотеки программ, которая называется BDE (Borland Database Engine – машина баз данных корпорации Borland). BDE представляет собой набор библиотек DLL, предназначенных для низкоуровневого доступа к данным самых различных форматов. BDE автоматически устанавливается в процессе установки Delphi и регистрируется в реестре 32-разрядной версии Windows. В BDE имеется собственный интерпретатор языка SQL, что позволяет создавать запросы не только к серверам БД, но и к таблицам файл-сервера /2/.

Но в Delphi-7 имеются технологии, позволяющие обойтись без BDE: ADO InterBase Express и dbExpress. Однако скорость доступа к данным, обеспечиваемая технологией ADO, существенно ниже, чем при использовании BDE, а технология InterBase Express может работать только с серверами InterBase версии 5.5 и выше. Технология dbExpress позволяет обращаться непосредственно к SQL-серверам InterBase, MySQL, Oracle, DB2.

Механизм доступа к данным – это программный инструмент, позволяющий получить доступ к базе данных и ее таблицам. Как правило, это драйвер в виде *.dll

файлов, который устанавливается на ПК разработчика (и клиента) и который используется программой для связи с БД.

Сравним механизмы доступа к данным BDE и ADO, обсудим их плюсы и минусы.

Borland Database Engine (BDE) – первая разработка фирмы Borland. Этот механизм доступа к данным позволяет обращаться к локальным и файл-серверным форматам баз данных dBase, FoxPro и Paradox, к различным серверам SQL и ко многим другим источникам данных, доступ которых поддерживался при помощи драйверов ODBC. Например, с помощью BDE можно напрямую работать с табличными файлами MS Excel /2/.

Увы, на сегодняшний день механизм доступа BDE признается устаревшим даже самой компанией Borland.

Тем не менее, многие программисты до сих пор используют данный инструмент в разработке приложений для небольших компаний. BDE имеет множество простых и удобных возможностей для программиста, таких, например, как создание таблиц программно.

Удобство работы с BDE трудно переоценить, однако нельзя не сказать и о минусах. Основной минус – распространение приложений. Если приложение использует для доступа к данным компоненты BDE, то и у клиента, который будет пользоваться вашей программой, должен быть установлен BDE. Причем, если вы использовали алиасы (псевдонимы базы данных), то настройка на эти же алиасы должна быть и у клиента.

Известно, что среди программистов, работающих в средах Delphi и C++ Builder, всегда были популярными VCL-компоненты, которые не требуют обращения к BDE. BDE, по мнению многих разработчиков, замедляют работу приложения и требуют слишком больших ресурсов памяти – оперативной и дисковой, поэтому познакомимся с другой технологией.

ActiveX Data Object (ADO) – это механизм доступа к данным, разработанный корпорацией Microsoft. Если точнее, то ADO – это надстройка над технологией OLE DB, посредством которой можно связываться с различными данными приложений Microsoft.

По своим возможностям ADO напоминает BDE, хотя конечно, является более мощным инструментом. Компания Borland разработала набор специальных компонентов для доступа к данным: TADOConnection (для подключения к БД), TADOTable (аналог TTable из BDE), TADOQuery (аналог TQuery из BDE для выполнения запросов и получения набора данных) и TADODataset (предназначенный для набора данных, полученных через SQL-запрос).

Отметим, что технология ADO, как и BDE, независима от конкретного сервера БД, имеет поддержку как локальных баз данных различных типов, так и некоторых клиент-серверных БД. Плюсов у этой технологии много.

Драйверы, разработанные корпорацией Microsoft для собственных нужд, более надежные, чем драйверы сторонних производителей. Поэтому если вам требуется работать с базами данных MS Access или для архитектуры клиент-сервер использовать MS SQL Server, то использование ADO будет наиболее предпочтительным.

Кроме того, имеется плюс и в вопросе распространения программ – во всех современных Windows встроены драйверы ADO. Другими словами, программа будет работать на любом ПК, где установлен Windows /1/.

Как ни странно, но основной минус также заключается в вопросе распространения программ: при использовании технологии ADO приходится думать о том, какая версия Windows стоит у конечного пользователя, будет ли ваша программа работать у него на ПК.

Еще один серьезный минус ADO в том, что он для подключения к БД использует довольно медлительную технологию COM. Если база данных будет содержать несколько тысяч записей, то скорость работы с таблицами может стать в сотни раз более медленной, чем если бы вы использовали BDE! Правда, на современных ПК, имеющих частоту

процессора до 2 ГГц и выше, эти замедления могут быть и незаметны, но работа с огромной базой данных на более медленных ПК превратится в сплошное ожидание.

У ADO, в сравнении с BDE, есть и другие недостатки.

В отличие от BDE, для однородного объединения таблиц из разных баз данных нельзя использовать SQL в сочетании с ADO-компонентами (причем это не зависит от того, созданы ли эти базы данных с помощью одной и той же СУБД или разных СУБД) /1/.

Есть и еще одно предостережение, которое касается сопровождения приложений, использующих ADO-компоненты. Нужно постоянно отслеживать изменения, которые вносятся разработчиками в элементы операционной среды, связанные с работой ADO-компонентов и СУБД (например, ядро Jet, с которыми работают базы данных Access).

Таким образом, рассматривая эти две технологии, можно сделать следующий вывод: в некоторых отношениях компоненты ADO мощнее компонентов BDE, но в то же время ряд возможностей компонентов BDE в них не реализован. Например, они не могут использовать словари свойств, что приводит к лишней работе при создании приложений. Не все источники данных ADO могут работать со всеми типами полей. Например, источник данных Paradox ADO не работает с графикой. Так что надо серьезно рассматривать все за и против, прежде чем выбрать, какую технологию (BDE или ADO) вы будете использовать для разработки базы данных.

При этом необходимо учесть, что если вы хотите работать с MS SQL 7 или MS SQL 2000 без ограничений, которые накладывает BDE (BDE по-прежнему полностью поддерживает только MS SQL 6.5 и обновления BDE пока не предвидится), то вам необходимо выбрать технологию ADO.

Также можно сказать относительно баз Access, что программистам, уставшим от «парадоксального» поведения Paradox+BDE, необходимо обратить на них внимание, так как начиная с версии Access 2000 Microsoft удалось догнать и перегнать старый, добрый Paradox (не менявшийся уже довольно давно) /3/.

Учитывая все вышесказанное, можно рекомендовать студентам уделять больше внимания вопросу проектирования базы данных. Кроме того, при выборе СУБД студенты могут использовать любую из предлагаемых технологий, все зависит от того, какая задача перед ними стоит.

Список литературы

1. Ачкасов В.Ю. Программирование баз данных в Delphi-7. – СамГТУ, 2010. – 255 с.
2. Фаронов В.В. Система программирования Delphi-7: Наиболее полное руководство. – СПб.: Питер, 2003. – 888 с.
3. <http://inf8.gym5cheb.ru/p28aa1.html>