

# Лабораторная работа №4. Физическое моделирование данных. Генерация баз данных

**ЦЕЛЬ РАБОТЫ:** Ознакомиться с особенностями процесса физического моделирования данных, процесса создания физических баз данных на основе моделей ERwin и применить полученные знания для разработки модели базы данных и реализации ее для СУБД Microsoft Access.

Если целью проектирования является не просто построение модели данных, а генерация полнофункциональной базы данных, то в этом случае, помимо логического моделирования данных, требуется проведение ряда работ на физическом уровне модели. Прежде всего это физическое моделирование и прямое проектирование базы данных.

## Выбор СУБД

Прежде чем начинать что-либо делать на уровне физической модели (имена объектов, типы данных и другие физические характеристики), необходимо выбрать сервер СУБД (так называемая СУБД назначения), на котором будет создана физическая схема, и задать тип данных по умолчанию, нулевые режимы и другие значения, которые ERwin будет использовать по умолчанию при генерации колонок.

Для того, чтобы выбрать СУБД, используется команда меню *Server* → *Target Server* (см. рис. 1). В открывшемся редакторе *Target Server* можно выбрать СУБД и задать тип данных по умолчанию, нулевые режимы и правила ссылочной целостности, которые будут использованы при генерации физической схемы базы данных.

С помощью групп *SQL DBMS* и *Desktop DBMS* Вы можете выбрать интересующий тип СУБД и его версию. Первая группа содержит перечень промышленных СУБД, а вторая – настольных. При переходе на другую СУБД, ERwin спрашивает, хотите ли Вы автоматически преобразовать типы данных схемы в типы данных, которые поддерживаются новой СУБД.

С помощью списка *Default Datatype* вы можете определить тип данных, который будет назначен домену *default* и использован при операции преобразования типов данных при переходе от одной СУБД к другой. Значение *CHAR(18)* означает, что все колонки будут заданы таким образом, чтобы в них можно было хранить текстовую строку длиной до 18 символов.

В редакторе также можно задать нулевые значения, присваиваемые по умолчанию для всей схемы, то есть, разрешены ли нулевые (пустые) значения в неключевых колонках (*non-key NULL option*), а также набор правил ссылочной целостности (*RI defaults*), которые

определяют, каким образом на колонках будет отражаться вставка, изменение или удаление строки в таблицу. Режим нулевых значений по умолчанию и установки ссылочной целостности различаются в зависимости от СУБД.

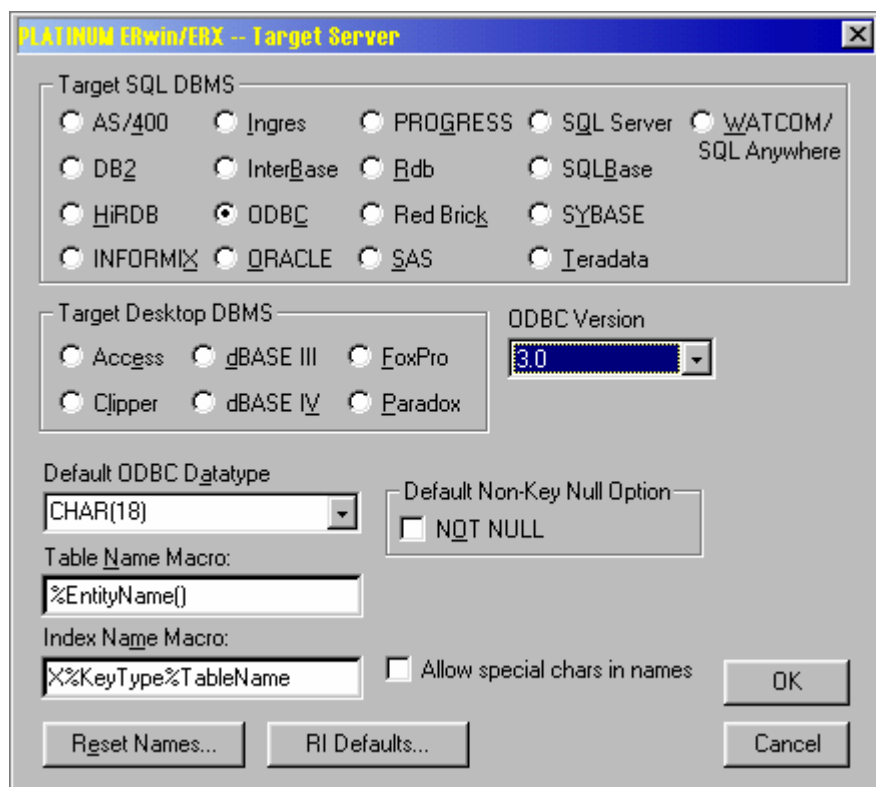


Рис. 1. Окно редактора СУБД назначения

## Задание информации физической схемы

Процесс физического моделирования данных подразумевает проведение следующих операций:

- настройка параметров таблиц и колонок;
- настройка ограничений и правил проверки значений колонок;
- настройка параметров индексов;
- создание представлений базы данных и установление их связей.

Процесс построения физической модели базы данных в ERwin существенно упрощен в связи с тем, что большая часть информации физической схемы создается автоматически из логической модели данных «на лету». Поэтому разработчику обычно остается лишь настроить параметры уже существующих объектов или добавить некоторые новые (например, представления).

Как сущности, таблицы имеют собственный редактор (рис. 2), чей интерфейс в общих чертах похож на редактор сущностей. Поле *owner* позволяет определить владельца каждой таблицы, флажок *Physical Only* позволяет добавлять на физическую модель объекты, не

отражаемые в логическую модель, а *Generate* указывает какие таблицы участвуют в генерации схемы, а какие – нет.

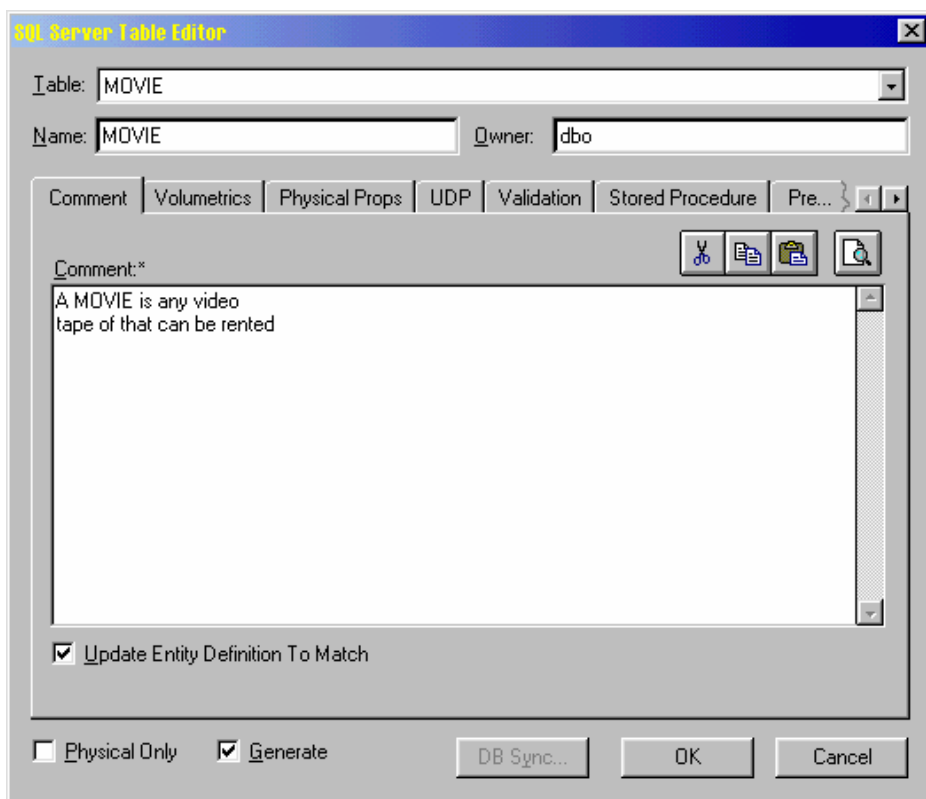


Рис. 2. Окно редактора таблиц

Страничка *Comment* позволяет определить комментарий таблицы (или почерпнуть его из сущности). С помощью вкладки *Volumetrics* имеется возможность указать критерии, используемые для оценки потребностей таблицы в памяти (начальный объем – *Initial Rows*, максимальный объем – *Max Rows*, прирост в месяц – *Grow By*). Вкладка *Physical Props* определяет особенности хранения таблицы в физической памяти БД (зависит от конкретной СУБД). Вкладка *Validation* определяет ограничения, накладываемые на таблицу целиком. С помощью вкладок *Stored Procedure* и *Pre&Post Script* определяются триггеры и процедуры, относящиеся к данной таблице.

Когда Вы добавляете атрибут к сущности в ERwin, имя атрибута автоматически используется в качестве имени колонки, присваиваемого по умолчанию соответствующей колонке в физической схеме. Другие характеристики автоматически присваиваются новым колонкам, исходя из значений, присваиваемых по умолчанию и заданных в редакторе *Target Server*. Например, если в редакторе *Target Server* установлен тип данных по умолчанию для новых колонок *CHAR(10)*, то новым колонкам при создании будет автоматически присвоен этот тип данных.

Колонке физической таблицы могут быть присвоены следующие характеристики: *Имя колонки*, *Тип данных*, *Режим нулевых значений*, *Правило валидации*, *Значение по умолчанию*,

*Правила миграции характеристик колонки.* ERwin позволяет использовать несколько различных редакторов для определения, модифицирования и присвоения характеристик колонкам физической таблицы. Значение каждой из характеристик, присваиваемое по умолчанию, можно изменить для каждой колонки в отдельности в редакторе *Column Editor* (рис. 3). Редакторы *Default* (рис. 4), *Validation Rule* (рис. 5) и *Valid Value* позволяют Вам задавать начальные значения, выражения сервера и фиксированные списки значений, которые присваиваются в качестве ограничений и ограничивают значения данных, которые могут быть записаны в конкретную колонку.

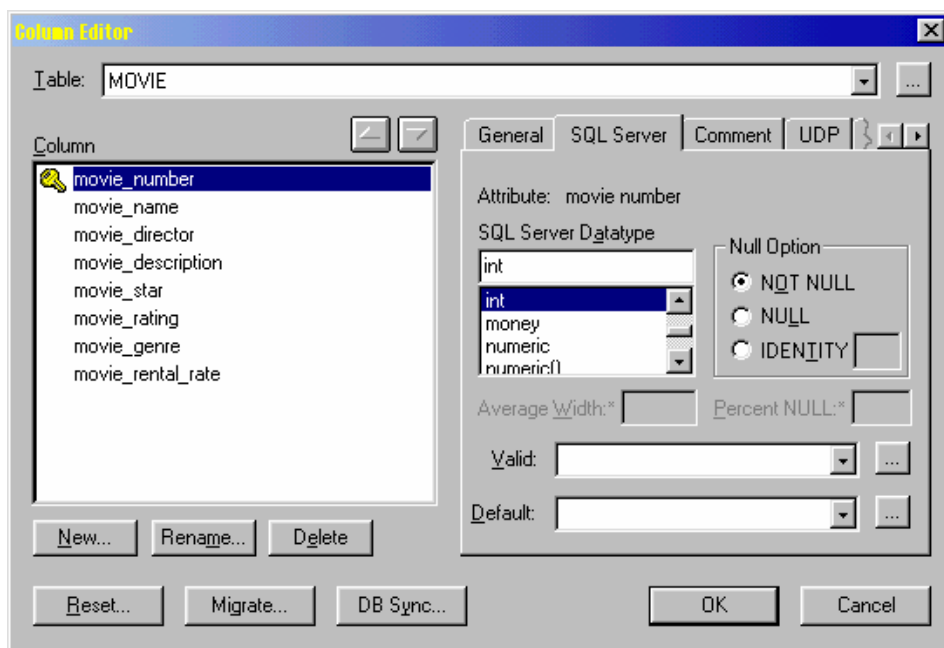


Рис. 3. Окно редактора колонок

Для ускорения процесса определения характеристик физических колонок редактор *Domain Dictionary* позволяет сгруппировать вместе несколько характеристик колонок и сохранить их в качестве *домена*. С помощью домена можно быстро присвоить набор характеристик данной колонке путем простого связывания имени колонки с именем домена. Если Вы используете домены для задания характеристик физических колонок, то, вместо того чтобы изменять характеристики каждой колонки в отдельности, Вы можете изменить характеристику один раз в определении домена, и это изменение автоматически распространяется на все колонки, использующие этот домен.

Редактор *Column Property* демонстрирует имя выбранной сущности и соответствующей физической таблицы в верхней части диалога. Прямо под именем сущности/таблицы находится список, содержащий информацию о каждой колонке выбранной таблицы, включая имя колонки, тип данных, имя атрибута и текущий режим нулевых значений.

Другие управляющие окна в редакторе *Column Editor* содержат текущие значения каждое - свои характеристики колонки. Чтобы просмотреть характеристики, связанные с конкретной колонкой, просто щелкните по колонке из списка в окне сверху и выделите ее.

Кнопка *Reset...* открывает окно-диалог, которое позволяет повторно установить значение одной или нескольких характеристик в соответствии со значениями, устанавливаемыми по умолчанию и определенными в домене.

Кнопка *DB Sync...* открывает окно-диалог *DB Sync* для синхронизации имени колонок, определенные в модели данных, с информацией, хранящейся на сервере СУБД.

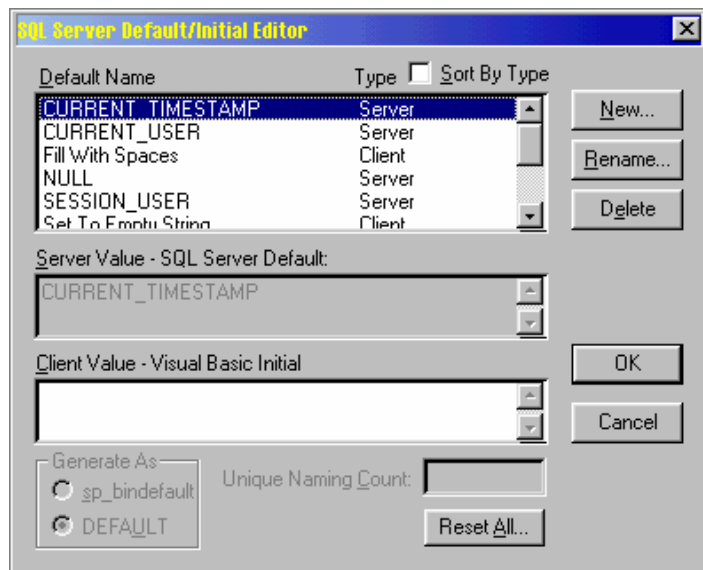


Рис. 4. Окно редактора значений по умолчанию

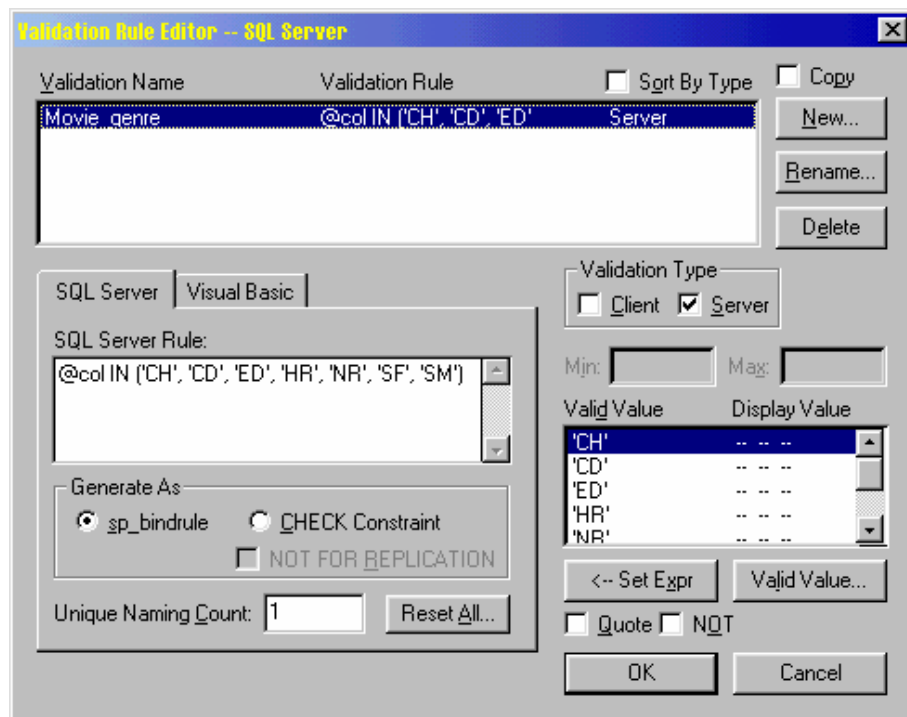


Рис. 5. Окно редактора правил проверки значений колонки

## Усовершенствование модели с помощью ограничений

ERwin поддерживает ввод (по желанию) правил валидации для колонок (в зависимости от выбранной СУБД), а также информации, присваиваемой колонкам по умолчанию.

*Значение по умолчанию* - значение, которое нужно ввести в колонку, если никакое другое значение не задано явным образом во время ввода данных. С каждой колонкой (атрибутом) можно связать значение по умолчанию. *Правило валидации (правило проверки значения)* задает список *допустимых значений* для конкретной колонки или использует выражение для задания какого-то вида кода для валидации данных.

Правила валидации и значения колонок по умолчанию создаются с использованием подходящих выражений языка определения данных для СУБД, либо активным образом через подсоединение к системному каталогу, либо через скрипт файла ASCII. ERwin не производит проверки на непротиворечивость данных в доменах и значений, которые Вы присваиваете колонкам по умолчанию.

Редактор *<DB> Default* (рис. 4) позволяет Вам создать значение, которое автоматически, по умолчанию, присваивается колонке. Во многих случаях наиболее часто встречающееся в колонке значение присваивается этой колонке в качестве значения по умолчанию. В верхней части редактора находится список, содержащий все существующие значения по умолчанию. Чтобы создать новое значение по умолчанию, введите имя в текстовое окно *Default Name* и затем нажмите кнопку *New*. Чтобы изменить существующее значение по умолчанию, выберите из списка то значение, которое Вы хотите изменить, измените значение в текстовом окне *Default Value* и нажмите кнопку *Update*. Чтобы удалить значение по умолчанию, выберите из списка значение и нажмите кнопку *Delete*.

Создав значение по умолчанию, Вы можете затем присвоить его одной или более колонкам с помощью редактора *Column Editor*. Для этого выберите ту колонку, к которой Вы хотите присоединить значение по умолчанию, и выберите значение из списка «Default».

## Работа в редакторе Validation Rule

Редактор *Validation Rule* (рис. 5) позволяет Вам создать ограничение при проверке, которое называется *правилом валидации*. Правило валидации - это выражение, которое задает диапазон допустимых значений, которые можно хранить в колонке, и определяет, является ли вводимое значение допустимым.

Если Вы хотите ввести выражение в синтаксисе СУБД, щелкните по текстовому окну *Server <DB> Rule* и введите выражение, соблюдая синтаксис выбранной СУБД. Если Вы хотите создать множество допустимых значений, нажмите кнопку *Valid Value...* для входа в

редактор *Valid Value* и создайте список допустимых значений для правила валидации. После того как Вы нажмете *OK*, Вы вернетесь в редактор *Validation Rule*. Если Вы нажмете кнопку *Server Expr*, то ERwin создаст выражение - команду языка СУБД, содержащую список допустимых значений, введенный Вами.

По умолчанию ERwin создает выражение - команду языка СУБД, используя значения, связанные с правилом валидации, и разделяя значения запятыми (например, C,D,M). В некоторых случаях правила синтаксиса данной СУБД требуют, чтобы каждое значение в команде заключалось в апострофы ('C', 'D', 'M'). Чтобы автоматически заключить каждое значение в апострофы, щелкните по флажку *Quote*. Если Вы хотите убрать апострофы, можно отредактировать выражение и удалить их в текстовом окне *Server <DB> Rule*.

После того как Вы создали правило валидации, Вы можете присвоить его одной или нескольким колонкам в редакторе *Column Editor*. Для этого выберите колонку, с которой Вы хотите связать правило валидации, и выберите правило из списка *Valid*.

## Генерация физической базы данных

После завершения всех манипуляций по подготовке физической модели базы данных наступает этап генерации физической базы данных. Иногда этот процесс называют *прямым проектированием* базы данных. Процесс проектирования является наиболее ответственным, так как на нем выявляются многие ошибки и просчеты предыдущих этапов проектирования. В общем случае процесс генерации выглядит по-разному в зависимости от конкретной СУБД. ERwin предоставляет две возможности при генерации схемы базы данных:

- подсоединение ERwin непосредственно к системному каталогу базы данных и генерация схемы за один шаг.
- генерация скрипта ASCII DDL (на языке определения данных). Скрипт DDL должен быть выполнен на сервере для генерации схемы, и это должно быть отдельное действие.

В первом случае последовательность работ такова:

- 1) подготовка физической модели;
- 2) настройка типов данных;
- 3) подключение к серверу баз данных;
- 4) генерация схемы;
- 5) отключение от сервера баз данных.

Чтобы сгенерировать схему, Вы должны сначала определить и затем выполнить специальный тип отчета ERwin, который называется *schema generation report* (рис. 6.).

Определение этого отчета создается в редакторе *Schema Generation*. Для открытия этого редактора применяется пункт меню *Tasks* → *Forward Engineer/Schema generation*. Редактор *Schema Generation* позволяет выбрать те определения физических объектов ERwin, то есть таблицы, индексы, триггеры, хранимые процедуры и т.д., которые необходимо включить в генерируемую схему.

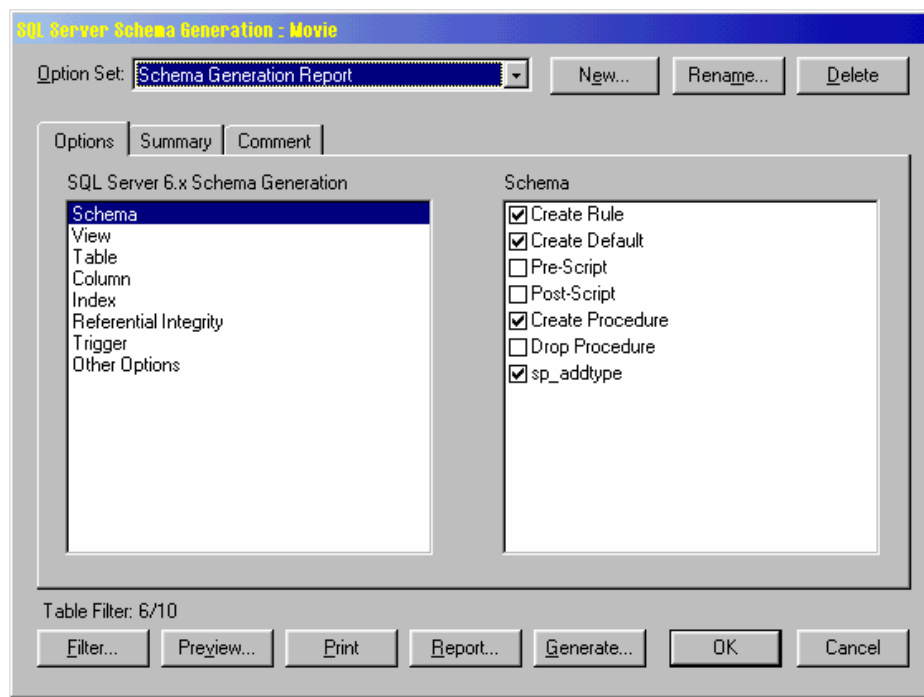


Рис. 6. Окно редактора состава генерируемой базы данных

Возможности, доступные в редакторе *Schema Generation*, различаются в зависимости от того, какие возможности поддерживаются СУБД. Тем не менее для всех СУБД Вы можете определить отчет *Schema Generation* в интерактивном режиме, сохранить его в файле отчета в формате ASCII, просмотреть отчет на экране и распечатать его непосредственно из диалогового окна.

Как и другие типы отчетов ERwin, отчет *Schema Generation* по умолчанию включает в себя все сущности текущей области. Если Вы хотите использовать только часть сущностей текущей области для генерации схемы, то выберите нужные Вам сущности, пользуясь для этого кнопкой *Filter...*

Кнопка *Generate...* в редакторе *Schema Generation* служит для запуска процесса генерации схемы. Когда Вы нажимаете кнопку *Generate...*, ERwin выводит на экран диалог *<DB> Connection*, который позволяет Вам подсоединиться к базе данных и связать ERwin с системным каталогом базы данных. После того как ERwin закончит генерацию схемы, он вернется в редактор *Schema Generation*. Если Вы не поставите метку в окне *Stop If Failure*, ERwin будет игнорировать ошибки и продолжит генерацию оставшейся части схемы, иначе он приостановит работу в случае ошибки. Нажмите кнопку *Continue*, чтобы продолжить



генерацию схемы. Нажмите кнопку *Abort*, чтобы отменить генерацию схемы. Если ERwin выдает сообщение об ошибке, просмотрите команды SQL, используя для этого окно *Preview*. Вы можете отредактировать коды в окне *Preview*, а затем нажать кнопку *Generate*, чтобы сгенерировать отредактированную версию.

## Интерфейс ODBC

Одним из способов генерации базы данных является использование интерфейса ODBC. ODBC – это специализированный интерфейс для доступа к базам данных из прикладных приложений. Его отличительной чертой является унифицированный интерфейс для доступа к любой поддерживаемой СУБД, что делает его очень удобным при генерации баз данных в ERwin. При использовании этого интерфейса процесс генерации базы данных выглядит так:

- 1) выбор ODBC как СУБД назначения;
- 2) создание БД-прототипа – пустой базы данных в целевой СУБД, в которую будет производится добавление объектов физической модели (например, для СУБД Access необходимо создать новую БД);
- 3) создание псевдонима ODBC для БД-прототипа, для этого необходимо:
  - запустить менеджер ODBC (*Пуск->Панель Управления->ODBC*);
  - добавить псевдоним – add (рис. 7.1);
  - выбрать драйвер СУБД (рис. 7.2);
  - настроить параметры псевдонима – обязательно указать его имя (рис. 7.3);
  - закрыть менеджер ODBC;
- 4) подключение по псевдониму (*Server→ODBC Connection, Connect*). Для подключения указать правильные имя пользователя и пароль;
- 5) генерация схемы БД;
- 6) отключение от базы данных (*Server→ODBC Connection, Disconnect*).

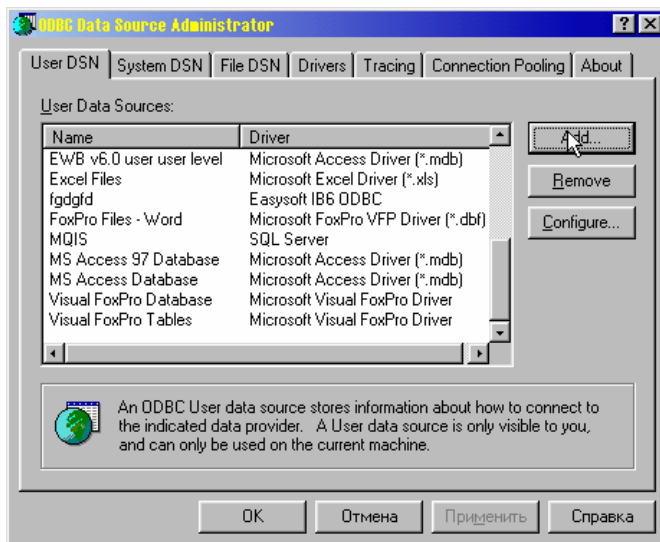


Рис 7.1. Интерфейс менеджера ODBC

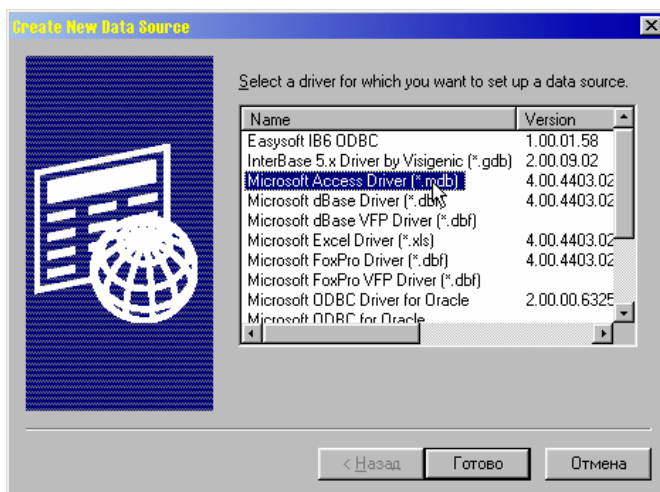


Рис 7.2. Выбор драйвера ODBC для псевдонима

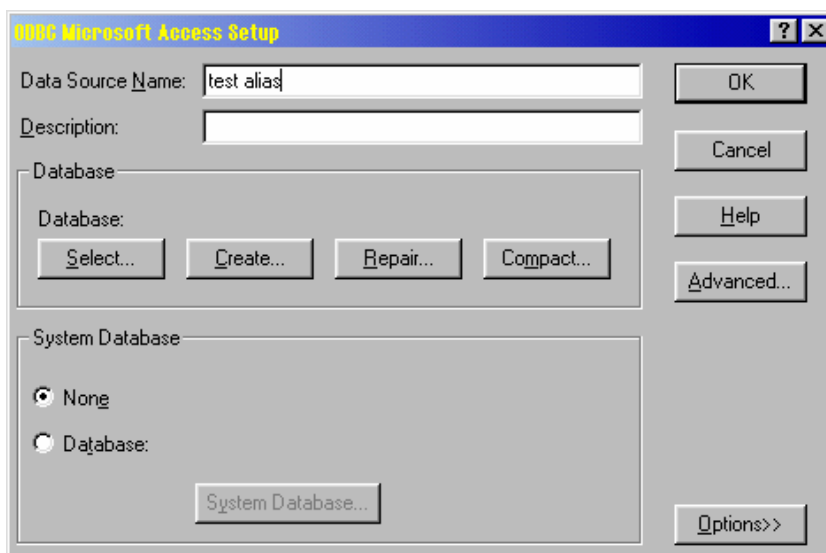


Рис. 7.3. Настройка параметров псевдонима для СУБД Access.

## Задание на лабораторную работу

1. Внимательно изучить приведенные материалы теоретического введения.
2. Сгенерировать физическую базу данных для задания из лабораторной работы №3 для СУБД Microsoft Access.
3. Разработать модель и построить физическую базу данных для решения следующей задачи. Пусть имеется транспортное агентство, занимающееся пассажирскими перевозками по железной дороге. Агентство имеет набор данных о маршрутах движения железнодорожных поездов, а именно список остановочных пунктов каждого маршрута и время движения между отдельными пунктами. Каждый маршрут расценивается как перечень пунктов, для каждого из которых известно время движения из предыдущего пункта и время стоянки. Каждый пункт маршрута характеризуется названием, и его типом (начальный, конечный или обычный). Агентство использует эти сведения для поиска оптимального маршрута проезда из города А в город Б. Оптимальным считается маршрут, который достаточно короток, максимально быстр и с минимальным количеством пересадок с одного поезда на другой. Пересадка может быть сделана в любом остановочном пункте маршрута. Требуется разработать базу данных, которая **позволяла бы** производить автоматизированный поиск оптимального маршрута.
4. Провести документирование полученной модели.
5. Сгенерировать физическую базу данных для полученной модели.
6. Сохранить итоговую модель в свою папку под именем **task4-1.er1**.

## Контрольные вопросы

1. В чем заключается сущность физического моделирования в ERwin, каковы его основные этапы?
2. Какие инструменты используются для настройки параметров колонки таблицы. Каковы различия между редакторами атрибутов и колонок?
3. Каким образом можно указать для атрибута “пол” список из двух возможных значений “муж” и “жен”?
4. Для чего используются флажки *Physical only* и *Generate* в редакторах физической модели?
5. Что такое ODBC, в чем его особенности и какие СУБД можно использовать с этим интерфейсом на Вашем компьютере?