

Лабораторная работа №8. Реинжиниринг и обратное проектирование баз данных

ЦЕЛЬ РАБОТЫ: Познакомиться с понятиями реинжиниринга баз данных и процессом их обратного проектирования, изучить процедуры обратного проектирования и синхронизации баз данных, разобраться с особенностями генерации схемы базы данных и применить полученные знания для обратного проектирования базы данных СУБД Microsoft Access.

Прямое и обратное проектирование

Процесс генерации физической схемы базы данных из логической модели данных называется *прямым проектированием (Forward Engineering)*. Когда Вы генерируете физическую схему, ERwin позволяет Вам включать триггеры ссылочной целостности, хранимые процедуры, индексы, ограничения и другие возможности, доступные при определении таблиц в СУБД. Аналогично, процесс генерации логической модели из физической базы данных называется *обратным проектированием (Reverse Engineering)*. ERwin позволяет быстро создать модель данных путем обратного проектирования имеющейся базы данных. После того как модель создана, Вы можете произвести обратное проектирование структуры базы данных, а затем легко перенести его в другой формат базы данных.

При подсоединении к базе данных ERwin создает активное соединение в двух направлениях с системным каталогом базы данных. Это соединение позволяет производить прямое и обратное проектирование схемы непосредственно в каталог базы данных. Аналогичным образом можно синхронизировать изменения, вносимые в модель ERwin, непосредственно с системным каталогом. При осуществлении синхронизации ERwin запрашивает системный каталог и сообщает о различиях, найденных между базой данных и ERwin. Независимо от того, переносится ли информация из ERwin в базу данных или наоборот, прямое соединение ERwin с системным каталогом базы данных позволяет легко синхронизировать модель с физической схемой на протяжении всего цикла разработки.

Обратное проектирование базы данных

ERwin может произвести обратное проектирование существующей базы данных SQL, считывая определения схемы и автоматически создавая соответствующую диаграмму модели данных. После того как диаграмма будет сгенерирована в результате обратного проектирования, Вы можете, используя инструменты и редакторы ERwin, добавлять новые

объекты, создавать системную документацию и перепроектировать структуру базы данных, основываясь на изменениях технических и (или) организационных требований.

При обратном проектировании необходимо иметь в виду, что разные СУБД на разных уровнях обеспечивают синтаксическую поддержку связей, индексов, ссылочной целостности и других свойств. Например, DB2 обеспечивает синтаксис, с помощью которого связи выражаются через внешние ключи и ссылочную целостность. Но не все СУБД, доступные из *ERwin*, поддерживают эти возможности.

ERwin позволяет произвести обратное проектирование базы данных:

- непосредственно из системного каталога базы данных;
- путем открытия и прочтения файла скрипта схемы SQL.

Независимо от того, какой из методов обратного проектирования используется, *ERwin* автоматически создает новое окно диаграммы Главной области и показывает на экране схему в виде графической модели данных.

Помимо импорта информации, явным образом определенной в физической схеме, *ERwin* извлекает значительный объем информации из схемы и встраивает ее в диаграмму в процессе создания новой модели данных путем обратного проектирования. Если синтаксис языка SQL СУБД поддерживает объявление внешних ключей, то *ERwin* автоматически извлекает из схемы идентифицирующие и неидентифицирующие связи и имена ролей по умолчанию, которые использует в генерируемой модели данных. В случаях DB2, SQL Server и Sybase *ERwin* может извлечь всю основную информацию по модели, за исключением связей подтипа, которые не поддерживаются в настоящее время ни одной из SQL-СУБД.

Если скрипт схемы на SQL содержит таблицы, в которых атрибуты первичного ключа не являются первыми колонками в таблице, то *ERwin* может произвести синтаксический анализ скрипта и сохранить верную информацию о порядке расположения колонок.

Синхронизация *ERwin* с базой данных

ERwin позволяет синхронизировать таблицы, колонки и объекты физической памяти в модели данных *ERwin* с теми, которые хранятся в базе данных. Когда Вы используете функцию синхронизации *ERwin* с базой данных, то *ERwin* проверяет информацию, заданную в модели данных, сравнивает ее с определениями, хранящимися в базе данных, и сообщает, что объекты в *ERwin* не синхронизированы с базой данных и наоборот.

Чтобы поддерживать синхронизацию *ERwin* и базы данных, нужно произвести обратное проектирование несинхронизированных таблиц и колонок (импорт) из базы данных в *ERwin* или прямое проектирование (экспорт) из модели данных *ERwin* в базу данных. Вы можете также переименовать объект в *ERwin* или в базе данных таким образом, чтобы имена и определения, используемые в обоих средах, совпадали.

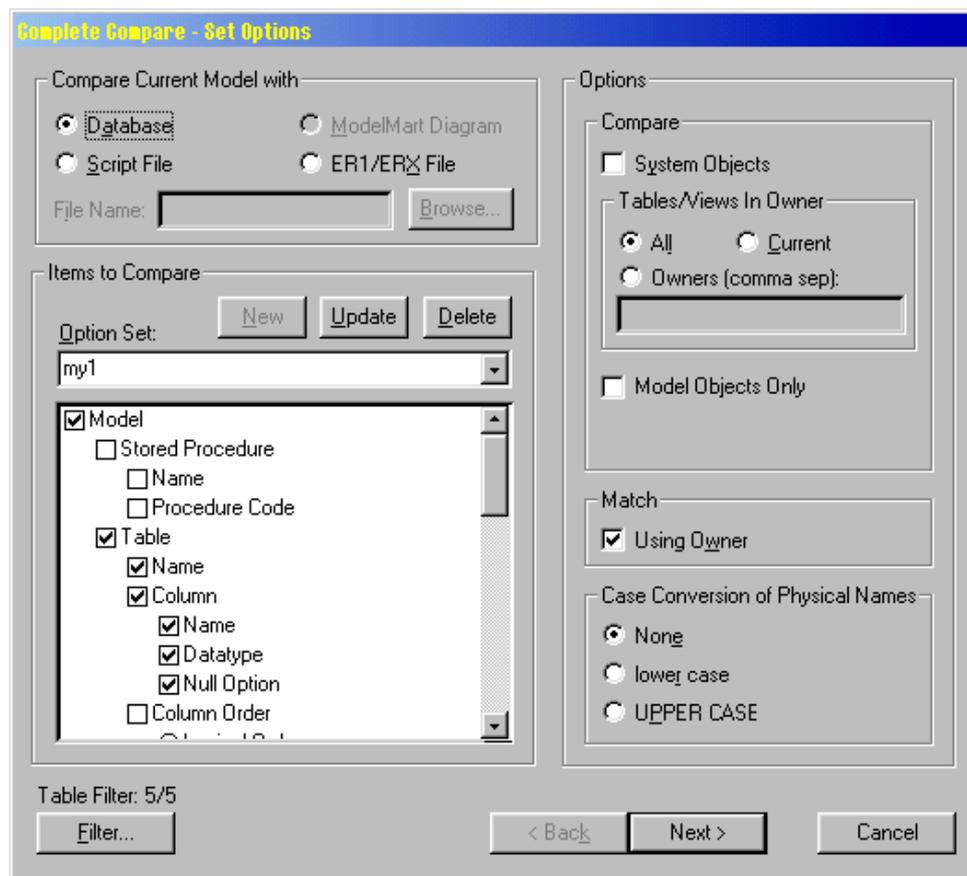
Для того чтобы сравнить информацию, хранящуюся на сервере, с информацией, определенной в ERwin, и импортировать (экспортировать) данные, ERwin должен создать активную связь с сервером базы данных. ERwin может подсоединиться к системному каталогу базы данных двумя способами:

- через собственный интерфейс, утилизировав функции поставщика для общения с этим каталогом;
- через промежуточную программу ODBC (Open Database Connectivity).

Процесс синхронизации модели и базы данных может протекать в одном из трех режимов:

- режим дополнения модели до схемы базы данных (*Tasks* → *Update model*);
- режим обновления базы данных до уровня модели (*Tasks* → *Alter database*);
- полная синхронизация (*Tasks* → *Complete compare*).

Первые два режима являются частными случаями синхронизации и рассматриваться не будут. Процесс полной синхронизации является многоступенчатым. Сначала определяются элементы модели подлежащие синхронизации, затем производится анализ различий и потом производится операция синхронизации.



На этапе отбора элементов модели (см. рис. 1) необходимо определить источник синхронизации (*Compare current model with*), синхронизируемые элементы модели (*items to compare*) и различные опции (*options*). Источником синхронизации могут служить

физическая БД (*database*), DDL скрипт на языке SQL (*script file*) или другая модель (*ER1/ERX file*). Синхронизируемые элементы модели представлены в виде упорядоченного дерева параметров. Набор параметров синхронизации может быть сохранен по некоторым именем (*option set*) и использован впоследствии. Кнопка *new* добавляет набор параметров, *update* обновляет его, а *delete* – удаляет. С помощью кнопки *filter* можно выделить сузить круг таблиц, подлежащих синхронизации. Параметры синхронизации определяют круг объектов БД, подлежащих сравнению (*compare, match*) и способ сравнения имен модели ERwin и имен физической БД (*Case conversion of physical names*). По нажатии кнопки Next производится открытие системного каталога базы данных, извлечение информации о ее схеме и сравнение информации с имеющейся в модели. Результаты сравнения выводятся в соответствующее окно (см. рис. 2).

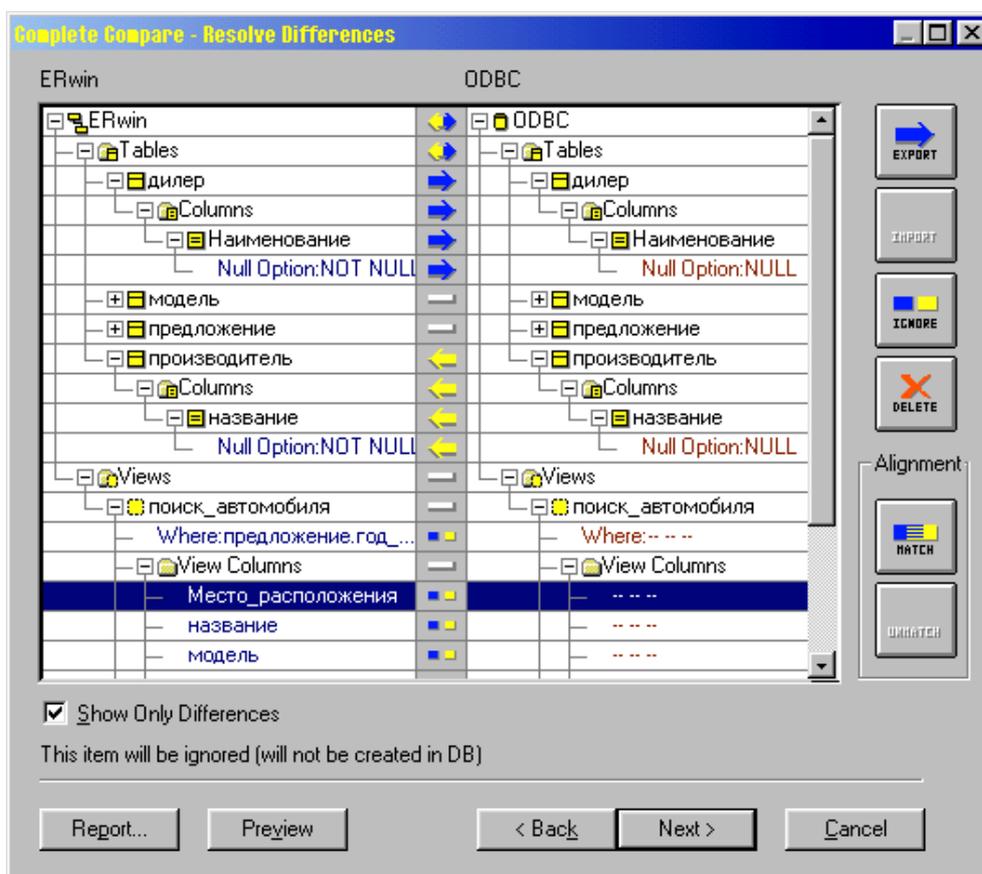


Рис. 2. Окно анализа различий схемы базы данных и модели ERwin.

В этом окне приводятся иерархически организованные структуры модели ERwin и схемы БД, где каждому элементу модели ставится в соответствие элемент БД. Если где-либо не было найдено соответствия, то вместо отсутствующей позиции выводится (-- -- --). Флажок *Show only difference* позволяет отфильтровать только имеющиеся отличия между моделью и схемой БД. Кнопка *report* позволяет построить краткий отчет о результатах сравнения, кнопка *preview* отображает примерный сценарий проведения синхронизации на языке SQL.

По каждому найденному факту отличия модели и БД Вы можете принять одно из пяти решений:



Синхронизировать элемент физической базы данных с моделью.



Синхронизировать отличный элемент модели с определением из базы данных



Игнорировать данное различие



Данный вариант возможен, если одна из сторон отсутствует. В этом случае можно удалить элемент который присутствует либо из модели, либо из базы данных.



Принудительно установить соответствие между элементами модели и базы данных. Используется только тогда, когда ERwin некорректно установил соответствие между моделью и базой данных.

В случае необходимости, на данном этапе можно приостановить операцию синхронизации по нажатию на кнопку *Cancel*, в противном случае нажатием кнопки *Next* начинается процесс синхронизации. Он может протекать по-разному в зависимости от выбора, сделанного на предыдущем этапе.

Если была выбрана операция импорта, то тогда сначала модель ERwin приводится в соответствие с БД и только затем БД приводится в соответствие с моделью. Настройка процесса импорта информации из БД производится в соответствующем редакторе (см. рис. 3). В этом окне Вы можете указать каким образом преобразовывать имена схемы БД в имена модели (*Case conversion of logical names*), как поступать с таблицами, уже имеющимися с модели (*If table to import exists in model*), какие действия осуществлять для поиска зависимостей между сущностями (*Infer*) и выделения представлений (*Views*). Если при импорте окажется, что какая-то таблица уже имеется в модели то можете либо перестроить непосредственно ее (*use existing table*), либо создать ее дубликат.

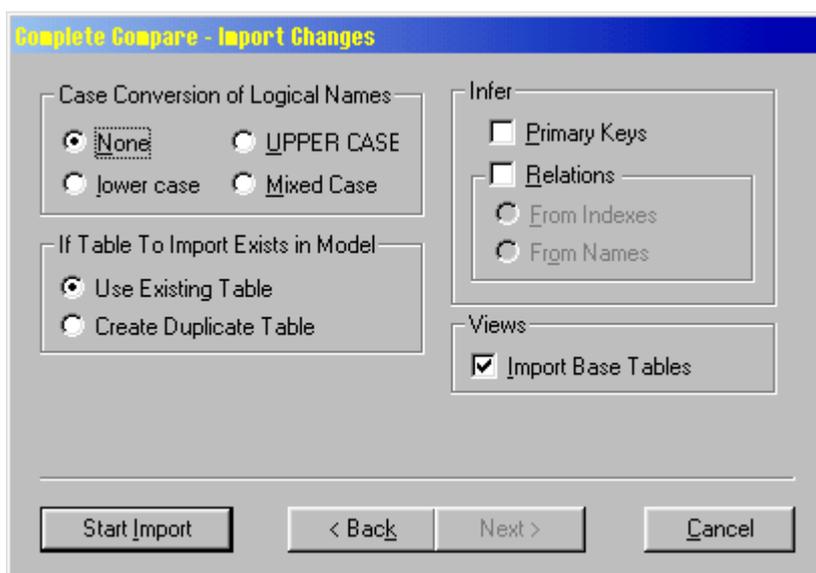


Рис. 3. Окно настройки параметров импорта схемы БД.

При импорте информации физической БД ERwin может автоматически восстанавливать часть информации физической модели, а именно определять первичные ключи сущностей (*primary key*), зависимости между ними (*relations*) и базовые таблицы представлений (*base tables*).

При необходимости обновления базы данных до уровня модели открывается окно настройка процесса экспорта (см. рис. 4). Центральное место занимает окно редактора, в котором приводится SQL скрипт, который осуществит синхронизацию. Его содержимым управляет целый ряд элементов. Флажки *Display warnings*, *Display changes*, *Display impact* управляют включением в текст скрипта пояснительной информации. Флажок *Stop if failure* позволяет остановить процесс синхронизации при возникновении ошибок. Кнопка *Gen Options* определяет структуру скрипта, осуществляющего генерацию новых элементов схемы БД (редактор генератора схемы). Кнопки *Reset*, *Abort*, *Start Export* контролируют процесс синхронизации. *Reset* отменяет операцию синхронизации, *Abort* прерывает ее, *Start Export* запускает синхронизацию

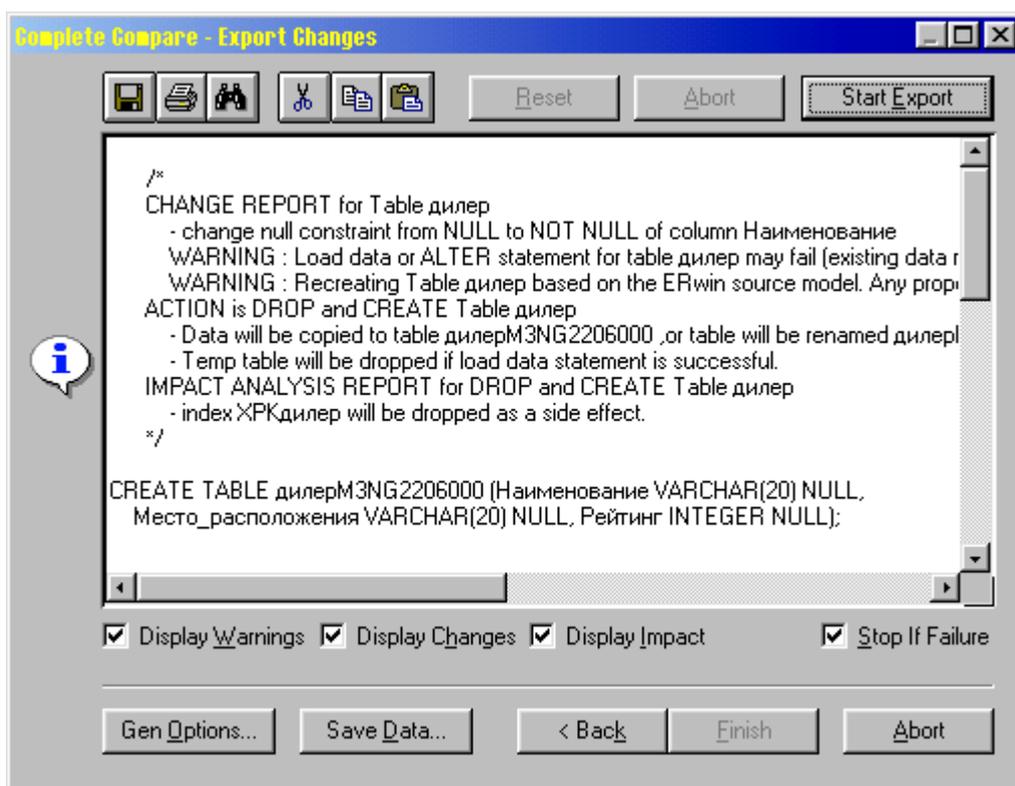


Рис. 4. Окно настроек процесса экспорта модели.

Ввиду того, что процесс экспорта модели в общем случае приводит к изменению таблиц, хранящих информацию, в процессе синхронизации ERwin предпринимает ряд мер по обеспечению сохранности данных в процессе экспорта. Особенности этого процесса управляет окно, открываемое по нажатию кнопки *Save Data* (см. рис. 5). Это окно определяет опции сохранения данных при экспорте. В верхнем списке перечисляются задействованные таблицы модели (*from table*) и БД (*to table*), условие сохранения имеющейся информации

(*where condition*), опции преобразования каждой задействованной колонки таблицы (*from, to, function*) и прочие параметры. С помощью флажка *DO NOT preserve data* вы можете указать, что ВСЯ информация, хранящаяся в таблице может быть удалена. Пользоваться данной возможностью необходимо с осторожностью, так как удаление таблицы может повлечь за собой каскадное выполнение операций, что может привести к потере данных. В процессе модификации структуры таблицы ERwin требует наличия временной таблицы для копирования данных. Флажок *DROP temp table* говорит, что эта таблица должна быть удалена после изменения таблицы (если его убрать, то появляется возможность отмены сделанных изменений). Флажок *FORCE drop/create temp table* позволяет удалять уже имеющиеся временные таблицы при проведении синхронизации. Для каждой колонки в процессе изменения таблицы может быть назначена функция приведения типов данных (используется для разрешения спорных моментов при перенесении данных из одной таблицы в другую). Функция преобразования пишется в соответствии с правилами языка SQL СУБД. Кнопка *Preview SQL* позволяет проследить конкретную операцию синхронизации в виде SQL скрипта.

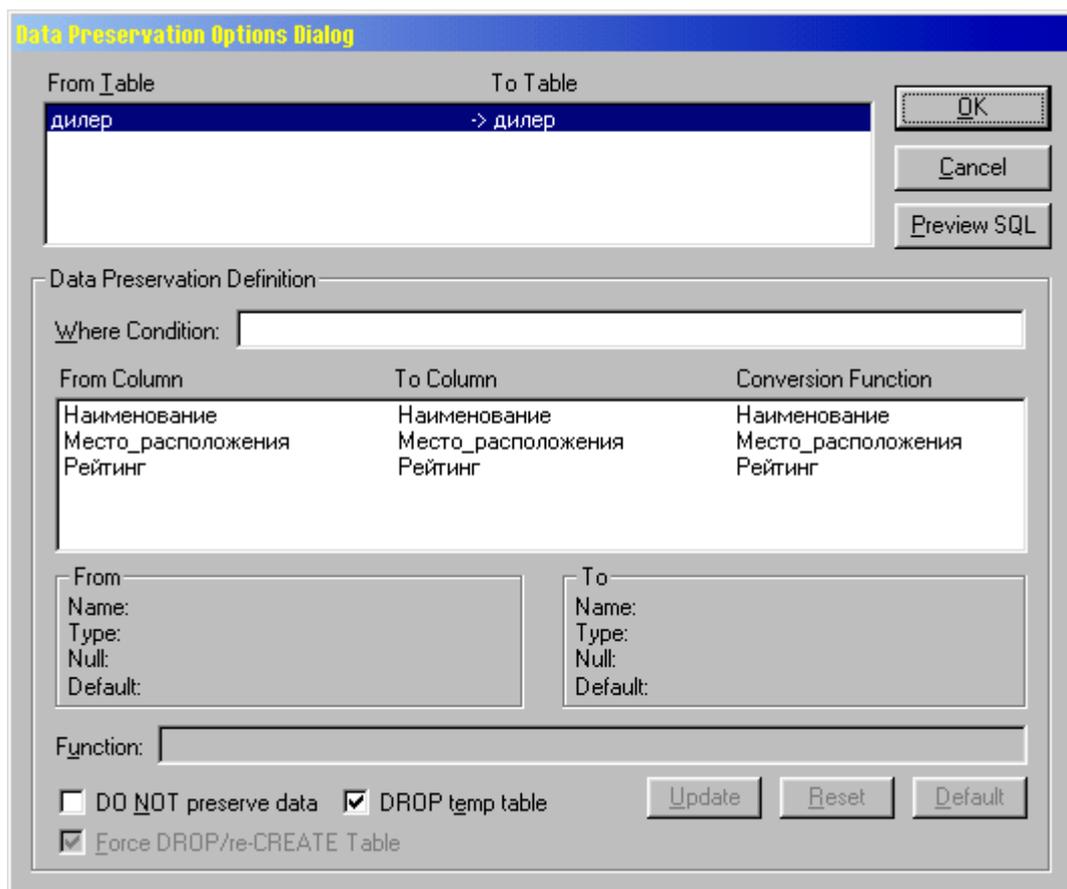


Рис. 5. Окно настроек процесса сохранения имеющихся данных в процессе экспорта модели.

После завершения процесса экспорта модели необходимо нажать кнопку *Finish*.

Режимы генерации схемы

При работе в редакторе *Schema Generation Report* Вам доступны различные режимы, в зависимости от СУБД. Поддерживаемые режимы находятся в соответствующей группе. Список групп опций: *Referential Integrity*, *Trigger Option*, *Statement Format*, *Table Option*, *Index Option*, *Column Option*, *Schema Option* и *Other Options*.

Referential Integrity

Режимы RI (ссылочной целостности) позволяют Вам указывать, как поступать со связанными записями, если значение в поле ключа изменяется или удаляется. Выберите один или несколько возможных режимов:

- Primary Key** для усиления уникальности определения каждой строки в таблице.
- Foreign Key** для усиления заданного правила ссылочной целостности в случае, когда значение во внешнем ключе изменяется.
- On Delete** для усиления заданного режима ссылочной целостности в случае, если значение удаляется в поле первичного или внешнего ключа.
- Unique (AK)** для усиления правила ссылочной целостности, требующего, чтобы значения альтернативных ключей были уникальными.
- sp_primary key** для включения системной процедуры, создающей первичный ключ в каждой таблице.
- sp_foreign key** для включения системной процедуры, создающей внешние ключи.

Trigger Option

Режимы триггера позволяют Вам переопределить шаблоны RI, устанавливаемые *ERwin* по умолчанию, с целью усиления ссылочной целостности. Выберите один или несколько возможных режимов:

- RI Type Override** для переопределения шаблона, устанавливаемого по умолчанию, для всех связей, которые были присвоены определенному типу правила ссылочной целостности.
- Relationship Override** для переопределения шаблона, устанавливаемого по умолчанию, для какой-то конкретной связи.
- Entity Override** для переопределения шаблона, устанавливаемого по умолчанию, для какой-то конкретной сущности.

Statement Format

Режимы формата команд позволяют Вам указать, будет ли схема содержать команду **CREATE** или **ALTER** для каждого первичного или внешнего ключа. Если таблица, которой

требуется новый ключ, уже есть в базе данных, используйте режим ALTER. Выберите один из возможных режимов, как для первичного, так и для внешнего ключа:

CREATE/PK для генерации команды CREATE PRIMARY KEY.

ALTER/PK для генерации команды ALTER PRIMARY KEY.

(по умолчанию)

CREATE/FK для генерации команды CREATE FOREIGN KEY.

ALTER/FK для генерации команды ALTER FOREIGN KEY.

(по умолчанию)

Table Option

Режимы для таблиц позволяют Вам указать, какие операторы языка определения данных будут использованы при создании схемы. Выберите один или несколько возможных режимов:

CREATE TABLE для выполнения операторов SQL CREATE TABLE в процессе генерации схемы.

Entity Integ для включения операторов SQL, создающих правила, которые накладывают ограничения, для каждой сущности.

DROP TABLE для выполнения операторов SQL DROP TABLE перед выполнением операторов CREATE TABLE при генерации схемы.

Physical Storage для включения в схему объектов и параметров физической памяти.

Table CHECK для включения операторов SQL, создающих правила, которые накладывают ограничения, для каждой сущности.

Table Pre-Script для включения в схему пре-скриптов (скриптов, выполняемых непосредственно перед генерацией схемы).

Table Post-Script для включения в схему пост-скриптов (скриптов, выполняемых непосредственно после генерации схемы).

Index Option

Режимы индексирования позволяют Вам указывать, каким образом будут создаваться и храниться индексы и какие из ключевых атрибутов будут индексированы. Выберите один или несколько возможных режимов:

Primary Key (PK) для создания индекса по первичному ключу в каждой сущности.

Alternate Key (AK) для создания индекса по альтернативным ключам в каждой сущности.

Foreign Key (FK) для создания индекса по внешним ключам в каждой сущности.

Inversion Entry (IE) для создания индекса по инверсионным ключам в каждой

сущности.

CLUSTERED or CLUSTERED HASHED для создания в схеме индекса CLUSTERED или CLUSTERED HASHED.

Physical Storage для включения в схему информации, относящейся к объектам физической памяти.

Column Option

Режимы для колонок позволяют добавлять ограничения в операторы SQL CREATE TABLE. Выберите один или несколько возможных режимов:

Attribute Integ для включения операторов SQL, содержащих ограничения, для каждого атрибута.

Column CHECK для включения операторов SQL, содержащих ограничения, для каждой колонки.

Default or DEFAULT Value для включения значения колонки по умолчанию в оператор схемы.

Physical Order для сохранения физического порядка расположения колонок при генерации новой схемы.

sp_bindrule для включения оператора, связывающего правило ограничения с колонкой.

sp_bindefault для включения оператора, связывающего с колонкой правило ограничения, устанавливаемое по умолчанию.

User Datatype для включения в оператор схемы типа данных, заданного пользователем для колонки.

Validation для включения в оператор схемы правила валидации для колонки.

Other Options

Другие доступные режимы поддерживают специальные возможности, предоставляемые выбранной СУБД. Выберите один или несколько возможных режимов:

Comments для включения определений сущности и атрибутов ERwin в базу данных в качестве комментариев.

Constraint Name для включения в схему имен ограничений.

Quote Names для заключения имен таблиц и колонок в кавычки.

Задание на лабораторную работу

1. Внимательно изучить материалы введения и освоить соответствующие редакторы ERwin.

2. С помощью менеджера ODBC создайте псевдоним под именем «исходная БД» для базы данных БОРЕЙ из состава примеров СУБД Access.
3. Осуществить обратное проектирование этой базы данных с использованием интерфейса ODBC и созданного псевдонима.
4. Изучить полученную модель и осуществить ее перепроектирование. Для этого измените все русские имена на английские, восстановите первичные ключи и связи между сущностями. Отразить произведенные изменения с помощью поля Notes соответствующих сущностей.
5. Осуществить прямое проектирование полученной модели в СУБД SQL Server (имя базы данных – task8-*<ваше имя>*, сервер – сервер ЛВС).
6. Сохранить полученную модель в свою под именем **task8-1.er1**.

Контрольные вопросы

1. Что такое реинжиниринг базы данных и в чем его отличие от прямого проектирования?
2. В чем разница между пунктами меню *Tasks* → *Alter Database*, *Tasks* → *Update Model* и *Tasks* → *Complete compare*? Для чего они используются?
3. Что необходимо сделать чтобы сохранить данные таблицы при обновлении базы данных?
4. Можно ли при проведении повторного проектирования таблицы «автомобиль» сохранить только те записи, у которых в графе «год выпуска» стоит число меньше 1960?
5. Каким образом при повторной генерации базы данных обновить только индексы таблиц и не трогать всей остальной информации базы данных? Возможно ли это?