

МОБИЛЬНАЯ  
РЕЛЯЦИОННАЯ  
СУБД **ЛИНТЕР**<sup>®</sup>

Linter Standard  
Linter Bastion  
Linter RealTime  
Linter Multiversion

## Конвертор модели данных

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ

---

 **РЕЛАКС**<sup>®</sup>

## **Товарные знаки**

РЕЛЭКС™, ЛИНТЕР® , НЕВОД® , LAV™, ЛАКУНА являются товарными знаками, принадлежащими ЗАО НПП «Реляционные экспертные системы» (далее по тексту – компания РЕЛЭКС). Прочие названия и обозначения продуктов являются товарными знаками их производителей, продавцов или разработчиков.

## **Интеллектуальная собственность**

Правообладателем продуктов ЛИНТЕР®, НЕВОД®, LAV™, ЛАКУНА является компания РЕЛЭКС (1990–2011). Все права защищены. Данный документ является собственностью компании РЕЛЭКС. Ни одна его часть не может быть воспроизведена, передана, преобразована, сохранена в системе поиска информации, переведена на другой язык или компьютерный язык в какой-либо форме, какими-либо средствами, электронными, механическими, магнитными, оптическими, химическими, ручными или иными, без предварительного разрешения компании РЕЛЭКС.

## **О документе**

Материал, содержащийся в данном документе, прошел тщательную проверку, но компания РЕЛЭКС не гарантирует, что документ не содержит ошибок и пропусков. Компания РЕЛЭКС оставляет за собой право в любое время вносить в документ исправления и изменения, пересматривать и обновлять содержащуюся в нем информацию.

## **Адрес**

394006, г. Воронеж, ул. 20-летия Октября, 119.

Тел./факс: (473) 2-711-711, 2-778-333.

e-mail: market@relex.ru.

## **Адрес для корреспонденции**

394000, г. Воронеж, а/я 137.

## **Техническая поддержка**

Отдел поддержки и сопровождения программных продуктов:

телефон: (473) 2-711-711 с 9:00 до 18:00 мск.

e-mail: support@relex.ru, market@relex.ru.

С целью повышения качества разрабатываемых программных средств и предоставляемых услуг в компании РЕЛЭКС действует автоматизированная система учёта и обработки рекламаций. Обо всех обнаруженных недостатках и ошибках в программном продукте и/или документации на него просим сообщать нам на Internet-странице [рекламация](#).

# Оглавление

<b>Предисловие</b> .....	<b>1</b>
Назначение документа .....	1
Для кого предназначен документ .....	1
Принятые обозначения и соглашения .....	1
<b>Общие сведения</b> .....	<b>3</b>
Возможности конвертора.....	4
<b>Условия выполнения</b> .....	<b>6</b>
<b>Конвертирование модели данных</b> .....	<b>7</b>
Командная строка.....	7
Конвертирование синонимов таблиц .....	7
Конвертирование таблиц.....	8
Конвертирование столбцов таблицы .....	9
Конвертирование ссылочной целостности.....	9
Конвертирование первичных и внешних ключей.....	10
Конвертирование индексов .....	11
Конвертирование имен объектов БД.....	12
Конвертирование типов данных .....	12
<b>Сообщения конвертора</b> .....	<b>14</b>



# Предисловие

## Назначение документа

Документ содержит описание утилиты ewc, выполняющей конвертирование модели данных, разработанной с помощью программного средства ERWIN для некоторой БД, в модель данных СУБД ЛИНТЕР.



Документ может использоваться для работы с любой версией СУБД ЛИНТЕР. Особенности конкретных версий оговариваются по тексту.

## Для кого предназначен документ

Документ предназначен для разработчиков и проектировщиков баз данных СУБД ЛИНТЕР.

## Принятые обозначения и соглашения

<u>Обозначение</u>	<u>Пример</u>	<u>Значение</u>
Курсив	<i>Растровым</i> называется изображение...	Новый термин в тексте.
Полужирный шрифт	В этом случае необходимо переносить <b>все</b> физические файлы.	Выделение в тексте.
Подчеркнутый шрифт	Подробную информацию о работе программы можно получить на сайте <a href="http://www.dmk.ru">www.dmk.ru</a> .	Адреса страниц Internet.
Текст, разделенный знаком ⇒	Выполните команду <b>View ⇒ Properties</b> (Вид ⇒ Свойства).	Последовательность выполнения команд.
Текст, заключенный в <>, со знаком + между ними	<Ctrl>+<C>	В <> заключаются клавиши клавиатуры, знак + означает сочетание клавиш.
Крупный моноширинный текст	SQL> _q	Текст командной строки.
Мелкий моноширинный текст	Page Time Count	Текст программы.
Заглавные буквы	BROWSE	Названия команд, слова, зарезервированные в SQL, ключевые слова.
Курсив в <>	<return statement>	Определяемый элемент синтаксической конструкции.
Символ ::=		Равенство по определению. Слева от знака стоит определяемое понятие, справа – собственно определение понятия.

<u>Обозначение</u>	<u>Пример</u>	<u>Значение</u>
Квадратные скобки [ ]	<code>DBSTORE [-d -n -o -p -r -t -u]</code>	Необязательные элементы конструкции. В данном примере ключи не являются обязательными элементами команды.
Вертикальная черта	<code>&lt;return value&gt; ::= &lt;value expression&gt;   NULL</code>	Указывает на то, что все предшествующие ей элементы списка являются необязательными и могут быть заменены любым другим элементом списка после этой черты.
Фигурные скобки { }	<code>CODEPAGE {866   1251   KOI8}</code>	Указывают на то, что все, находящееся внутри них, является единым целым.
Многоточие «...»	<code>Характеристики столбца MAKE CHAR(20) MODEL CHAR(20) ... SQL&gt;</code>	Означает, что предшествующая часть может быть повторена любое количество раз.
Многоточие, внутри которого находится запятая «,...»		Указывает на то, что предшествующая часть оператора, состоящая из нескольких элементов, разделенных запятыми, может иметь произвольное число повторений.
Текст со знаком  на сером фоне	<div data-bbox="566 1294 975 1489" style="background-color: #e0e0e0; padding: 5px;"> Если конфигурация страницы-шаблона не учитывала свойств, команда будет выполнена некорректно.</div>	Примечание.

# Общие сведения

Утилита ewc предназначена для конвертирования схемы БД, разработанной с помощью программного средства моделирования данных ERWIN/ERX в схему БД СУБД ЛИНТЕР. Результатом работы ewc является набор SQL-скриптов на диалекте СУБД ЛИНТЕР, начиная с версии 5.x.

Исходная (конвертируемая) схема БД должна быть предварительно создана с помощью средства разработки схем БД ERWIN версии 2.6 или 3.5 для любой БД, поддерживаемой ERWIN, и сохранена в формате .erx.

Чтобы более четко понять, для чего нужна утилита ewc, рассмотрим основные возможности ERWIN/ERX.

Средство моделирования данных ERWIN/ERX предназначено в основном для разработчиков, проектировщиков БД, системных аналитиков. Разработчики с помощью ERWIN/ERX могут сначала, используя визуальные средства, описать схему БД, а затем автоматически сгенерировать файлы данных для выбранной реляционной СУБД (прямое проектирование). Автоматически генерируются также триггеры, обеспечивающие ссылочную целостность БД. Поддерживаются хранимые процедуры. Пользователь описывает структуру данных визуально. Он задает служащие прообразами реляционных таблиц сущности с их атрибутами и при помощи «мыши» устанавливает между ними связи, которые являются прототипами реляционных отношений.

Возможна также обратная разработка. ERWIN/ERX позволяет по уже существующим файлам БД восстанавливать логическую структуру данных. Это называется *обратным проектированием*. Оно позволяет, во-первых, переносить структуру БД (но не данные!) из одной СУБД в другую и, во-вторых, исследовать старые проекты. Обратное проектирование наиболее распространено в процессе перехода с одной технологии на другую (с файл-сервера на клиент-сервер), а также при смене сервера БД.

ERWIN/ERX поддерживает прямое и обратное проектирование более **20** БД на основе прямого доступа к системному каталогу баз данных (отпадает потребность в использовании ODBC-драйвера).

В ERWIN/ERX встроена поддержка следующих СУБД:

- Oracle;
- Sybase;
- Informix;
- CA Ingres;
- DB2, Rdb;
- Watcom;
- Centura SQLBase;
- Microsoft SQL Server;
- AS/400;
- Progress;
- FoxPro;
- InterBase;
- dBASE;
- Clipper;
- Paradox;
- Access и др.

Как видно, в списке поддерживаемых ERWIN БД ЛИНТЕР не значится, поэтому использовать ERWIN напрямую для моделирования данных СУБД ЛИНТЕР нельзя. Однако

с помощью утилиты-конвертора ewc можно воспользоваться всей мощью средства моделирования ERWIN.

Для этого необходимо:

1. Описать модель данных для СУБД ЛИНТЕР с помощью ERWIN на основе любой СУБД, поддерживаемой ERWIN (желательно, чтобы SQL-язык используемой для моделирования БД был максимально приближен к диалекту SQL-языка СУБД ЛИНТЕР). Результатом будет выходной ERX-файл системы ERWIN, содержащий описание модели данных СУБД ЛИНТЕР в SQL-терминах исходной БД.
2. Пропустить полученную модель данных СУБД ЛИНТЕР в виде ERX-файла через утилиту-конвертор ewc. Результатом будет описание модели данных в виде SQL-скриптов СУБД ЛИНТЕР.
3. При необходимости отредактировать вручную полученные SQL-скрипты с учетом особенностей диалекта языка SQL СУБД ЛИНТЕР.
4. Выполнить SQL-скрипты. Результатом будет набор объектов БД ЛИНТЕР, реализующих созданную с помощью ERWIN модель данных.

## Возможности конвертора

Конвертор позволяет генерировать следующие SQL-конструкции:

1) создание таблиц.

В зависимости от параметров запуска конвертора создание таблиц подразумевает генерацию:

- предложений удаления таблиц;
- предложений добавления в таблицы первичного ключа;
- предложений присвоения столбцу таблицы атрибута **UNIQUE**;
- предложений создания индексов;
- предложений создания синонимов.

2) модификацию параметров таблиц.

Модификация таблиц подразумевает генерацию предложений добавления ссылочных ключей.

3) имена полей (физические, логические, роли).

ERWIN позволяет назначать полям таблиц имена трех типов:

- логическое имя (имя, присваиваемое полю самой программой ERWIN при описании логической схемы БД);
- физическое имя (имя, задаваемое при описании поля);
- роль (дополнительное имя, задаваемое для уточнения роли данного поля в схеме БД. Используется для различения одноименных полей в разных объектах БД).

Если обозначено только логическое имя, то поле создаваемой таблицы будет задано именем, совпадающим с логическим, с символом '\_' вместо пробелов.

Если обозначено физическое имя, то поле создаваемой таблицы будет задано физическим именем.

Если для логического имени заданы роли, в таблице будет создано столько полей, сколько ролей имеет логическое имя.

Если для одной или нескольких таблиц заданы триггеры, конвертор помещает текст хранимых процедур в файл с именем исходного файла и расширением .src.

#### 4) поддерживаемые типы БД.

Различные СУБД поддерживают неодинаковые названия типов данных. Поэтому конвертор предоставляет два варианта установления соответствия типов данных между типами данных той СУБД, для которой разрабатывалась схема и СУБД ЛИНТЕР.

Первый вариант заключается в создании таблицы соответствия типов данных этой СУБД и СУБД ЛИНТЕР. При таком способе неважно, для какой СУБД велась разработка схемы, главное, чтобы в таблице соответствий типов нашелся соответствующий ему тип СУБД ЛИНТЕР. Если таблица соответствий типов не задана, или данный конкретный тип отсутствует в ней, предлагается второй вариант.

Второй вариант – это стандартные преобразования типов данных СУБД к типам данных СУБД ЛИНТЕР. Для удобства разработчика конвертор поддерживает формат данных нескольких типов серверов, для которых создается схема: DB/2, NetWare SQL Server, Oracle, Watcom SQL Anywhere (см. таблицу 1).

Если схема создана для сервера, не входящего в этот список, и нет таблицы соответствий типов либо такой тип отсутствует в данной таблице, то при трансляции будет выдана ошибка **Неподдерживаемый тип БД**. Если возникла такая ошибка, необходимо пересоздать схему для любого сервера из списка, приведенного выше, либо использовать таблицу соответствия типов.

Если таблица содержит поле типа, неподдерживаемое СУБД ЛИНТЕР, его тип остается пустым.

**Таблица 1. Соответствие типов данных СУБД ЛИНТЕР различным серверам**


ЛИНТЕР	DB/2	NetWare SQL	Oracle	Watcom SQL Anywhere
BLOB	GRAPHIC			
BYTE()		BIT()		BINARY()
CHAR()	CHAR(), CHARACTER(), VARCHAR()	CHARACTER(), VARCHAR()	CHAR(), CHARACTER(), VARCHAR()	CHAR(), VARCHAR()
DATE	DATE	DATE	DATE	DATE
DECIMAL	DECIMAL	DECIMAL	DECIMAL	DECIMAL
DOUBLE				DOUBLE
INT		INT		INT
INTEGER	INTEGER		INTEGER, LONG	INTEGER
INT, AUTOINC		AUTOINC(4)	ROWID	
NUMERIC		NUMERIC	NUMBER	NUMERIC
REAL	FLOAT, REAL	BFLOAT, FLOAT	FLOAT	FLOAT, REAL
SMALLINT	SMALLINT	LOGICAL	SMALLINT	SMALLINT

## Условия выполнения

Для работы конвертора EWC необходимо:

- 1) активное ядро СУБД ЛИНТЕР версии 5.7 и выше;
- 2) минимальные права доступа к БД (т.е. Resource) пользователя, работающего с EWC;
- 3) отсутствие в БД таблиц с именами:
  - <имя владельца>.ERW\_TEXT;
  - <имя владельца>.ERWIN\_ENTITY;
  - <имя владельца>.ERWIN\_RELATIONSHIP;
  - <имя владельца>.ERWIN\_EAU;
  - <имя владельца>.INDEX\_MEMBER;
  - <имя владельца>.DOMAIN;
  - <имя владельца>.DOMAIN\_PROP\_VALUE;
  - <имя владельца>.ENT\_PHYSICAL\_PV;
  - <имя владельца>.ERWIN\_DEFAULT;
  - <имя владельца>.ENTITY\_SYNONYM,

где <имя владельца> - имя пользователя, под которым в данный момент выполняется конвертор EWC.

 Указанные выше имена конвертор еwc использует для именования своих рабочих таблиц, поэтому при запуске ERWIN таблицы с этими именами будут предварительно удалены из БД, а вместо них созданы новые.

# Конвертирование модели данных

Для запуска конвертора необходимо в интерпретаторе команд операционной системы подать командную строку:

```
ewc <командная строка>
```

## Командная строка

### Синтаксис

```
ewc -u <имя пользователя> / <пароль>
  [ -n <имя сервера> ]
  [ <правила конвертирования синонимов таблиц> ]
  [ <правила конвертирования таблиц> ]
  [ <правила конвертирования столбцов таблиц> ]
  [ <правила конвертирования ссылочной целостности> ]
  [ <правила конвертирования первичных и внешних ключей> ]
  [ <правила конвертирования индексов> ]
  [ <правила конвертирования типов данных> ]
  [ <правила конвертирования имен объектов БД> ]
  [ <язык интерфейса> ]
  [ <справочная информация> ]
  [ <информация о версии> ]
  <входной файл>
  [ <выходной файл> ]
```

<язык интерфейса> ::= -si <кодовая страница>

<справочная информация> ::= - { h | ? }

<информация о версии> ::= -v

<входной файл> ::= спецификация файла формата .egx

<выходной файл> ::= спецификация выходного файла

## Конвертирование синонимов таблиц

### Синтаксис

```
<правила конвертирования синонимов таблиц> ::=
  -A [<тип синонима>] [<владелец>]
  <тип синонима> ::= y | n
  y – создавать PRIVATE-синоним
  n – создавать PUBLIC-синоним
  <владелец> ::= идентификатор владельца
```

### Умолчания

- -A соответствует -Ay (создаются PRIVATE-синонимы для всех владельцев);
- если <владелец> указан, то <тип синонима> должен быть явно задан.

### Примеры

-A /\* Создавать PRIVATE-синоним для всех владельцев объектов БД \*/  
-An /\* Создавать PUBLIC-синоним для всех владельцев объектов БД \*/  
-Au /\* Создавать PRIVATE-синоним для всех владельцев объектов БД \*/  
-An"Петров" /\* Создавать PUBLIC-синоним только для тех объектов БД, владельцем которых является Петров \*/  
-Auau /\* Создавать PRIVATE-синоним только для тех объектов БД, владельцем которых является au \*/

### Общие правила

1) SQL-предложения создания синонимов генерируются только в том случае, если разрешено создание таблиц (опция -Т) и задана опция -А.

2) Генерируемое SQL- предложение имеет следующий вид:

```
CREATE [PUBLIC] SYNONYM <имя синонима>  
FOR TABLE [<владелец таблицы> . ] <имя таблицы> ;
```

В качестве <имени синонима> используется соответствующее имя из входного файла .egx.

## Конвертирование таблиц

### Синтаксис

<правило конвертирования таблицы> ::=

-Т { <создавать таблицу?> } { <удалять таблицу?> }

<создавать таблицу?> ::= у | n

<удалять таблицу?> ::= у | n

### Умолчания

- -Т соответствует -Туп (включать конструкции CREATE TABLE... и не включать DROP TABLE...);
- если опция <правило конвертирования таблиц> не задана, по умолчанию принимается -Т (т.е. SQL-предложения создания таблиц генерируются всегда, если только это явно не отменено).

### Примеры

-Т  
-Туу  
-Тnn  
-Тпу

### Общие правила

1) Генерируемое SQL-предложение в общем случае имеет следующий вид:

```
CREATE TABLE <имя таблицы> (<имя столбца> <тип данных столбца>, ...).
```

2) Табличные параметры (MAXROWID, MAXROW, PCTFILL и др.) и файловые (INDEXFILES, DATAFILES и др.) не генерируются. При необходимости эти параметры могут быть добавлены в SQL-предложения вручную (путем редактирования выходного файла).

3) Если заданы опции конвертирования столбцов ссылочной целостности (см. ниже), то в SQL-предложение добавляются соответствующие SQL-конструкции.

## Конвертирование столбцов таблицы

### Синтаксис

```
<правила конвертирования столбцов таблицы> ::=
  -С {<использовать значения по умолчанию ?>} {<физический порядок столбцов ?>}
    <использовать значения по умолчанию ?> ::= у | n
    <физический порядок столбцов ?> ::= у | n
```

### Умолчания

- -С соответствует -Суу (использовать значения по умолчанию и располагать столбцы в таблице в физическом порядке);
- если опция <правила конвертирования столбцов таблицы> не задана, по умолчанию принимается -С.

### Общие правила

- 1) Для отработки опции -С необходимо, чтобы было разрешено создание таблиц.
- 2) Если задана опция -Су (использовать значения по умолчанию), то в генерируемое SQL-предложение создания таблицы для всех столбцов, имеющих значения по умолчанию, добавляется конструкция DEFAULT <значение>, в противном случае присвоенные значения по умолчанию игнорируются.
- 3) Если задана опция -С {у|n} n (не использовать физический порядок столбцов), то столбцы располагаются в соответствии с присвоенными им ERWIN порядковыми номерами.

### Примеры

```
-Сn      /* Не использовать значения по умолчанию, столбцы размещать в физическом порядке */
-Суп     /* Использовать значения по умолчанию и располагать столбцы в логическом порядке */
```

## Конвертирование ссылочной целостности

### Синтаксис

```
<правила конвертирования ссылочной
целостности> : : =
  -R {<создавать первичные ключи ?> }
    {<создавать внешние ключи ?> }
    {<создавать альтернативные ключи ?> }
    <создавать первичные ключи ?> : : = у | n
    <создавать внешние ключи ?> : : = у | n
    <создавать альтернативные ключи ?> : : = у | n
```

### Умолчания

- -R соответствует -Рууп (создавать только первичные и внешние ключи);

## Конвертирование модели данных

---

- если опция <правила конвертирования ссылочной целостности> не задана, по умолчанию принимается -R.

### Общие правила

- 1) Данная опция определяет только необходимость поддержки ссылочной целостности с помощью набора различных ключей. Механизм создания ключей (т.е. в какой конкретно SQL-оператор (CREATE TABLE... или ALTER TABLE...) должны генерироваться соответствующие конструкции) задается опцией -S <правила конвертирования первичных и внешних ключей> (см. ниже), поэтому для отработки опции -R необходимо согласование параметров опции -S, чтобы было разрешено создание или модификация таблиц (см. опцию -T на стр. 8).
- 2) Для создания внешних ключей необходимо, чтобы было разрешено создание первичных ключей.
- 3) Под первичными ключами подразумеваются как простые, так и составные первичные ключи.

### Примеры

- Rnnn /\* Не создавать никаких ключей \*/
- Rnyn /\* Создавать только внешние ключи \*/

## Конвертирование первичных и внешних ключей

### Синтаксис

<правила конвертирования первичных и внешних ключей> : :=

-S {PC | PA} {FC | FA}

PC – выполнять создание первичного ключа в операторе CREATE TABLE

PA – выполнять создание первичного ключа в операторе ALTER TABLE

FC – выполнять создание внешнего ключа в операторе CREATE TABLE

 В текущей версии программы параметр не реализован.

FA - выполнять создание внешнего ключа в операторе ALTER TABLE

### Умолчания


- -S соответствует -SPCFA (включать конструкцию создания первичных ключей в оператор CREATE TABLE..., а внешних – в оператор ALTER TABLE...);
- если опция <правила конвертирования ссылочной целостности> не задана, по умолчанию принимается -S.

### Общие правила

- 1) Если задано создание первичных ключей, то в генерируемое SQL-предложение CREATE TABLE... или ALTER TABLE... добавляется конструкция PRIMARY KEY <имя столбца> для столбца, являющегося простым первичным ключом или PRIMARY KEY (<имя столбца>, <имя столбца>[,...]) – для составного первичного ключа.
- 2) Если задано создание внешних ключей, то в генерируемое SQL-предложение ALTER TABLE... добавляется конструкция:  
FOREIGN KEY (<имя столбца>)  
{ REFERENCES <ссылка на таблицу> (<ссылка на столбец>)

```
[ ON UPDATE
{ CASCADE
| SET NULL
| SET DEFAULT
| { NO ACTION | RESTRICT} } ]
[ ON DELETE
{ CASCADE
| SET NULL
| SET DEFAULT
| { NO ACTION | RESTRICT } } ]
```

3) Если задано создание альтернативных ключей, то в генерируемое SQL-предложение CREATE TABLE... добавляется конструкция UNIQUE для соответствующего столбца.

 Столбец с атрибутом UNIQUE допускает, в отличие от PRIMARY KEY, NULL-значения.

## Конвертирование индексов

### Синтаксис

```
<правила конвертирования индексов> ::=
-I {<создавать индексы для первичного ключа ?>
{<создавать индексы для внешнего ключа ?>
{<создавать индексы для альтернативного ключа ?> }
<создавать индексы для первичного ключа ?> ::= y | n
<создавать индексы для внешнего ключа ?> ::= y | n
<создавать индексы для альтернативного ключа ?> ::= y | n
```

### Умолчания

- -I соответствует -Iууп (создавать индексы только для первичных и внешних ключей);
- если опция <правила конвертирования индексов> не задана, по умолчанию принимается -I.

### Общие правила

- 1) Данная опция распространяется на простые и составные индексы.
- 2) Для простого индекса генерируется конструкция:

```
CREATE INDEX <имя столбца> ON <имя таблицы>;
```

Для составного индекса генерируется конструкция:

```
CREATE INDEX <имя индекса> ON <имя таблицы> <список столбцов>;
```

В обоих случаях файловый параметр INDEXFILE <номер файла> не генерируется. Для его указания необходимо вручную откорректировать выходной файл.

### Примеры

```
-Innn /* Не создавать никаких индексов */
-Inupn /* Создавать индексы только для внешних ключей */
```

# Конвертирование имен объектов БД

## Синтаксис

<правила конвертирования имен объектов БД> ::=  
-O {<обрамлять кавычками ?>}  
<обрамлять кавычками ?> ::= у | п

## Умолчания

- -O соответствует -Op (кавычки не использовать);
- если опция <правила конвертирования имен объектов БД> не задана, по умолчанию принимается -O.

## Общие правила

В соответствии с реализацией языка SQL в СУБД ЛИНТЕР все имена объектов БД приводятся к верхнему регистру латинского алфавита. Чтобы имя объекта не преобразовывалось, необходимо заключить его в двойные кавычки.

## Примеры

Имя объекта	Опция	Значение в выходном файле
TEST	-O	TEST
Test	-O	TEST
Test	-Oy	"Test"
Банк	-O	<FNR
Банк	-Oy	"Банк"

# Конвертирование типов данных

## Синтаксис

<правило конвертирования типов данных> : : =  
-B <спецификация таблицы соответствий типов>

Таблица соответствия типов данных задает соответствие типов данных, поддерживаемых ERWIN и СУБД ЛИНТЕР. Она должна быть оформлена в виде текстового файла со строками следующей структуры:

<исходный тип данных> # <целевой тип данных>

<исходный тип данных> : : = <разделитель><тип данных ErWin><разделитель>

<целевой тип данных> : : = <разделитель><тип данных СУБД ЛИНТЕР><разделитель>

<разделитель> : : = { символ пробела | символ табуляции } [<разделитель>]


<тип данных ErWin> – имя типа данных, поддерживаемое конвертором ErWin.

<тип данных СУБД ЛИНТЕР> – имя типа данных, поддерживаемое СУБД ЛИНТЕР и соответствующее <типу данных ErWin>.

## Синтаксические правила для таблицы

- 1) Длина строки не должна превышать 512 символов.
- 2) Имя может состоять из нескольких слов.
- 3) Слова в написании имени можно разделять любым количеством пробелов.


- 4) Слова в написании имени могут быть набраны на любом регистре.
- 5) Длина имени типа не должна превосходить 80 символов.
- 6) Если строка содержит символ «!» в любой позиции, то она считается комментарием и пропускается.
- 7) Пустые строки пропускаются.
- 8) Всего в таблице должно быть не более 64 описаний типов.

 Если в таблице соответствия задано более 64 строк, то конвертор ewc выдаст предупреждение **файл соответствий типов содержит более 64 описаний типов** и использует только первые 64 строки, остальные будут проигнорированы.

### Примеры строк таблиц

```

:~ это комментарий
и это тоже комментарий !
и это ! тоже комментарий !!!
LONG VARCHAR          #      BLOB
LONG BINARY           #      BLOB
VARCHAR ()            #      CHAR
VARCHAR                #      CHAR
```

 Типы данных <Имя типа>() и <Имя типа> (например, VARCHAR() и VARCHAR) считаются разными. После имени типа СУБД ЛИНТЕР скобки ставить нельзя. Будет считаться, что такого типа нет.

# Сообщения конвертора

<u>Текст сообщения</u>	<u>Причина</u>	<u>Примечание</u>
Ошибка открытия входного файла	Входной файл отсутствует, или ошибка файловой системы.	
Ошибка создания выходного файла	Невозможно создать выходной файл.	
Внутренняя ошибка	Ошибка во внутренних структурах конвертора.	Обратитесь к разработчикам.
Недостаточно памяти	Недостаточно ресурсов для выделения требуемого количества памяти.	
Ошибка чтения входного файла	Ошибка файловой системы.	
Ошибка записи выходного файла	Нет места на диске, или ошибка файловой системы.	
Ошибка позиционирования в файле		Обратитесь к разработчикам.
Строковый буфер переполнен		Обратитесь к разработчикам.
Версия ERWIN .ERX не 2.x файл .erx испорчен или не .erx формата	ERWIN .erx не 2.x версии. Входной файл не .erx формата.	
Ошибка Linter	Ошибка СУБД ЛИНТЕР.	Обратитесь к разработчикам.
Логическая ошибка во внутренних таблицах		Обратитесь к разработчикам.
Неподдерживаемый тип БД		См. стр. 12.
Ошибка открытия файла соответствий типов	Файл соответствия типов отсутствует по указанному пути.	
файл соответствий типов содержит более 64 описаний типов		См. стр. 12.
Неверный формат файла соответствий типов		См. стр. 12.
Ошибка создания выходного файла		
Невозможно создать выходной файл		
Ошибка файловой системы		



