СИСТЕМА
УПРАВЛЕНИЯ
БАЗАМИ
ДАННЫХ

ЛИНТЕР®

ЛИНТЕР БАСТИОН

ЛИНТЕР СТАНДАРТ

Экспорт структур и данных

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ

РЕЛЭКС

Товарные знаки

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

Интеллектуальная собственность

Правообладателем продуктов ЛИНТЕР® является компания РЕЛЭКС (1990-2026). Все права защищены.

Данный документ является результатом интеллектуальной деятельности, права на который принадлежат компании РЕЛЭКС.

Все материалы данного документа, а также его части/разделы могут свободно размещаться на любых сетевых ресурсах при условии указания на них источника документа и активных ссылок на сайты компании РЕЛЭКС: relex.ru и linter.ru.

При использовании любого материала из данного документа несетевым/печатным изданием обязательно указание в этом издании источника материала и ссылок на сайты компании РЕЛЭКС: relex.ru и linter.ru.

Цитирование информации из данного документа в средствах массовой информации допускается при обязательном упоминании первоисточника информации и компании РЕЛЭКС.

Любое использование в коммерческих целях информации из данного документа, включая (но не ограничиваясь этим) воспроизведение, передачу, преобразование, сохранение в системе поиска информации, перевод на другой (в том числе компьютерный) язык в какой-либо форме, какими-либо средствами, электронными, механическими, магнитными, оптическими, химическими, ручными или иными, запрещено без предварительного письменного разрешения компании РЕЛЭКС.

О документе

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

Контактные данные

394006, Россия, г. Воронеж, ул. Бахметьева, 2Б.

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

e-mail: info@linter.ru.

Техническая поддержка

С целью повышения качества программного продукта ЛИНТЕР и предоставляемых услуг в компании РЕЛЭКС действует автоматизированная система учёта и обработки пользовательских рекламаций. Обо всех обнаруженных недостатках и ошибках в программном продукте и/или документации на него просим сообщать нам в раздел Поддержка на сайте ЛИНТЕР.

Содержание

Предисловие 3
Назначение документа 3
Для кого предназначен документ 3
Необходимые предварительные знания 3
Дополнительные документы 3
Назначение 4
Условия выполнения 5
Запуск программы 6
Ключи 10
Управление утилитой 10
Авторизация пользователя 10
Спецификация каталога выгрузки 10
Спецификация БД-источника 10
Режим экспорта данных 11
Размер расширения таблицы при последующей загрузке 12
Завершение работы при ошибочных данных 12
Формат представления NULL-значений 12
Кодовая страница для представления строки байтов 12
Управление работой в интерактивном режиме 13
Управление внешними ссылками 13
Экранирование символа $ 13
Экспорт объектов конкретного владельца 13
Экспорт объектов конкретного пользователя одновременно с общедоступными объектами 14
Экспорт объектов конкретного пользователя без запроса пароля 14
Экспорт объектов группы пользователей 14
Экспорт таблиц в файловую систему, не поддерживающую кодировку БД 15
Экспорт таблиц без сохранения кодировки данных 15
Экспорт таблиц для последующего импорта в ранние версии СУБД 16
Указание о кодировке пути к BLOB-файлам 17
Кодировка экспортируемых данных 18
Кодировка командных файлов 18
Спецификация файла регистрационных данных 18
Подавление запроса на ввод пароля 19
Оптимизация размера SQL-скрипта 19
Полный запрет экспорта нулевых default-значений 20
Выборочный запрет экспорта нулевых default-значений 20
Запрет экспорта внешних файлов 20
Разрешение экспорта нулевых дат 20
Экспорт только данных 21
Спецификация файла экспорта 21
Спецификация разделителя данных в файле выгрузки 21
Экспорт в XML-формате 22
Замена некорректного значения типа REAL и DOUBLE на NULL-значение 22
Замена некорректного значения типа REAL и DOUBLE на DEFAULT-значение 22
Замена некорректного значения типа REAL и DOUBLE на указанное значение 23
Экспорт объектов 23
Экспорт базовых таблиц 23
Режим экспорта базовых таблиц 23
Экспорт имен столбцов 24
Экспорт триггеров 25
Экспорт хранимых процедур 25
Экспорт глобальных переменных 25

Экспорт хранимых событий 25
Экспорт структур БД 26
Экспорт пользователей БД 26
Экспорт ролей 26
Экспорт представлений 26
Экспорт синонимов 26
Экспорт всех привилегий 27
Экспорт привилегий к таблице 27
Экспорт последовательностей 27
Экспорт BLOB-данных с блокировкой 27
Экспорт узлов репликации 27
Экспорт правил репликации 27
Экспорт объектов расширенных средств безопасности 28
Экспорт установок аудита 28
Экспорт результата SQL-запроса 28
Экспорт объектов и данных с оперативным созданием внешних ссылок 28
Экспорт объектов и данных с отложенным созданием внешних ссылок 29
Экспорт BLOB-данных в шестнадцатеричном виде 30
Настройка интерфейса 30
Русскоязычный интерфейс 30
Англоязычный интерфейс 30
Сведения об утилите 30
Полный экспорт БД 32
Протокол выполнения 34
Формат экспортированного файла данных 36
Коды завершения 38
Приложение. Примеры использования утилиты 39

Предисловие

Назначение документа

Документ содержит описание утилиты `dbstore`, выполняющей экспорт объектов БД ЛИНТЕР (таблиц, представлений, групп пользователей и т.п.) в текстовые файлы.

Утилита `dbstore` может использоваться:

- совместно с утилитой `loarel` (см. документ «Импорт данных») – для переноса всех или отдельных объектов БД ЛИНТЕР при переходе к другой версии СУБД ЛИНТЕР и/или к другой аппаратной платформе (операционной системе);
- для переноса объектов БД ЛИНТЕР в другие БД, системы управления которыми поддерживают SQL-синтаксис и структуру файла экспорта СУБД ЛИНТЕР;
- для создания резервной копии объектов базы данных в виде текстовых файлов.

Документ предназначен для СУБД ЛИНТЕР БАСТИОН 6.0 сборка 20.8, далее по тексту СУБД ЛИНТЕР.

Для кого предназначен документ

Документ предназначен для системных администраторов и профессиональных пользователей СУБД ЛИНТЕР.

Необходимые предварительные знания

Для работы с утилитой необходимо:

- знать основы реляционных баз данных;
- быть зарегистрированным пользователем СУБД ЛИНТЕР;
- уметь работать в соответствующей операционной системе на уровне пользователя.

Дополнительные документы

- Импорт данных
- Сетевые средства
- Создание и конфигурирование базы данных

Назначение

Утилита `dbstore` предназначена для выгрузки данных и структуры объектов БД ЛИНТЕР в текстовые файлы.

Утилита может использоваться:

- для переноса всех (отдельных) данных либо структуры базы данных (БД) ЛИНТЕР между различными версиями этой СУБД и/или различными платформами (операционными системами);
- для переноса данных и структуры БД ЛИНТЕР на другие СУБД, которые поддерживают SQL-синтаксис и структуру файла экспорта, принятые у СУБД ЛИНТЕР;
- для возможности наглядно увидеть SQL-запрос на создание того или иного объекта СУБД ЛИНТЕР;
- для возможности извлечения данных из таблиц в неинтерактивном режиме;
- для создания своеобразной резервной копии БД.

Для работы с утилитой необходимо:

- знать основы реляционных БД;
- обладать соответствующими правами в СУБД ЛИНТЕР;
- уметь работать в операционной системе на уровне простого пользователя.

Условия выполнения

Утилита функционирует на всех аппаратных и программных платформах, где возможен запуск СУБД ЛИНТЕР.

Для работы утилиты необходимы следующие условия:

- СУБД ЛИНТЕР должна быть активна;
- в момент запуска утилиты СУБД ЛИНТЕР должна иметь в общем случае до шести свободных каналов;
- необходимое свободное место на носителе выгружаемых данных.

Примечание
При работе с ядром в режиме `/COMPATIBILITY=STANDARD` в команду запуска утилиты `dbstore` следует добавлять ключ `-b`.

Запуск программы

Запуск утилиты на выполнение осуществляется стандартными средствами запуска задач, имеющимися в каждой операционной системе. Исполняемый файл утилиты – `dbstore`. Идентификаторы ключей, их возможные значения и ссылки на подробное описание приведены в таблице 1.

Команда запуска:

[1] <синтаксис>::= dbstore [<командная строка>]
[2] <командная строка>::= <ключ>[{<пробел><ключ>} [...]]
[3] <ключ>::= -<идентификатор ключа>[<пробел><значение ключа>]

Таблица 1. Идентификаторы и значения ключей командной строки

<идентификатор ключа>	<значение ключа>	Комментарий
Переключатели		
`-u`	<имя>[/<пароль>]	см. Авторизация пользователя
`-d`	<каталог>	см. Спецификация каталога выгрузки
`-n`	<сервер>	см. Спецификация БД-источника
`-xml`		см. Экспорт в XML-формате
`-o`	<режим>[<режим>…]	см. Режим экспорта данных
`-r`	<резерв>	см. Размер расширения таблицы при последующей загрузке
`-sf`		см. Экспорт схемы и данных пользователя БД
`-af`		см. Экспорт объектов и данных с отложенным созданием внешних ссылок
`-bh`		см. Экспорт BLOB-данных в шестнадцатеричном виде
`-p`	<NULL-значение>	см. Формат представления NULL-значений
`-sep`	[<символ>]	см. Спецификация разделителя данных в файле выгрузки
`-g`		см. Кодовая страница для представления строки байтов
`-q`		см. Управление работой в интерактивном режиме
`-end`		см. Завершение работы при ошибочных данных
`-s`		см. Управление внешними ссылками
`-z`		см. Экранирование символа $
`-w`	<пользователь>|<схема>	см. Экспорт объектов конкретного владельца

© Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

<идентификатор ключа>	<значение ключа>	Комментарий
-wu	<пользователь>	см. Экспорт объектов конкретного пользователя без запроса пароля
-k	<кодовая страница>	см. Кодировка экспортируемых данных
-b		см. Экспорт BLOB-данных с блокировкой
-skipCharsets		см. Экспорт таблиц без сохранения кодировки данных
-compatibleWith5		см. Экспорт таблиц для последующего импорта в ранние версии СУБД
-pathsBlobOEM	[0|1]	см. Указание о кодировке пути к BLOB-файлам
-tn		см. Замена некорректного значения типа REAL и DOUBLE на NULL-значение
-td		см. Замена некорректного значения типа REAL и DOUBLE на DEFAULT-значение
-ts	[значение]	см. Замена некорректного значения типа REAL и DOUBLE на указанное значение
Ключи управления паролями		
-plist	<файл> [-u <имя>]	см. Спецификация файла регистрационных данных
-ep		см. Подавление запроса на ввод пароля
Ключи сохранения объектов		
-ot|-t	[<таблица>[, …]]	см. Экспорт базовых таблиц
-otrig	[<триггер>[, …]]	см. Экспорт триггеров
-op	[<процедура>[, …]]	см. Экспорт хранимых процедур
-ol	[<глобальная переменная>[, …]]	см. Экспорт глобальных переменных
-ostruct		см. Экспорт структур БД
-ou	[<пользователь>[, …]]	см. Экспорт пользователей БД
-or	[<роль>[, …]]	см. Экспорт ролей
-ov	[<представление>[, …]]	см. Экспорт представлений
-os	[<синоним>[, …]]	см. Экспорт синонимов
-oa		см. Экспорт всех привилегий
-oat	<имя таблицы>	см. Экспорт привилегий к таблице

<идентификатор ключа>	<значение ключа>	Комментарий
-oq	[<последовательность>[, …]]	см. Экспорт последовательностей
-on	[<сервер>[, …]]	см. Экспорт узлов репликации
-ob	[<правило>[, …]]	см. Экспорт правил репликации
-og		см. Экспорт объектов расширенных средств безопасности
-od		см. Экспорт установок аудита
-oe	[<событие>[, …]]	см. Экспорт хранимых событий
-j	<SQL-запрос>	см. Экспорт результата SQL-запроса
-sep	[<символ>]	см. Спецификация разделителя данных в файле выгрузки
-useOwnerName		см. Экспорт объектов группы пользователей
-wp	<пользователь>	см. Экспорт объектов конкретного пользователя одновременно с общедоступными объектами
Ключи, управляющие созданием файлов результата		
-ex		см. Запрет экспорта внешних файлов
-ds	[число]	см. Оптимизация размера SQL-скрипта
-rep		см. Режим экспорта базовых таблиц
-dz		см. Полный запрет экспорта нулевых default-значений
-duz		см. Выборочный запрет экспорта нулевых default-значений
-dus		см. Разрешение экспорта нулевых дат
-l	<таблица>	см. Экспорт только данных
-f	[<файл>]	см. Спецификация файла экспорта
-cl	[<имя файла>]	см. Экспорт имен столбцов
-tableFileByID		см. Экспорт таблиц в файловую систему, не поддерживающую кодировку БД
Ключи установки языка интерфейса		
-le		см. Англоязычный интерфейс
-ln		см. Кодировка командных файлов

© Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

<идентификатор ключа>	<значение ключа>	Комментарий
`-lr`		см. Русскоязычный интерфейс
Справочная информация о программе		
`-briefversion`		см. Сведения об утилите
`-version`		см. Сведения об утилите
`-h | -?`		см. Сведения об утилите

Синтаксические правила командной строки:

1) за один сеанс работы утилита обрабатывает только одну командную строку;
2) ключи в командной строке можно располагать в любой последовательности;
3) имена ключей являются регистронезависимыми (кроме ключа `-o`);
4) при вводе параметров ключей должны соблюдаться правила именования объектов СУБД ЛИНТЕР;
5) если задан неизвестный ключ, то утилита выводит на консоль ошибочный ключ и прекращает свою работу;
6) признаком ключа является знак минус «-».

БД-источником для утилиты является локальная БД по умолчанию, установленная на том компьютере, на котором выполняется утилита. Для работы с другой локальной БД или удаленной БД необходимо использовать ключ `-n` (см. Спецификация БД-источника).

Примеры использования утилиты рассмотрены в приложении.

Ключи

Управление утилитой

Авторизация пользователя

`-u <имя>[/<пароль>]`

Задает регистрационные данные пользователя БД, от имени которого должна работать утилита.

Если в командной строке этот ключ не задан, задан не полностью, или утилита запущена вообще без командной строки, то отсутствующие регистрационные данные будут запрошены в интерактивном режиме после старта утилиты в виде вопросов:

```
Ваше имя:

Ваш пароль:
```

Примечание
Если пользователь идентифицируется по протоколу (`identified by protocol`), то пароль запрашиваться не будет.

В этом случае после ввода регистрационных данных утилита выполняет полный экспорт локальной БД по умолчанию в подкаталог `/DB_STORE` установочного каталога СУБД ЛИНТЕР.

Примеры

```
-u SYSTEM/MANAGER8
-u "System"/MANAGER8
-u "Админ"/"5Юъ67~"
-u SYSTEM (пароль будет запрошен интерактивно)
```

Спецификация каталога выгрузки

`-d <каталог>`

Задает полную спецификацию каталога, в котором должны быть размещены файлы с выгружаемыми данными (файлы с расширением `.lod`). По умолчанию используется (при отсутствии – предварительно создается) подкаталог `DB_STORE` в каталоге запуска утилиты.

Пример

```
-d /usr/linter/export
```

Спецификация БД-источника

`-n <сервер>`

Задает имя ЛИНТЕР-сервера, на котором расположена БД-источник экспортируемых данных. Ключ необходим, если выполняется экспорт данных из локальной (не по

умолчанию) БД или удаленной БД. Если ключ не задан, используется локальная по умолчанию БД (см. документ «Сетевые средства»).

Примеры

```
-n "ReplServ"
-n "Склад"
-n BANK
```

Режим экспорта данных

`-o<режим>[<режим>…]`

Задает режим экспорта данных.

Ключ `-o` задается только в строчном виде, режимы – только прописными буквами.

Виды режимов:

- `D` – оптимизация восстановления выгруженных данных. Если задан этот режим, утилита добавляет ключ «быстрой» загрузки в генерируемую командную строку для утилиты импорта структур и данных `loarel`. Используется по умолчанию;
- `N` – устанавливает режим «медленной» загрузки для утилиты импорта структур и данных `loarel` (в противоположность режиму «быстрой» загрузки);
- `B` – режим восстановления выгруженной БД. Если он задан, то восстанавливаемая БД создается с параметрами, используемыми по умолчанию утилитой создания базы данных `gendb`; если не задан – БД восстанавливается в соответствии со своими исходными параметрами;
- `T` – минимизирует размеры выгружаемых таблиц. Если режим установлен, то для пустой таблицы сохраняется значение параметра `MAXROWID`, равное `1`, для непустой таблицы – равное количеству записей в ней (реально то и другое при восстановлении таблицы округляется по модулю 1022 с избытком). Количество и размеры файлов таблицы не сохраняются и при восстановлении таблицы берутся значения по умолчанию. Если этот режим не установлен, сохраняются и впоследствии восстанавливаются и текущее значение параметра `MAXROWID`, и число, и размеры файлов таблицы;
- `S` – экспорт только структуры объекта (без сохранения данных). По умолчанию всегда экспортируются и структура, и данные;
- `F` – выгрузка всех BLOB-значений столбца таблицы в один файл. По умолчанию каждое BLOB-значение таблицы выгружается в отдельный файл;
- `M` – режим быстрого (пакетного) экспорта данных. По умолчанию используется режим экспорта по одной записи таблицы.

По умолчанию (если ключ `-o` не задан) используются следующие режимы:

- последующая быстрая загрузка данных;
- исходные размеры восстанавливаемой БД;
- запрет создания пустых таблиц при импорте данных;
- экспорт и структуры, и данных объекта;
- раздельная выгрузка BLOB-данных;
- медленный (по одной записи) экспорт данных.

Примечание
Экспорт таблиц, представлений, синонимов, последовательностей, хранимых процедур, триггеров и событий в SQL-файлы выполняется с указанием имени схемы, поэтому в SQL-файлы добавляется создание схемы и назначение текущей схемы.

Примеры

```
-o S
-o SFM
-o TD
```

Размер расширения таблицы при последующей загрузке

```
-r <резерв>
```

Задает размер расширения таблицы во время ее загрузки утилитой `loarel`. Параметр `<резерв>` – целое положительное число, равное количеству страниц расширения (1 страница равна 4096 байтам). Ключ применяется в том случае, если предполагается увеличение размера таблицы после ее загрузки. Если ключ не задан, то при загрузке в БД таблица создается в соответствии со своими исходными размерами.

Действие ключа распространяется на все выгружаемые в текущем сеансе утилиты таблицы.

Пример

```
-r 100
```

Завершение работы при ошибочных данных

```
-end
```

Заставляет утилиту завершать работу при возникновении любой ошибки. Если ключ не задан, то в интерактивном сеансе реакцию утилиты на ошибку задает пользователь (продолжить выполнение, завершить, игнорировать ошибки), в пакетном режиме по умолчанию используется режим игнорирования ошибок.

Формат представления NULL-значений

```
-p <NULL-значение>
```

Задает способ представления NULL-значений в выгружаемых файлах. Если ключ не задан, по умолчанию используется ключ `-p <NULL>`.

Примеры

```
-p null
-p "не определено"
-p <NULL>
```

Кодовая страница для представления строки байтов

```
-g
```

Задает вывод строки байтов в виде строки символов в кодовой странице 866. По умолчанию строка байтов выводится в виде строки шестнадцатеричных цифр.

Управление работой в интерактивном режиме

`-q`

В интерактивном режиме на все вопросы утилиты о продолжении работы автоматически генерируется ответ `Continue` (продолжение работы). Сами вопросы в этом случае на консоль не выводятся.

Управление внешними ссылками

`-s`

Заставляет автоматически генерировать SQL-скрипт (текстовый файл с расширением `.sql` и конструкцией `ALTER TABLE`) для создания ссылок (внешних ключей) на таблицу, создаваемую утилитой `loarel` из экспортированной таблицы. Если ключ не задан, внешние ссылки, при необходимости, должны создаваться вручную.

Экранирование символа $

`-z`

Заменяет символ `$` на `\$` в генерируемых командных файлах пакетной загрузки. Это необходимо для корректной интерпретации командной строки в ОС Linux, ЗОСРВ Нейтрино, так как этот символ является служебным.

Экспорт объектов конкретного владельца

`-w <пользователь>|<схема>`

Задает экспорт только тех объектов, владельцем которых является указанный `<пользователь>` или `<схема>`. Если в БД есть объекты, владельцем которых является `<пользователь>` (или `<схема>`), то при отсутствии ключа `-u` утилита потребует ввести в интерактивном режиме пароль этого `<пользователя>` или владельца `<схемы>`. Для выполнения данной операции пользователь БД, от имени которого работает утилита `dbstore` (ключ `-u`), должен иметь соответствующие привилегии.

Пример

```
dbstore -u SYSTEM/MANAGER8 -w SYS

           Сохраняю базу данных.
           ---------------------
Сохраняю уровни...
           Уровни отсутствуют.
Сохраняю станции...
           Станции отсутствуют.
Сохраняю устройства...
           Устройства отсутствуют.
...
```

```
Введите пароль для пользователя 'SYS':
```

…

Экспорт объектов конкретного пользователя одновременно с общедоступными объектами

`-wp <пользователь>`

Выполняет функции ключа `-w` и дополнительно экспортирует все общедоступные (`public`) объекты БД (последовательности и синонимы).

Экспорт объектов конкретного пользователя без запроса пароля

`-wu <пользователь>`

Задает экспорт только тех объектов, владельцем которых является указанный `<пользователь>`. В процессе работы пароль `<пользователя>` не запрашивается. Для выполнения данной операции пользователь БД, от имени которого работает утилита `dbstore` (ключ `-u`), должен иметь соответствующие привилегии.

Если привилегии предоставить нельзя (например, для ролей) надо использовать `-plist` или вводить пароль с консоли.

При экспорте пользовательской последовательности дополнительно экспортируется `PUBLIC SEQUENCE`.

Примеры

1) Утилита запущена от имени пользователя, являющегося владельцем экспортируемых таблиц.

```
dbstore -u USR1/123 -wu USR1 -ot TAB1,TAB2,TAB3
```

В каталоге выгрузки будет создан подкаталог `USR1.lod` с файлами `TAB1.lod`, `TAB2.lod`, `TAB3.lod`.

2) Утилита запущена от имени пользователя, не являющегося владельцем экспортируемых таблиц. В данном случае у пользователя `SYSTEM/MANAGER8` должна быть привилегия на чтение данных из таблицы `USR1.AUTO`:

```
inl>username USR1/123
inl>grant select on TAB1 to SYSTEM;
dbstore -u SYSTEM/MANAGER8  -wu USR1 -ot TAB1,TAB2
```

В каталоге выгрузки будет создан подкаталог `USR1.lod` с файлами `TAB1.lod`, `TAB2.lod`.

Экспорт объектов группы пользователей

`-useOwnerName`

Задает экспорт объектов, принадлежащих различным пользователям БД. Используется одновременно с ключами экспорта объектов. При работе в неинтерактивном режиме требуется также ключ `-plist`.

 © Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

Имена экспортируемых объектов должны быть представлены в виде <имя владельца>.<имя объекта>.

<Имя владельца> может быть задано в виде шаблона. Знак «%» в шаблоне соответствует любой последовательности символов, знак «_» – любому символу в данной позиции.

В процессе работы требуется предоставить пароли тех владельцев, чьи объекты экспортируется: либо ввести с консоли, либо указать их в файле паролей с помощью ключа -plist.

Для выполнения данной операции пользователь БД, от имени которого работает утилита dbstore (ключ -u), должен иметь привилегии на доступ к данным экспортируемых объектов.

Примеры

1) Экспорт всех имеющихся в БД таблиц TAB1. Пароли владельцев таблиц берутся из файла dbpls.txt.

```
dbstore -u SYSTEM/MANAGER8 -useOwnerName -plist dbpls.txt -ot
 %.TAB1
```

2) Экспорт всех имеющихся в БД таблиц TAB1, владельцы которых имеют четырехсимвольные имена вида USRx, где x – любой знак. Пароли владельцев таблиц берутся из файла dbpls.txt.

```
dbstore -plist dbpls.txt -u USR1/123 -useOwnerName -ot USR_.TAB1
```

Экспорт таблиц в файловую систему, не поддерживающую кодировку БД

-tableFileByID

Позволяет сохранять таблицы, если файловая система ОС не поддерживает имена в кодировке словаря БД (например, русскоязычные названия файлов). В этом случае имена экспортируемых файлов (lod-файлов) будут иметь вид:

<идентификатор владельца таблицы><идентификатор таблицы>.lod

В случае если числовое значение идентификаторов меньше четырех цифр, то их представление дополняется нулями слева.

Пример

```
dbstore -u SYSTEM/MANAGER8 -tableFileByID -ot AUTO
```

В подкаталоге SYSTEM.lod будет создан файл 00010122.lod.

Экспорт таблиц без сохранения кодировки данных

-skipCharsets

Ключ используется для экспорта данных из БД ЛИНТЕР более поздней версии для их последующего импорта в БД ЛИНТЕР версии 5.9, не поддерживающей кодировку данных, либо в БД любой версии, установленную в другой ОС.

При задании ключа:

- информация о кодировке данных не включается в генерируемые SQL-запросы на создание выгруженных таблиц;
- данные выгружаются в текущей кодировке;
- не создается SQL-скрипт `db_chars.sql`, содержащий SQL-запросы на установление кодировок словаря БД и кодировки БД по умолчанию (`database names` и `default character set`).

Примеры

1)

```
dbstore -u USR1/123 -useOwnerName -ot USR1.TAB1,USR1.TAB2
```

В оператор создания таблицы будет добавлено указание о кодировке:

```
CREATE TABLE "TAB1" CHARACTER SET "CP1251" …
```

2)

```
dbstore -u USR1/123 -useOwnerName -skipCharsets -ot
 USR1.TAB1,USR1.TAB2
CREATE TABLE "TAB1" …
```

В этом примере выгрузка специально делается максимально эквивалентной для загрузки в БД версии 5.9 (напрмер: переводит все строки и пути в OEM-кодировку, убирает лишние ключи в скриптах). Разумеется, если БД в принципе не может быть сконвертирована в 5.9, то данная выгрузка будет некорректна.

Экспорт таблиц для последующего импорта в ранние версии СУБД

`-compatibleWith5`

Ключ используется для экспорта данных из БД ЛИНТЕР более поздней версии, которая поддерживает различные кодировки, с целью последующего импорта в БД ЛИНТЕР версии 5.x, которая поддерживает только OEM-кодировку (CP866 – для ОС типа Windows, KOI8-R/ISO – для ОС Linux, ЗОСРВ Нейтрино).

Примечание
Ключ учитывается только при экспорте всей БД (а не отдельных её объектов).

При задании ключа:

- в команды скриптов загрузки БД для утилиты `SPC` (транслятор триггеров и хранимых процедур) не добавляется параметр `-c` <имя кодировки>;
- если кодировка словаря БД CP1251, то запросы на создание триггеров/процедур будут переведены в кодировку CP866 (для ОС типа Windows) или KOI8-R/ISO (для ОС Linux, ЗОСРВ Нейтрино);
- для ОС типа Windows данные и пути к файлам выгруженных данных сохраняются в кодировке OEM;

 © Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

- для ОС типа Linux, ЗОСРВ Нейтрино будет использована кодировка KOI8-R/ISO, при этом ключ задания кодировки `-k` игнорируется;
- в параметры утилиты загрузки данных (`loarel`) в скриптовом файле не добавляется ключ `-pathsBlobOEM`;
- в bat-файл для утилиты `inl` не добавляются команды создания системных таблиц `$$$CHARSET`, `$$$TRANSL`, `$$$CSALIAS` (файл `cstables.sql`), заполнения таблицы `$$$CHARSET` кодировками (файл `charsets.sql` для однобайтовых кодировок и файл `mbcps.sql` для многобайтовых кодировок).

При экспорте данных информация о кодировке словаря БД и кодировке БД по умолчанию (SQL-команды `SET DATABASE NAMES` и `SET DATABASE DEFAULT CHARACTER SET`) сохраняется в файле `db_chars.sql`. Чтобы исключить создание этого файла для режима совместимости, необходимо вместе с ключом `-compatibleWith5` задавать одновременно и ключ `-skipCharsets`.

Пример

```
dbstore -u SYSTEM/MANAGER8 -ot AUTO -compatibleWith5 > dbs.txt
set LINTER_CP=CP866
inl -u "SYSTEM"/ -f tab_0001.sql
set LINTER_CP=
loarel -u "SYSTEM"/ -t "AUTO" -d "SYSTEM.lod" -s 1 -f "AUTO.lod"
 -k CP866
set LINTER_CP=CP866
inl -u "SYSTEM"/ -f idx_0001.sql
inl -u "USR1"/ -f tab_001f.sql
inl -u "USR2"/ -f tab_0020.sql
inl -u "USR3"/ -f tab_0021.sql
```

Указание о кодировке пути к BLOB-файлам

```
-pathsBlobOEM[0|1]
```

Ключ поддерживается только в среде ОС Windows.

Если значение ключа равно `1`, то путь к BLOB-файлам прописывается в кодировке OEM (кодовая страница CP866); если 0 – то в кодировке ANSI (кодовая страница CP1251).

Если ключ или его значение не заданы, по умолчанию используется `-pathsBlobOEM=1`.

Примечание
При вызове из утилиты `migration` значение по умолчанию `0`.

Примеры

Пусть в БД есть таблица пользователя `SYSTEM` с BLOB-столбцом "Каталог пиктограмм".

1)

```
dbstore -u SYSTEM/MANAGER8  -pathsBlobOEM=1 -ot "Каталог
 пиктограмм"
```

В подкаталоге SYSTEM.lod в файле "каталог пиктограмм.lod" путь к BLOB-файлам будет прописан так:

```
,1,0 "DB_STORE\SYSTEM.lod\КАТАЛОГ ПИКТОГРАММ.blb\00000001.002",
,2,0 "DB_STORE\SYSTEM.lod\КАТАЛОГ ПИКТОГРАММ.blb\00000002.002",
```

2)

```
dbstore -u SYSTEM/MANAGER8  -pathsBlobOEM=0 -ot "Каталог
 пиктограмм"
```

В подкаталоге SYSTEM.lod в файле "каталог пиктограмм.lod" путь к BLOB-файлам будет прописан так (в кодировке ANSI):

```
,1,0 "DB_STORE\SYSTEM.lod\КАТАЛОГ ПИКТОГРАММ.blb\00000001.002",
,2,0 "DB_STORE\SYSTEM.lod\КАТАЛОГ ПИКТОГРАММ.blb\00000002.002",
```

Кодировка экспортируемых данных

```
-k <кодовая страница>
```

Задает кодировку, в которой должны выгружаться данные.

Если ключ не задан, по умолчанию для ОС Windows используется кодовая страница CP866, для ОС Linux, ЗОСРВ Нейтрино используется кодовая страница KOI8-R.

Примечание
Значение переменной LINTER_CP, задающее кодировку, не распространяется на кодировку соединения с ЛИНТЕР-сервером, необходимо указывать кодировку с помощью ключа -k.

Примеры

```
-k CP1251
-k UTF-8
```

Кодировка командных файлов

```
-ln
```

Указывает, что командные файлы (формат .bat) должны генерироваться на латинице (необходимо для корректного переноса БД, например, между разными операционными системами).

Спецификация файла регистрационных данных

```
-plist  <файл> [-u <имя>]
```

Задает полную спецификацию текстового файла в кодовой странице 866, содержащего полные или частичные регистрационные данные пользователей БД, чьи объекты должны быть экспортированы. Если утилите в процессе работы потребуется пароль пользователя для выгрузки данных из его таблицы, то пароль сначала будет искаться в указанном файле. Если такового в файле не найдется, он будет запрошен интерактивно. Регистрационные данные пользователей должны располагаться в отдельных строках файла.

Возможны два варианта запуска утилиты с ключом `-plist`:

1) с указанием имени пользователя в командной строке (пароль будет прочитан из <файла>):

```
dbstore -plist plist.txt -u SYSTEM
```

2) без указания имени пользователя в командной строке. В этом случае имя пользователя придётся вводить вручную, а пароль будет читаться из <файла>:

```
dbstore -plist plist.txt
```

Содержимое файла `plist.txt`:

```
SYSTEM/MANAGER8
```

Пример

```
-plist /usr/linter/psw.txt
```

Файл `psw.txt`:

```
SYSTEM/MANAGER8
"Админ"/76 9<bz
"Bank_FC"/0cfdsa
...
```

Подавление запроса на ввод пароля

```
-ep
```

Указание использовать пустой пароль, если он не известен. Для экспорта данных конкретного пользователя необходимо знать его регистрационные данные (имя и пароль), которые в общем случае извлекаются из файла регистрационных данных (см. ключ -plist). Если пароль пользователя в указанном файле не найден, он будет запрашиваться интерактивно (режим по умолчанию). Ключ `-ep` подавляет эту операцию. Но чтобы данные были реально экспортированы, пользователь БД действительно не должен иметь пароля.

Оптимизация размера SQL-скрипта

```
-ds [число]
```

Заставляет удалять лишние пробелы из генерируемых утилитой SQL-скриптов (файлов с расширением `.sql`).

Значение ключа (если задано) задает длину выводимых в sql-файл имен таблиц и столбцов:

- если [число] больше длины конкретного имени, то выводимое имя будет дополнено пробелами до заданной в [числе] длины;
- если [число] меньше длины конкретного имени, то все завершающие пробелы в имени будут удалены.

В любом случае длина имени будет не больше 66 символов.

Если [число] не задано, то все правосторонние пробелы будут обрезаны, эквивалентно `-ds 0`.

Примеры

1)

```
-ds 10
CREATE TABLE "$$$OBJ_COMMENTS" CHARACTER SET "CP866" (
        "$$$ID     " INTEGER DEFAULT NULL,
        "$$$OBJ_TYPE" INTEGER DEFAULT NULL,
        "$$$OBJ_ID " INTEGER DEFAULT NULL,
        "$$$NMR    " INTEGER DEFAULT NULL,
…)
```

2)

```
-ds 5
CREATE TABLE "$$$OBJ_COMMENTS" CHARACTER SET "CP866" (
        "$$$ID" INTEGER DEFAULT NULL,
        "$$$OBJ_TYPE" INTEGER DEFAULT NULL,
        "$$$OBJ_ID" INTEGER DEFAULT NULL,
        "$$$NMR" INTEGER DEFAULT NULL,
…)
```

Полный запрет экспорта нулевых default-значений

`-dz`

Запрещает экспорт нулевых DEFAULT-значений для столбцов всех типов.

Конструкция "... `DEFAULT 0`" в SQL-скрипты включаться не будет.

Выборочный запрет экспорта нулевых default-значений

`-duz`

Запрещает экспорт нулевых DEFAULT-значений только для столбцов с уникальными значениями.

Конструкция "... `DEFAULT 0`" для столбцов с атрибутами `PRIMARY KEY` и `UNIQUE` в SQL-скрипты включаться не будет.

Запрет экспорта внешних файлов

`-ex`

Запрещает экспортировать внешние файлы (данные типа `EXTFILE`).

По умолчанию экспорт внешних файлов разрешен.

Разрешение экспорта нулевых дат

`-dus`

Разрешает экспортировать нулевые даты в файл выгрузки (lod-файл).

По умолчанию экспорт нулевых дат запрещен.

Экспорт только данных

`-l <таблица>`

Экспортировать только данные из базовой таблицы. Если в качестве имени таблицы задан шаблон, то экспорт производится только из первой таблицы получаемого по шаблону списка таблиц.

По умолчанию выполняется экспорт данных и структуры таблицы.

Пример

```
-l AUTO
```

Спецификация файла экспорта

`-f [<файл>]`

Экспортировать данные в указанный файл (использовать только с ключами `-l`, `-j`, `-sf`).

Если имя файла не задано, по умолчанию создается файл с атрибутами `<имя таблицы>.lod`.

Если имя выгружаемой таблицы содержит специальные символы, то имя выгружаемого файла генерируется по шаблону:

`TBL_<идентификатор пользователя>_<идентификатор таблицы>.`

Примечание
В паре с ключом `-sf` ключ `-f` используется для задания имени единого SQL-скрипта, в который экспортируются объекты и данные.

Примеры

```
-l AUTO -f auto.txt
-l SYS."Банк" -f /usr/linter/export/Bank.lod
```

Спецификация разделителя данных в файле выгрузки

`-sep [<символ>]`

Заставляет использовать в качестве разделителя данных в файле выгрузки указанный одиночный <символ>.

Если ключ не задан, по умолчанию в качестве разделителя используется символ «,» (запятая).

Примечание
Если указанный разделитель использовать невозможно, необходимо подобрать другой символ.

Пример

```
dbstore -u SYSTEM/MANAGER8 -ot AUTO -sep !
```

Экспорт в XML-формате

`-xml`

Заставляет экспортировать данные и структуру объектов БД в XML-формате.

Примечание
Режим устарел. Применять не рекомендуется.

По умолчанию экспорт данных и структуры выполняется в текстовом формате.

Пример

```
-l AUTO -f auto.txt -xml
```

Выгруженные данные будут представлены в следующем виде:

1) описание структуры столбца

```
<column num="1" name="MAKE" length="20" type="CHAR" precision="0"
 scale="0" />
```

2) описание строки таблицы

```
<tr num="1">
  <td num="1">FORD</td>
  <td num="2">MERCURY COMET GT V8</td>
  <td num="3">COUPE</td>
  <td num="4">8</td>
  <td num="5">143</td>
  <td num="6">302</td>
  <td num="7">2900</td>
  <td num="8">BLACK</td>
  <td num="9">71</td>
  <td num="10">O333306891523808</td>
  <td num="11">720308</td>
  <td num="12">49173</td>
  <td num="13">1</td>
</tr>
```

Замена некорректного значения типа REAL и DOUBLE на NULL-значение

`-tn`

Выполняет замену некорректных значений типов `REAL` и `DOUBLE` на NULL-значение.

Замена некорректного значения типа REAL и DOUBLE на DEFAULT-значение

`-td`

Выполняет замену некорректных значений типов REAL и DOUBLE на DEFAULT-значение.

Замена некорректного значения типа REAL и DOUBLE на указанное значение

```
-ts [значение]
```

Выполняет замену некорректного значения типа REAL и DOUBLE на указанное значение.

Экспорт объектов

При использовании данных ключей следует руководствоваться следующими правилами:

1) <имя таблицы> должно относиться к пользовательской базовой таблице. Если пользователь предпринимает попытку выгрузить системную таблицу, утилита не выдает сообщение об ошибке, но выгрузка этой таблицы производиться не будет;
2) спецификация имени экспортируемого объекта (<таблица>, <представление>, <процедура> и др.) не должна содержать имя владельца объекта. Если в БД имеются объекты с одинаковыми именами (но разными владельцами), будут выгружены все такие объекты, к которым имеется доступ на выборку данных;
3) для спецификации имени объекта можно использовать маску шаблона, которая должна соответствовать требованиям конструкции LIKE языка SQL СУБД ЛИНТЕР (то есть в маске допускается использовать символ % и символ подчеркивания «_»);
4) если выгружаемый объект пуст (нет данных), файл выгрузки не создается;
5) если в ключе не задан конкретный список экспортируемых объектов, то экспортируются все объекты данного типа.

Экспорт базовых таблиц

```
-ot|-t [<таблица>[, …]]
```

Задает полный экспорт базовой таблицы:

- имени схемы;
- комментария таблицы и столбцов;
- данных.

Примечание
При экспорте таблицы со столбцами типа BIGINT с атрибутом autoinc initial <n> начальное значение устанавливается в 1, о чем выдается соответствующее предупреждение. Правильное начальное значение должно быть установлено вручную путем корректировки соответствующего sql-файла.

Примеры

```
-ot AUTO
-ot "Авто",PERSON,BA%
```

Режим экспорта базовых таблиц

```
-rep
```

Задает экспорт базовых таблиц в режиме замены (добавление конструкции OR REPLACE в оператор создания таблицы).

По умолчанию оператор создания таблицы имеет вид:

```
create table <имя таблицы> ….
```

При указании ключа -rep оператор создания таблицы будет иметь вид:

```
create or replace table <имя таблицы> ….
```

Примеры

1) Формат оператора создания таблицы по умолчанию:

```
create table "auto" character set "cp866" …
```

2) Формат оператора создания таблицы при использовании ключа -rep:

```
create or replace table "auto" character set "default" …
```

Экспорт имен столбцов

```
-cl [<имя файла>]
```

Заставляет создавать файл с именами столбцов экспортированной базовой таблицы или выполненного SQL-запроса.

Ключ должен использоваться одновременно с одним из ключей -l или -j.

Вариант с ключом -l:

задан ключ -l (экспорт только данных таблицы). В этом случае список экспортированных столбцов будет сформирован в указанном файле. Если <имя файла> не указано, то по умолчанию будет создан файл <имя таблицы>.txt.

Примеры

1)

```
dbstore -u SYSTEM/MANAGER8 -l AUTO -cl
```

Список столбцов будет выгружен в файл AUTO.txt (в каталоге запуска dbstore).

2)

```
dbstore -u SYSTEM/MANAGER8 -l AUTO -cl "/usr/linter/Столбцы
 AUTO.txt"
```

Список столбцов будет выгружен в файл /usr/linter/Столбцы AUTO.txt.

Вариант с ключом -j:

задан ключ -j (то есть экспорт результата выполнения SQL-запроса). Параметр <имя файла> является обязательным. Если в выборке встречаются неименованные столбцы, то в экспортированном списке столбцов они будут именоваться как UNDEFINED COLUMN #<номер>, где <номер> – порядковый номер столбца в выборке, начиная с 1.

 © Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

Пример

```
dbstore -u SYSTEM/MANAGER8 -j "select 1, sysdate, null" -cl
 AUTO_SQL.txt
```

Список столбцов выборки выгружен в файл `AUTO_SQL.txt` (в каталоге запуска `dbstore`):

```
UNDEFINED COLUMN #1
UNDEFINED COLUMN #2
UNDEFINED COLUMN #3
```

Экспорт триггеров

```
-otrig [<триггер>[, …]]
```

Задает экспорт триггеров.

Примеры

```
-otrig INSERT_AUTO
-otrig "Корр Авто",TRIG%
```

Экспорт хранимых процедур

```
-op [<процедура>[, …]]
```

Задает экспорт хранимых процедур.

Примеры

```
-op "Оборот за %"
-op "Saldo",SP,_%%R
```

Экспорт глобальных переменных

```
-ol [<глобальная переменная>[, …]]
```

Задает экспорт глобальных переменных.

Примечание
Поддерживается со сборки 6.0.17.92.

Пример

```
-ol dlob_var
```

Примечание
Поддерживается со сборки 6.0.17.92.

Экспорт хранимых событий

```
-oe [<событие>[, …]]
```

Задает экспорт хранимых событий.

Пример

```
-oe "Удаление заказа",Ev1
```

Экспорт структур БД

```
-ostruct
```

Задает экспорт структуры базы данных, то есть генерирование командного файла для создания пустой базы данных с помощью утилиты `gendb`.

Экспорт пользователей БД

```
-ou [<пользователь>[, …]]
```

Задает экспорт информации о пользователях базы данных.

Примеры

```
-ou
-ou SYSTEM,"Админ"
```

Экспорт ролей

```
-or [<роль>[, …]]
```

Задает экспорт информации о ролях базы данных.

Примеры

```
-or USER,ADMIN,MANAGER8,
-or "Нач.%",Manager_,
```

Экспорт представлений

```
-ov [<представление>[, …]]
```

Задает экспорт представлений.

Представления экспортируются в том порядке, в котором они были созданы в БД, чтобы дать возможность создавать представления, использующие в своём составе другие представления.

Экспорт синонимов

```
-os [<синоним>[, …]]
```

Задает экспорт синонимов.

Примечание
При экспорте синонимов конкретного пользователя по команде `-w <пользователь> -os` общедоступные (`public`) синонимы не экспортируются.

Примеры

```
-os
```

```
-os "Авто","Люди"
```

Экспорт всех привилегий

```
-oa
```

Задает экспорт привилегий доступа пользователей БД.

Экспорт привилегий к таблице

```
-oat <имя таблицы>
```

Задает экспорт привилегий доступа к указанной таблице для всех пользователей.

Экспорт последовательностей

```
-oq [<последовательность>[,…]]
```

Задает экспорт последовательностей.

> **Примечание**
> Последовательности, созданные для IDENTITY-столбцов, не экспортируются.

Примеры

```
-oq SEQ_%
-oq "Платежки%","Договор_"
```

Экспорт BLOB-данных с блокировкой

```
-b
```

Задает блокировку записей при экспорте BLOB-данных.

Если ключ не задан, то:

1) при выгрузке таблицы с BLOB-данными блокируется/деблокируется вся таблица;
2) при выгрузке запросов (ключ `-j`) все блокировки отменяются.

Экспорт узлов репликации

```
-on [<сервер>[, …]]
```

Задает экспорт информации о ЛИНТЕР-серверах (только для версии с поддержкой репликации данных).

Примеры

```
-on
-on "Платежки%","Договор_"
```

Экспорт правил репликации

```
-ob [<правило>[, …]]
```

Задает экспорт информации о правилах репликации (только для версии с поддержкой репликации данных).

Примеры

```
-ob NODE_S%
-ob OFFICE%, "ОтделениеN%"
```

Экспорт объектов расширенных средств безопасности

`-og`

Задает экспорт расширенных средств безопасности данных: группы, уровни доступа, описание станций и устройств в виде SQL-запросов.

Экспорт установок аудита

`-od`

Задает экспорт установок аудита в виде SQL-запросов.

Экспорт результата SQL-запроса

`-j <SQL-запрос>`

Выполняет <SQL-запрос> и сохраняет результат в файле выгрузки.

Пример

```
dbstore -j "select distinct make, model from auto" -u SYSTEM/
MANAGER8 -f auto_count.sql
```

Экспорт объектов и данных с оперативным созданием внешних ссылок

`-sf`

Задает сохранение структуры объектов и данных конкретного пользователя БД в единый текстовый SQL-скрипт (одновременно неявно задается и ключ `-s` для создания ссылок (внешних ключей) на выгружаемую таблицу, создаваемую утилитой `loarel` из экспортированной таблицы). В этом случае при последующей загрузке в БД экспортированных данных корректность внешних ссылок будет проверяться при добавлении каждой записи.

По умолчанию (без ключей `-d` и `-f`) экспорт производится в SQL-скрипт с именем `dbstore.sql`, который создается в каталоге запуска утилиты.

Формат экспорта BLOB-данных задается ключом `-bh`.

При отсутствии ключа `-w` утилита будет производить попытку экспорта объектов и данных всех пользователей, которые зарегистрированы в БД. Экспорт будет успешным только при указании правильных паролей пользователей.

При применении ключа `-sf` происходит игнорирование следующих ключей: `-o` (в режимах `D`, `N`, `B`, `T`, `F`, `M`), `-r`, `-s`, `-z`, `-pathsBlobOEM`, `-j`, `-useOwnerName`, `-cl`.

Экспорт данных, представленных в разных кодировках, выполняется следующим образом:

- по умолчанию (без ключа `-k`) в SQL-скрипте перед текстом команды на создание таблицы вставляется команда `SET NAMES` "<кодовая страница>", где <кодовая страница> – это кодировка символьных данных экспортируемой таблицы;
- перед созданием других объектов БД записывается аналогичная команда, при этом "<кодовая страница>" принимает значение кодировки канала, установленного при запуске утилиты;
- задание ключа `-k` "<кодовая страница>" определяет экспорт всех данных в единой кодировке. Запрос `SET NAMES` записывается при этом один раз в начале SQL-скрипта;
- при использовании совместно с ключом `-sf`, ключа `-compatibleWith5` команды `SET DATABASE NAMES` и `SET DATABASE DEFAULT CHARACTER SET` не экспортируются, а команда `SET NAMES` "<кодовая страница>" записывается один раз в закомментированном виде. В этом случае <кодовая страница> зависит от используемой платформы (CP866 – для ОС типа Windows, KOI8-R/ISO – для ОС типа Linux, ЗОСРВ Нейтрино).

Примеры

1)

```
dbstore -sf -f export_db.sql -u SYSTEM/MANAGER8 -w SYSTEM -k
 "UTF-8"
```

Экспортирует данные и структуру объектов пользователя `SYSTEM` в файл `export_db.sql` текущего каталога, при этом все данные экспортируются в кодировке UTF-8.

2)

```
dbstore -sf -u SYSTEM/MANAGER8
```

Экспортирует структуру объектов и данные всех пользователей БД в единичный SQL-скрипт с именем по умолчанию `dbstore.sql` в подкаталоге `DB_STORE`.

3)

Для ОС Linux:

```
dbstore -sf -d /usr/linter/dump -f export_db.sql -u SYSTEM/
MANAGER8 -w SYSTEM
```

Экспорт объектов и данных с отложенным созданием внешних ссылок

`-af`

Поведение ключа идентично ключу `-sf`, но создание и проверка корректности внешних ссылок будет выполняться после загрузки всех экспортированных данных.

Пример

```
dbstore -af -f export_db.sql -u SYSTEM/MANAGER8 -w SYSTEM
```

В файле export_db.sql для таблицы с внешними ссылками будет сформирована команда создания таблицы без указания внешних, затем команды загрузки данных и далее команда создания внешних ссылок.

Экспорт BLOB-данных в шестнадцатеричном виде

-bh

Задает экспорт BLOB-данных в шестнадцатеричном виде.

Ключ работает только совместно с ключом -sf. При использовании данного ключа выдается предупреждение, если длина шестнадцатеричной строки, в которую экспортируются BLOB-данные, превышает длину строки, потенциально обрабатываемую утилитой inl. По умолчанию (в отсутствии ключа -bh) BLOB-данные экспортируются следующим образом:

- если размер BLOB-данных меньше или равен 2 Кбайт, то они записываются в единый SQL-скрипт в шестнадцатеричном виде;
- если размер BLOB-данных больше 2 Кбайт экспорт производится в отдельный файл в бинарном виде в тот же каталог, где располагается SQL-скрипт.

Примечание
В случае получения сообщения Предупреждение! Слишком большая длина BLOB объекта в шестнадцатеричном представлении не использовать для сохранения такой базы ключ -bh.

Пример

```
dbstore -sf -bh -f export_db.sql -u SYSTEM/MANAGER8 -w SYSTEM
```

Экспортирует схемы и данные пользователя SYSTEM в текущий каталог в файл export_db.sql, при этом BLOB-данные экспортируются в шестнадцатеричном виде.

Настройка интерфейса

Все ключи настройки интерфейса относятся только к текущему сеансу работы утилиты.

Русскоязычный интерфейс

-lr

Выводит сообщения утилиты по-русски (действует по умолчанию).

Англоязычный интерфейс

-le

Выводит сообщения утилиты по-английски.

Сведения об утилите

-version

Вывод на консоль полной информации о версии.

`-briefversion`

Вывод на консоль краткой информации о версии (например, 6.0.17.48).

`-h | -?`

Вывод на консоль справочной информации.

Полный экспорт БД

Полный экспорт БД выполняется по следующим правилам:

1) файлы выгрузки размещаются в каталоге, заданном ключом `-d`;
2) если ключ `-d` не задан, файлы выгрузки по умолчанию размещаются в подкаталоге `DB_STORE`, создающемся автоматически в том каталоге, из которого была запущена утилита `dbstore`;
3) для каждого владельца объектов БД создается индивидуальный подкаталог для размещения его файлов выгрузки (данных и командных);
4) создаваемые подкаталоги именуются как `<имя пользователя>.lod` или `<имя схемы>.lod`. Внутри этого подкаталога имена выгруженных файлов представлены в виде `<имя объекта>.lod`. Если указывается имя файла, в котором будут сохранены данные, то расширение `.lod` не добавляется.

   В случае если текущая схема (или имя пользователя) содержит специальные символы, эти символы при генерации имени подкаталога заменяются на символ '_'.

   Исключение составляют специальные ключевые слова PRN, CON, AUX, NUL, COM0-COM9, LPT0-LPT9. В этих словах третий символ заменяется на символ '_', например, CON на CO_, COM0 на CO_0;
5) имена подкаталогов для хранения BLOB-данных генерируются по следующему правилу: если имя выгружаемой таблицы содержит специальные символы, то имя выгружаемого файла генерируется по шаблону

```
TBL_<идентификатор пользователя>_<идентификатор таблицы>
```

иначе

```
<имя пользователя>.blb
```

или

```
<имя схемы>.blb.
```

В общем каталоге создаются перечисленные в таблице 2 файлы.

Таблица 2. Список файлов полного экспорта БД

Файл	Описание
`cr_db.gdb`	Текст оператора создания экспортированной БД (оператор `create database` утилиты `gendb`)
`create_0.bat`	Командный файл создания экспортированной БД
`create_1.bat`	Командный файл создания системных таблиц экспортированной БД
`create_2.bat`	Командный файл загрузки экспортированной БД
`ac_0001.sql`	SQL-скрипт для создания всех привилегий
`al_users.sql`	SQL-скрипт для замены паролей пользователей БД (используется после восстановления БД)
`cr_users.sql`	SQL-скрипт для создания пользователей БД с пустыми паролями
`idx_0001.sql`	SQL-скрипт для создания индексов
`syn_0001.sql`	SQL-скрипт для создания синонимов

Файл	Описание
tab_0001.sql	SQL-скрипт для создания базовых таблиц
trig0001.sql	SQL-скрипт для создания триггеров
proc0001.sql	SQL-скрипт для создания хранимых процедур
view0001.sql	SQL-скрипт для создания представлений
var0001.sql	SQL-скрипт для создания глобальных переменных
rol_0001.sql	SQL-скрипт для создания ролей
ref_0001.sql	SQL-скрипт для создания внешних ключей
seg_0001.sql	SQL-скрипт для создания последовательностей
replrul.sql	SQL-скрипт для создания правил репликации (только для версии с поддержкой репликации данных)
servers.sql	SQL-скрипт для создания серверов (только для версии с поддержкой репликации данных)
audit.sql	SQL-скрипт для создания установок аудита
groups.sql	SQL-скрипт для создания групп
levels.sql	SQL-скрипт для создания уровней доступа
stations.sql	SQL-скрипт для создания станций и их характеристик
devices.sql	SQL-скрипт для создания описания устройств
db_chars.sql	SQL-скрипт для установки кодировок по умолчанию системных и пользовательских таблиц
dbparams.sql	SQL-скрипт для установки параметров БД
ext.sql	SQL-скрипт для установки фильтров системы фразового поиска

Протокол выполнения

Результат работы утилиты выдается в виде протокола на консоль оператора.

Примечание
С помощью стандартных средств ОС выдача протокола выполнения может быть перенаправлена в файл.

В протокол выполнения вносятся следующие сведения:

- о выполняемых операциях экспортирования;
- о не экспортируемых объектах БД;
- о результате (успешном или неудачном) экспорта каждого объекта БД;
- о причинах неудачного экспорта объекта БД;
- статистические данные о количестве успешно экспортированных объектов.

Пример протокола

```
        Сохраняю базу данных.
Структура базы данных не сохраняется.
Пользователи не сохраняются.
Фильтры не сохраняются.
        Сохраняю таблицы...
        Сохраняю таблицу        PROC_PRIV_TYPES                 OK
        Сохраняю таблицу        COUNTER6                OK
        Сохраняю таблицу        PRIV_TYPES                OK
        Сохраняю таблицу        TYPEINFO                OK
        Сохраняю таблицу        COUNTER8                OK
        Сохраняю таблицу        COUNTER31                OK
        ...
        Сохраняю таблицу SESSION_PRIVS
Предупреждение! Не могу сохранить данные из таблицы
 'SESSION_PRIVS'
(неверный пароль пользователя)!
Продолжить                                      OK
        ...
Сохраняю представления...
        Сохраняю представление        TABLES                OK
        Сохраняю представление        TABLEPRIVILEGES        OK
        Сохраняю представление        COLUMNS                OK
        Сохраняю представление        TABLESTATISTICS        OK
        Сохраняю представление        PRIMARY_KEYS        OK
        ...
Сохраняю синонимы...
        Сохраняю синоним  TABLES                OK
        Сохраняю синоним  ALL_OBJECTS                OK
        Сохраняю синоним  ALL_CATALOG                OK
```


© Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

```
        Сохраняю синоним  DBA_OBJECTS                OK
        Сохраняю синоним  USER_USERS                OK
        ...
Роли не сохраняются.
Привилегии не сохраняются.
Триггеры не сохраняются.
Процедуры не сохраняются.
Последовательности не сохраняются.

                *** С Т А Т И С Т И К А ***

Пользо-Табли- Пред-  Сино-  Приви- Триг- Проце- Роли После- Расши-
 Собы- Пере-
ватели цы     ставл. нимы   легии  геры  дуры        довat. рения
 тия   менн.
------ ------ ------ ------ ------ ----- ------ ---- ------ ------
 ----- -----
0      48     35     27     0      0     0      0    0      0
 0     0

        РАБОТА ЗАВЕРШЕНА.
```

Формат экспортированного файла данных

Файл данных, генерируемый в процессе экспорта данных базовой таблицы, имеет следующий формат:

[1] <формат>::= <строка таблицы>[<LF><строка таблицы>…]
[2] <строка таблицы>::= <p> <поле данных> [<поле данных> …]
[3] <поле данных>::= <значение столбца> <p>
[4] <LF>::= переход на следующую строку
[5] <p>::= символ разделителя (по умолчанию запятая)
[6] <значение столбца>::=
символьное представление значения столбца таблицы

Общие правила

1) Разделители устанавливаются в начале и в конце <строки таблицы>, а также между данными столбцов.
2) Под <строкой таблицы> подразумевается строка базовой таблицы. В файле экспорта <строка таблицы> физически может быть представлена в виде нескольких строк, если данные текстовых столбцов содержат код перевода строки.
3) Утилита перед экспортом каждой строки таблицы проверяет, не содержатся ли среди экспортируемого текста символы разделителя по умолчанию (запятая). Если да, то подбирается другой символ разделителя, который не содержится в экспортируемом тексте данной строки. Таким образом, в сгенерированном файле могут оказаться строки с разными символами разделителей. По этой причине настоятельно не рекомендуется производить выгрузку из столбцов с типом данных `BYTE` и `VARBYTE` в виде символов, а не шестнадцатеричных значений.

В таблице 3 представлены форматы выгрузки всех типов данных СУБД ЛИНТЕР.

Таблица 3. Форматы выгрузки данных

Тип столбца	Возможные ключи	Пример	Комментарий
`CHAR, VARCHAR`		Hello	Представление в виде символов
`INTEGER, SMALLINT, BIGINT`		123456	Символьное представление числового значения
`BOOLEAN`		true, false	Фраза `true` или `false`
`BYTE, VARBYTE`		48656C6C6F	Символьное представление шестнадцатеричных цифр
`BYTE, VARBYTE`	`-g`	Hello	Символьное представление байтов
`NUMERIC`		123456789012345678 90.1234	Символьное представление числового значения
`DATE`	`-dus`	06.03.2003	Символьное представление даты в формате дд.мм. гггг.[чч : [ми: [сс: [тт]]]]

© Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

Тип столбца	Возможные ключи	Пример	Комментарий
			Default-значение даты выгружается в формате дд.мм.гггг
DOUBLE		+1.23456789012345e+009	Символьное представление числового значения
NCHAR, NCHAR VARYING		320438043A0442043 E04400438044F04	Шестнадцатеричное представление Unicode-символов. Одному Unicode-символу соответствует 4 цифры (два байта)
NCHAR, NCHAR VARYING	-g	Виктория	Данные выгружаются в виде символов в 866-кодировке
REAL		123.456	Символьное представление числового значения
BLOB		1 ALLTYPE.blb \00000001.002	Тип BLOB-данных и спецификация файла, содержащего экспортированные BLOB-данные
EXTFILE		/usr/linter/PDF/ DocFiles \LINTER.DOC? DOCRTF2TEXT	Спецификация экспортированного внешнего файла. До знака «?» указан путь к файлу на диске. После знака «?» – имя фильтра, который должен использоваться при загрузке данного внешнего файла

Во всех типах данных для представления NULL-значений по умолчанию применяется строка <NULL> (данное представление можно изменить с помощью ключа -p).

При указании ключа -sf схемы и данные экспортируются в виде SQL-запросов в единый текстовый файл. Перед запросами на создание объектов БД записываются комментарии с указанием типа объекта, владельца объекта и имени объекта, например:

```
-- Type: Table; Owner: "SYSTEM"; Name: "AUTO"
```

После SQL-запроса на создание таблицы следует комментарий на загрузку данных, например:

```
-- Data for table: "AUTO"
```

Коды завершения

Коды завершения утилиты `dbstore` приведены в таблице 4.

Таблица 4. Коды завершения утилиты `dbstore`

Код	Причина
31001	Ошибка открытия файла
31002	Некорректный аргумент
31003	Невозможно создать каталог
31004	Недостаточно памяти
31005	Невозможно подобрать разделитель для полей выгружаемых данных, то есть, скорее всего, данные строки таблицы содержат все символы алфавита
31006	Файл уже существует
31007	Достигнут конец файла
31008	Прерывание по комбинации клавиш <Ctrl>+<Break>
31009	Ошибка в формировании XML-файла
31010	Ошибка записи на диск (возможно, диск заполнен)

Все остальные коды завершения соответствуют кодам завершения ядра СУБД ЛИНТЕР.

Нулевой код завершения означает успешное окончание операции. Если в БД есть таблица `ERRORS`, то дополнительно к числовому коду завершения выдается и его текстовое описание.

© Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026

Приложение
Примеры использования утилиты

Пример П.1.

Полный экспорт БД в каталог FULL_DB с интерактивным вводом паролей пользователей БД.

1) Выполнить команду:

```
dbstore -u SYSTEM/MANAGER8 -d FULL_DB
```

2) В процессе экспорта БД утилита будет запрашивать ввод пароля пользователя, данные из таблиц которого в данный момент должны быть экспортированы, например:

```
Пользователь: Alex

Пароль:
```

3) Для продолжения экспорта БД ввести запрашиваемый пароль. Если пароль неизвестен, нажать клавиши **<Ctrl>+<Break>** для завершения работы утилиты.

Пример П.2.

Полный экспорт БД в каталог FULL_DB с пакетным вводом паролей пользователей БД.

1) Получить информацию о паролях всех пользователей БД и сформировать текстовый файл регистрационных данных пользователей экспортируемой БД, например:

   файл pasw.txt:

```
BOLT/bolt
KAS/kas
Alex/76bdFF
```

2) Выполнить команду:

```
dbstore -u SYSTEM/MANAGER8 -d FULL_DB -plist pasw.txt -ep
```

В результате все необходимые пароли будут искаться в файле pasw.txt. Если для некоторого владельца пароль в файле pasw.txt не найден, то для него будет использоваться пустой пароль (т.к. задан ключ -ep). Естественно, что данный пользователь должен быть зарегистрирован в БД с пустым паролем, в противном случае фиксируется ошибка доступа к данным.

Пример П.3.

Экспорт всех таблиц БД, начинающихся на букву B и заканчивающихся на букву Z пользователя <Kozma Prutkov> в каталог FULL_DB:

```
dbstore -u "Kozma Prutkov"/MANAGER8 -d FULL_DB -ot B%Z
```

Пример П.4.

Экспорт всех таблиц с двухсимвольными именами, начинающимися на букву G в каталог FULL_DB. Пароль пользователя содержит символ пробела и регистрозависимые символы:

```
dbstore -u SYSTEM/"34e3 rT5;ty" -d FULL_DB -ot G_
```

Пример П.5.

Экспорт в файл `errors.txt` только данных таблицы `ERRORS`:

```
dbstore -u SYSTEM/MANAGER8 -l ERRORS -f errors.txt
```

Файл `errors.txt` будет создан в том же каталоге, в котором выполнен запуск утилиты.

Пример П.6.

Экспорт всех хранимых процедур и триггеров БД:

```
dbstore -u SYSTEM/MANAGER8 -otrig -op
```

Пример П.7.

Полный экспорт БД в режиме пакетной выгрузки данных с записью всех BLOB-данных в один файл для последующего импорта БД в режиме медленной загрузки:

```
dbstore -u SYSTEM/MANAGER8 -oMFN
```

Пример П.8.

Экспорт всех привилегий доступа в XML-формате из БД на ЛИНТЕР-сервере «HOME»:

```
dbstore -u SYSTEM/MANAGER8 -n HOME -oa -xml
```

Пример П.9.

Экспорт структуры и данных всех таблиц пользователя <BOLT> без экспортирования внешних файлов:

```
dbstore -u SYSTEM/MANAGER8 -ot -w BOLT -ex
```



Примечание
Внешние файлы, если они есть в таблице, не экспортируются.

© Экспорт структур и данных. АО НПП «РЕЛЭКС», 1990-2026