

МОБИЛЬНАЯ
РЕЛЯЦИОННАЯ
СУБД **ЛИНТЕР**[®]

Linter Standard
Linter Bastion
Linter RealTime
Linter Multiversion

**Запуск и останов СУБД на
платформе UNIX, QNX**

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ

 **РЕЛЭКС**[®]

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 часть данного документа 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

Предисловие	1
Назначение документа.....	1
Для кого предназначен документ.....	1
Необходимые предварительные знания.....	1
Принятые обозначения и соглашения.....	1
Переменные среды окружения	3
Запуск СУБД	4
Переменные среды окружения	4
Порядок запуска.....	5
Ключи командной строки запуска СУБД ЛИНТЕР.....	6
Значения размеров очередей, задаваемые по умолчанию.....	14
Останов СУБД	15
Пользовательский останов ядра.....	15
Системный останов ядра.....	16
Синхронизация завершения работы СУБД	17
Коды завершения	18
Коды завершения ядра СУБД	18
Коды завершения программы останова СУБД.....	19
Приложение 1. Описание файла linter.log	20
Приложение 2. Описание файла linter.out	22
Указатель ключей	24

Предисловие

Назначение документа

Документ содержит описание процедуры запуска и останова СУБД ЛИНТЕР в среде ОС Unix.

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

Для кого предназначен документ

Документ предназначен для системных администраторов и профессиональных пользователей СУБД ЛИНТЕР.



Необходимые предварительные знания

Для запуска и останова СУБД ЛИНТЕР необходимо уметь работать в операционной системе на уровне простого пользователя.

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 все физические файлы.	Выделение в тексте
Подчеркнутый шрифт	Подробную информацию о работе программы можно получить на сайте www.dmk.ru .	Адреса страниц Internet
Текст, разделенный знаком ⇒	Выполните команду View ⇒ Properties (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


Предисловие

<u>Обозначение</u>	<u>Пример</u>	<u>Значение</u>
Символ ::=		Равенство по определению. Слева от знака стоит определяемое понятие, справа – собственно определение понятия
Квадратные скобки []	DBSTORE [-d -r -t -u]	Необязательные элементы конструкции. В данном примере ключи не являются обязательными элементами команды
Вертикальная черта	<return value> ::= <value expression> NULL	Указывает на то, что все предшествующие ей элементы списка являются необязательными и могут быть заменены любым другим элементом списка после этой черты
Фигурные скобки { }	CODEPAGE {866 1251 KOI8}	Указывают на то, что все находящееся внутри них является единым целым
Многоточие «...»	Характеристики столбца MAKE CHAR(20) MODEL CHAR(20) ... SQL>	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	 Если конфигурация страницы-шаблона не учитывала свойств, команда будет выполнена некорректно.	Примечание


Переменные среды окружения

Ядро и некоторые компоненты СУБД ЛИНТЕР используют для настройки следующие переменные среды окружения:

- 1) SY00 – указывает путь к файлам системной БД. По умолчанию используется текущий каталог;
- 2) LINTER_MBX – указывает имя почтового ящика для обмена ядра с клиентскими приложениями. Значение этой переменной должно быть числовым в диапазоне от 1 до 65535. Оно не должно совпадать ни с одним из значений для существующих почтовых ящиков или других механизмов обмена в ОС типа Unix – разделяемой памяти и семафора. По умолчанию – 20561;
- 3) LINTER_CP – определяет кодовую страницу на стороне клиента. В эту кодовую страницу будут преобразовываться все строковые ответы СУБД ЛИНТЕР, и, соответственно, все запросы от клиента будут перекодироваться из кодировки клиента во внутренний формат СУБД ЛИНТЕР. Доступные значения – 866, 1251, koI8. По умолчанию для ОС типа Unix – koI8. Но так как интерфейс всех компонентов СУБД ЛИНТЕР использует кодировку 866, то рекомендуется устанавливать эту переменную в 866;
- 4) LINTER_NET – аналогично с LINTER_MBX, но для сетевого клиента;
- 5) LINTER_TMP - определяет путь к каталогу временных файлов;
- 6) LINTER_EDIT – определяет местонахождение редактора скриптов, используемого программой inl;
- 7) LINTER_INLDATEOUT. – определяет формат выдачи даты на экран видеотерминала при выполнении SQL-запросов программой inl;

 Реально размер выходной переменной типа DATA в запросе не может быть больше 22 символов. Результирующее значение хранимой процедуры (типа «дата») не более 40 символов.

- 8) LINTER_INLDEFCONNMODE.- определяет режим открытия канала при работе с программой inl:
 - optimistic;
 - pessimistic;
 - exclusive.

 Наименование режимов регистрозависимо.

Запуск СУБД

Переменные среды окружения

В своей работе ядро и утилиты СУБД ЛИНТЕР используют переменные среды окружения, которые могут быть установлены средствами операционной системы до запуска ЛИНТЕР:

- 1) **SY00** – значение переменной определяет путь до каталога основных файлов базы данных (файлов системных таблиц). По умолчанию в качестве значения переменной используется текущий каталог ОС. Этот путь может быть задан также из командной строки при запуске ядра;
- 2) **LINTER_EDIT** – переменная, определяющая текстовый редактор, используемый для редактирования, например, SQL-запросов (в утилите inl);

 Используется утилитами СУБД ЛИНТЕР.

3) **LINTER_MBX** – переменная, определяющая имя «почтового ящика» для обмена данными между ядром и клиентскими приложениями. Под «почтовым ящиком» следует понимать межпроцессный механизм обмена между ядром ЛИНТЕР и приложением. Используется для передачи данных от приложения ядру ЛИНТЕР и обратно. Изменение значения этой переменной может быть использовано для запуска нескольких ядер на одной ЭВМ.

Значение **LINTER_MBX** должно быть уникальным для данного ядра ЛИНТЕР и приложений, работающих с этим ядром. Значение переменной по умолчанию: Linter.

4) **LINTER_CP** – определяет кодировку данного клиентского приложения. Ядро СУБД ЛИНТЕР будет работать с этим приложением в кодировке, определяемой переменной **LINTER_CP**. Для разных приложений, даже на одной машине, могут быть заданы различные значения переменной **LINTER_CP**.

Для СУБД ЛИНТЕР версии 5.9 и ниже осуществляется поддержка только русскоязычных кодовых страниц, поэтому переменная **LINTER_CP** может принимать одно из значений 866, 1251, koï8 (регистрозависимо).

СУБД ЛИНТЕР, начиная с версии 6.0 и выше, поддерживает однобайтовые, многобайтовые кодировки, UNICODE, UTF8, а также предоставляет возможность загрузить в БД кодировку, необходимую клиентской задаче. Переменная **LINTER_CP** может принимать значение любой кодировки, загруженной в БД (список кодировок хранится в системной таблице \$\$\$CHARSET).

Если значение переменной **LINTER_CP** не определено, то используется кодировка, соответствующая текущему значению locale.

Для консольных утилит СУБД ЛИНТЕР используется кодировка по умолчанию 866, а для графических утилит - 1251.

Для утилит СУБД ЛИНТЕР версии 5.9 и ниже используется кодировка по умолчанию – KOI8.


9) **NET_MBX** – переменная аналогична по своему назначению переменной **LINTER_MBX**, но для сетевого клиента. Т.е. **NET_MBX** определяет имя «почтового ящика» для обмена данными между приложением и сетевым драйвером клиента. Значение по умолчанию: LinDbc

Порядок запуска

Для запуска СУБД ЛИНТЕР:

1. указать путь к БД одним из следующих способов:
 - установить переменную окружения SY00 так, чтобы ее значение указывало на каталог БД;
 - указать путь при запуске ядра ЛИНТЕР в параметре /BASE (см. ниже);
 - сменить текущий каталог на каталог БД.
2. запустить исполняемый файл `linter` ядра СУБД ЛИНТЕР.

При этом каталог `/bin` должен быть включен в список каталогов переменной окружения `PATH`, или для запуска можно использовать полный абсолютный либо относительный путь к исполняемому файлу `linter`.

 Для проверки лицензии используется реальная текущая дата. Если БД была запущена с будущей датой, срок лицензии считается истекшим, даже если вернуть дату назад.

При успешном запуске на консоль выводятся параметры и установленные режимы работы ядра:

```
RDBMS Linter SQL v.6.1.3.1 connected to data base
"DEMO Database      "
POOL holds 89 pages
Table      queue size   : 100
Column    queue size   : 500
Channel   queue size   : 100
File      queue size   : 30
User      queue size   : 100
In-kernel backup table is absent
Auditing disabled
Transaction control is turned on
Sort queue size      : 1
Existing control point(s) in database:
Control point list is empty.
Kernel system parameters: MBX - "20561", Pid - 4819
$ Copyright (C) 1995-2002 Relex, Inc. All rights reserved.
Checking license in progress ... OK
It's a DEMO license.
*** RDBMS Linter is running
*** Press <ENTER> for shell prompt
```

При запуске СУБД ЛИНТЕР активизируются пять задач:

- 1) `linter` – ядро СУБД;
- 2) `sql` – транслятор запросов;
- 3) `intsrt` – процессор сортировки;
- 4) `tsp` – процессор хранимых процедур;
- 5) `loltp` – процессор распределенных запросов (может отсутствовать в списке запущенных задач).

Но, запуская СУБД, пользователь может этого не увидеть, если не обратится к средствам операционной системы за справкой о составе задач, обслуживаемых операционной системой.

Чтобы разрешить удаленный доступ клиентских приложений к данной БД, необходимо выполнить команду:

```
grant access on unlisted station to all;
```

(разрешить доступ к этой БД со всех компьютеров), либо команду на создание станции (разрешить доступ к этой БД только с конкретных компьютеров) (см. документ «Администрирование КСЗ НСД»).

Командная строка запуска СУБД ЛИНТЕР имеет следующий синтаксис:

```
linter [/<параметр> ...]
```

Ключи командной строки запуска СУБД ЛИНТЕР

/POOL=<размер>

Задает размер пула памяти ядра СУБД в страницах по 4 Кбайт.
По умолчанию – 200, минимум –50.

/SPOOL=<размер>

Задает размер пула памяти одного процесса сортировки в страницах по 4 Кбайт.
По умолчанию – POOL/2, минимум – 5.

/PPOOL=<размер>

Задает размер пула подсистемы фразового поиска СУБД в страницах по 4 Кбайт.
По умолчанию – 200, минимум –50.

/INMEMPOOL=<размер>

Задает максимально допустимое количество страниц таблиц “в памяти” в пуле страниц СУБД ЛИНТЕР. Если этот ключ не задан, то использование таблиц “в памяти” запрещено.

/BASE=<путь>

Задает полный путь к БД. Аналог переменной окружения SY00.

/NAME=<строка>

Задает имя почтового ящика для организации работы нескольких ядер ЛИНТЕР на одной ЭВМ. Аналог переменной окружения LINTER_MBX.

/MBX=<параметр>

Задаёт идентификатор межпроцессорного обмена. Значение <параметра> – символьный литерал. Если ключ не задан, используется значение переменной окружения NET_MBX. Если NET_MBX не определена, используется значение по умолчанию.

/WLNB

Позволяет завершить сохранение системного журнала во время останова ядра. В случае если ядро было запущено с этим ключом, утилита архивирования БД (LNB) выполняла операцию сохранения БД в режиме WAIT. Если далее последовала команда на останов ядра (от кого-нибудь), то ядро не завершит работу до тех пор, пока:

- утилита LNB не допишет несохраненную часть системного журнала;
- наступит тайм-аут утилиты LNB (прошло более 10 сек после предыдущей команды от LNB (например, LNB аварийно завершилась));
- прошло 3000 сек. с момента запроса на останов ядра.

В случае если останов ядра последовал в тот момент, когда журнал был уже дописан, ядро прекращает работу безо всяких ожиданий.

/NOOUTFILE

Запрещает печать сообщений в файл `linter.out` (протокол работы СУБД). По умолчанию `linter.out` создается.

/TMPDIR

Определяет местоположение (каталог) для размещения временных файлов.

/PIDFILE=<файл>

Задаёт имя файла, в который будет записан pid ядра. В случае корректного завершения работы СУБД этот файл удаляется.

/[NO]JEXIT

Задаёт режим автоматического завершения работы ядра СУБД в случае невозможности продолжения записи в системный журнал.

По умолчанию – JEXIT.

NOJEXIT заставляет ядро продолжать работать, при этом в случае сбоя ядра журнал не сможет восстановить БД.

/K=<значение>

Заставляет посылать сигнал (<значение>) родительскому процессу при успешной инициализации ядра СУБД ЛИНТЕР.

/TCORRECT

Заставляет игнорировать временное расхождение в случае, когда дата последнего запуска СУБД является «будущей» по сравнению с текущей датой операционной системы.

/JDBCP=<порт>

В этом режиме одновременно с запуском ядра СУБД будет запускаться и сетевой драйвер сервера. Значение <порт> задает сетевой номер порта, на котором будет запущен JDBC-сервер.

Задаёт запуск одновременно с запуском ядра СУБД сетевого драйвера сервера для JDBC-сервера с указанным номером порта.

/JDBCS

Задаёт запуск сетевого драйвера сервера для JDBC-сервера с номером порта по умолчанию.

/TCP=<порт>

Задаёт режим запуска сетевого драйвера сервера, работающего по протоколу TCP/IP, одновременно с запуском ядра. <Порт> – номер порта, по которому будет осуществляться соединение клиента с сетевым сервером.

/SPX

Задаёт режим запуска сетевого драйвера сервера, работающего по протоколу SPX, одновременно с запуском ядра (только для NetWare).

/CHECK

Задаёт режим периодической проверки живучести родительского процесса. Если родительский процесс завершился, ядро также завершает свою работу.

/NOLARGE

При старте СУБД определяет, поддерживает ли операционная система длинные файлы. Если нет, то запрещается стартовать на БД, содержащей такие файлы. Для исключения возникновения подобной ситуации необходимо задавать параметр /NOLARGE, который запрещает создание таблиц размером больше 2 Гбайта.

Если ядро запущено без ключа /NOLARGE, а ОС не поддерживает длинные файлы, то ядро запустится, но не будет работать с таблицами, которые расположены в длинных файлах. При попытке обратиться к таким таблицам будет выдано соответствующее сообщение.

Если ядро запущено с ключом /NOLARGE, и в базе данных присутствуют файлы размером больше 2Гб, то ядро не будет запущено. При попытке расширить файл до

размеров > 2Гб будет возвращен код, сигнализирующий о невозможности это сделать.

Запуск с ключом `/NOLARGE` позволяет пользователю создавать базу данных и работать с ней какое-то время на ОС, поддерживающей длинные файлы, а затем перенести БД на ОС, не поддерживающую длинные файлы.

`/NOOUTPUT`

Запрещает вывод на консоль протокольной информации о работе ядра.

`/OUTFLIMIT=<размер>`

Задаёт размер файла `linter.out` в страницах по 4 Кбайт.

Если в процессе работы размер файла `linter.out` превысит заданный лимит, то он будет переименован в файл `linter.ou1`. После этого информация будет записываться в пустой файл `linter.out`.

`/LOGFLIMIT=<размер>`

`/TRACEFLIMIT =<размер>`

Задают допустимые размеры файлов `linter.log` и `lintrace.log` соответственно. `<Размер>` задается в блоках. Задание размеров файлов приводит к тому, что перед проверкой ограничений пользователей осуществляется проверка размера файлов `linter.log` и `lintrace.log`. Если их размер превышает заданные границы, вся информация из текущих файлов `linter.log` и `lintrace.log` копируется в файлы с маской `linter_YYYYMMDDHH24MISS.log` и `lintrace_YYYYMMDDHH24MISS.log`, где `YYYY` – год, `MM` – месяц, `DD` – день, `HH24` – час, `MI` – минуты, `SS` – секунды.

Вслед за этим открываются пустые файлы с именами `linter.log` и `lintrace.log`, и работа продолжается в обычном режиме вплоть до очередной проверки.

`/DEBUG`

Запрещает процессу ядра переходить в фоновый режим.

`/[NO]LOG`

Запрещает/разрешает вести протокол обработки SQL-запросов (файл `linter.log`).

По умолчанию – `/NOLOG`.

`/LOGQUERY`

Задаёт режим протоколирования обработки SQL-запросов клиентских приложений в файле `linter.log`.



Структура файла `linter.log` описана ниже в Приложении 1.

`/LOGALL`

Задаёт режим полного протоколирования обработки SQL-запросов клиентских приложений в файле `linter.log`. В этом режиме в файл протокола заносится

дополнительная информация: время выполнения команды, сетевой адрес, идентификаторы процесса и нити, пославшие запрос, и др.

Если при запуске ядра указать два ключа (/NOLOG и /LOGALL), то определяющим будет ключ, стоящий последним в строке запуска. Например, пусть строка запуска имеет вид - `linter /NOLOG /LOGALL` - в этом случае ядро будет запущено в режиме полного протоколирования. Другой пример, пусть строка запуска имеет вид - `linter /LOGALL /NOLOG` - ядро не будет вести протокол обработки SQL-запросов.

/NOTSP

Запрещает запускать транслятор триггеров и хранимых процедур (предполагается, что работа с данными объектами БД не потребуются). Эта возможность позволяет использовать ЛИНТЕР в системах с ограниченными ресурсами.

/NOSQL

Запрещает запуск SQL-транслятора.

Эта возможность позволяет использовать ЛИНТЕР в системах с ограниченными ресурсами. При запуске с ключом /NOSQL ядро ЛИНТЕР может работать с уже скомпилированными приложениями, использующими только оттранслированные запросы.

/KILL=<время>

Задаёт временной промежуток (в секундах), через который проверяется «живучесть» программ-клиентов. Если через указанный промежуток времени обнаруживается, что какой-либо из клиентов закончил работу и не известил об этом ЛИНТЕР, то ядро автоматически освободит относящиеся к этому клиенту ресурсы.

По умолчанию – 120 секунд, минимум – 10 секунд.

/TRACE=DECOMP[=FULL]

Задаёт трассировку исполнения SQL-запросов.

Опция FULL указывает на необходимость полной трассировки (по умолчанию выполняется краткая трассировка).

Трассировка содержит следующую информацию:

- текст SQL-запроса, переданный на обработку ядру СУБД ЛИНТЕР (после его оптимизации SQL-транслятором);
- какие массивы данных (битвектора) были задействованы при обработке SQL-запроса. Используемые массивы влияют на время выполнения запроса;
- количество считанных/записанных блоков данных (физических/логических);
- количество считанных/записанных блоков системного журнала.

Пример трассировочной информации

```
C#4 QUERY:
SELECT
  T_0."MSG"
FROM
  <TABLE "SYSTEM"."ERRORS" AS T_0>
```

```

WHERE
  T_0."NMRERR" == 1503;
C#4 DECOMP.C (Start_Cur_Dec): Now computing derived set #0.
C#4 OBRSTRAT.C (OBRSTRAT): Start set: TABLE("SYSTEM"."ERRORS"
AS T_0). Set included 1022 rows.
  List of predicates:
  Predicate [strategy #2(one index)]:
  T_0."NMRERR" == 1503
C#4 DECOMP.C (End_Dekart): Derived set #0 is computed, Rows
count: 1.
C#4 FORMOTW.C (FORMOTW): Read: 0 blocks, write: 0 blocks.
C#4 FORMOTW.C (FORMOTW): Read logical: 4 blocks, write logical:
0 blocks.
C#4 FORMOTW.C (FORMOTW): Journal read: 0 blocks, written: 0
blocks.

```

/TRACE=DECOMP=DELAY[=<тики>]

Задаёт трассировку задержек длительности выполнения отдельных квантов ядра на время, не меньшее указанного в параметре <тики>. По умолчанию трассируются задержки не менее 100 тиков (0.1 сек).

Пример трассировочной информации

```

..
C#3 OBRSTRAT.C (OBRSTRAT): Start set: TABLE("SYSTEM"."LINEITEM"
AS T_1). Set included 6001215 rows.
  List of predicates:
  Predicate [strategy #1(full scan)]:
  T_1."L_SHIPDATE" > '15.03.1995:00:00:00:00'
C#3 PROZA.C (PROZA): Strategy: #1(full scan).
C#3 Delay for 1.87 sec
C#3 Delay for 0.35 sec
C#3 RIDSTRAT.C (RIDSTRAT): Snap_Bv: 3241776 rows.
..

```

/TRACELOG

Задаёт вывод в файл `linter.out` диагностических сообщений об открытии и закрытии соединения с БД (с полной информацией об источнике команды) и ошибок выполнения запросов по соединениям.

/LOCK

Задаёт блокирование выделенной для работы оперативной памяти ядра (под очереди системных объектов, пул страниц и т.п.) на уровне операционной системы. В случае если память заблокировать невозможно, ядро не запускается. В этом режиме за счёт отсутствия вытеснения памяти ядра другими программами ускоряется его работа. Пользоваться этой возможностью нужно с осторожностью, чтобы не

захватить практически всю физическую память и, таким образом, не повлиять отрицательно на общую производительность системы.

/RO

Задаёт работу с БД в режиме «только чтение» (модификация БД в этом режиме невозможна). При запуске СУБД с этим ключом доступны следующие команды:

SELECT	SET EVENT
EXECUTE PROCEDURE	CLEAR EVENT
TEST TABLE	GET COMPLEX EVENT
LOCK TABLE	WAIT COMPLEX EVENT
UNLOCK TABLE	GET LINTER DUMP
CREATE EVENT	SET LOG
DROP EVENT	SET NAMES (для ЛИНТЕР 6.0 и выше)
GET EVENT	SET SESSION BLOB LOG
WAIT EVENT	SET SORTPOOL LIMIT/UNLIMITED

При работе в режиме `read only` в каталоге временных файлов создается ряд файлов (файлы логирования, трассировки, временные рабочие файлы): `linter.out`, `linter.log`, `lintrace.log`, `phrase.idx`, 1.31, 1.41, 1.51.

При завершении работы СУБД ЛИНТЕР перечисленные выше файлы удаляются.

Каталог временных файлов определяется следующим образом (по убыванию приоритета):

1) может быть задан в явном виде в параметрах запуска ядра с помощью ключа `TMPDIR`, например (для Win32):

```
linter.exe /local /base=D:\database /RO /TMPDIR=D:\TMP
```

2) если ядро СУБД под ОС Unix имеет версию 6.0 и выше, путь к каталогу временных файлов берется из переменной среды окружения `LINTER_TMP`, а если она не задана, то по умолчанию устанавливается `/tmp/linter`.

/ANALYZE


Оптимизатор СУБД ЛИНТЕР будет анализировать все SQL запросы с точки зрения существования индексов, позволяющих ускорить их обработку. Список рекомендуемых индексов будет выведен на экран консоли ядра и в файл `linter.out` (только для версий 5.9 и старше).

/AUTOINDEX

Разрешает автоматически создавать необходимые индексы для ускорения обработки SQL-запросов. Созданные индексы сохраняются в БД (только для версий 5.9 и старше).

`/PASS=<спецификация файла>`

Задаёт ввод пароля доступа к шифрованной БД из указанного текстового файла. Файл должен содержать строку пароля доступа.

 В некоторых ОС для символа \$ требуется экранирование при указании в командной строке.

Пример

```
linternt.exe /local /pass=$gost$123456
```

`/SETPASS`

Задаёт интерактивный ввод пароля доступа к шифрованной БД. Пароль доступа надо будет ввести в предложенном диалоговом окне или в строке ввода.

Пример

```
linternt.exe /local /setpass
```

`/PASSFILE`

Задаёт путь к текстовому файлу, содержащему пароли доступа к шифрованной БД.

Пример

```
linternt.exe /local /passfile=d:\pass.txt
```

Содержимое файла `pass.txt` - строка:
`$gost$123456`

`/U=<имя>/<пароль>`

Задаёт регистрационные данные пользователя БД. При старте ядра СУБД ЛИНТЕР будет проверяться наличие данного пользователя. Если пользователь зарегистрирован в БД, является её администратором (имеет категорию доступа DBA) и принадлежит к нулевой группе, то ядро СУБД будет запущено; в противном случае происходит отказ в запуске ядра.

`/RAPID`

Задаёт режим «догона» системного журнала, развернутого из архивного файла БД. Используется при запуске СУБД ЛИНТЕР в системе резервирования.

`/VERSION`

Заставляет выводить на консоль информацию о версии ядра СУБД ЛИНТЕР.

Значения размеров очередей, задаваемые по умолчанию

По умолчанию при запуске ядра СУБД ЛИНТЕР устанавливаются следующие значения:

- длина или размер очереди, содержащей описания таблиц – 100;
- длина или размер очереди, содержащей описания столбцов – 500;
- длина или размер очереди файлов – 30;
- количество каналов для связи приложений с ядром СУБД ЛИНТЕР – 100;
- квант обработки записей - число записей, просматриваемых системой без прерывания при обработке одного запроса – 10;
- квант обработки индексов - число индексов, просматриваемых без прерывания при обработке одного запроса – 1.

Останов СУБД

Пользовательский останов ядра

Останов осуществляется специальной утилитой, которая подает команду останова ядру СУБД ЛИНТЕР и ожидает, пока оно завершится.

Для останова СУБД ЛИНТЕР используется команда:

```
shut [<командная строка>]
```

```
<командная строка>::=[<имя>] [<пароль>][<ЛИНТЕР-сервер>]
```

или

```
<командная строка>::=[-u имя/пароль][-n <ЛИНТЕР-сервер>][-r]
```

или

```
<командная строка>::=-version | -r |-h
```

Ключи утилиты

-u

Задаёт регистрационные данные (имя и пароль) пользователя БД с правами администратора БД.

-n

Задаёт имя узла удаленного ЛИНТЕР-сервера (должно присутствовать в файле nodetab). Если этот параметр не задан, то команда применяется к локальному узлу или к узлу по умолчанию.

-r

Заставляет выполнить немедленное завершение работы ядра, не зависимо от наличия активных транзакций в данный момент. В случае наличия таких транзакций они будут откачены, а пользователи извещены о принудительном останове ядра. По команде останова ядра при наличии активных транзакций ядро не будет остановлено, и утилита shut получит код возврата 1012.

-version

Выдает информацию о версии программы shut.

-h

Выдается справочная информация о программе.

Системный останов ядра

Системный останов ядра СУБД ЛИНТЕР инициируется при останове (shutdown) ОС.

В этом случае работа ядра завершается без уведомления пользователя. Реально останов ОС произойдет после полного завершения ядра или через 100 секунд после начала выполнения команды `shut` для СУБД ЛИНТЕР.

Синхронизация завершения работы СУБД

Для синхронизации завершения работы СУБД ЛИНТЕР с ОС типа Unix используется программа `lsyncd`. Данная программа передает управление ОС после полного завершения работы ядра СУБД. Обычно она применяется совместно с программой `shut` в скриптах на закрытие ОС.

Запуск программы:

```
lsyncd [-t <тайм-аут>]
```

<Тайм-аут> – тайм-аут (в секундах) ожидания завершения работы ядра СУБД ЛИНТЕР. Если параметр не задан, по умолчанию используется значение 30 сек.

Программа `lsyncd` предназначена для определения момента фактического завершения работы ядра СУБД ЛИНТЕР. При подаче команды `shut` с использованием программы `shut` или из прикладной программы пользователем, имеющим привилегии на выполнение этой операции, происходит только инициирование начала процесса завершения работы. Ответ ядра на эту команду немедленно передается пославшей ее программе, и она имеет возможность сообщить об успешном начале процесса завершения работы ядра СУБД. Однако сам процесс останова СУБД занимает определенное время. Для определения реального момента завершения работы ядра СУБД ЛИНТЕР служит программа `lsyncd`.

После запуска данной программы она переходит в состояние ожидания (об этом свидетельствует выводимая ею строка `wait for LINTER shutdown`) и находится в этом состоянии до тех пор, пока ядро СУБД ЛИНТЕР существует как задача ОС. По окончании процесса завершения работы ядра СУБД ЛИНТЕР программа выводит на экран строку `complete` и завершает свою работу.

Если при запуске `lsyncd` ядро СУБД ЛИНТЕР не активно, программа завершает свою работу немедленно с выдачей соответствующего сообщения.

Так же, как и для обычных клиентов, для перенаправления слежения за другим ядром можно использовать переменную окружения `LINTER_MBX`. Ее значение должно совпадать со значением этой переменной при запуске ядра СУБД ЛИНТЕР. Программа `lsyncd` может применяться только на том компьютере, на котором запущено выполнение ядра СУБД ЛИНТЕР. Работа по сети невозможна.

Программа `lsyncd` должна выполняться пользователем ОС, который запустил ядро СУБД ЛИНТЕР, или суперпользователем ОС.

Для `irc`-версии СУБД ЛИНТЕР проверка активности ядра производится каждую секунду.

Для `socket`-версии завершение работы программы происходит немедленно после завершения работы ядра СУБД ЛИНТЕР.

Если по истечении тайм-аута ядро СУБД ЛИНТЕР еще активно, происходит окончание работы `lsyncd` с кодом завершения -1. Если же ядро не загружено или в процессе работы `lsyncd` уже завершило свою работу, код завершения будет равен 0.

Коды завершения

Коды завершения ядра СУБД

Если во время запуска ядра СУБД ЛИНТЕР происходит сбой, то анализируется причина сбоя, и работа ядра завершается. При этом выдается код завершения.

LINTER_EXIT_CODE_NO_ERR 0x0000
Успешное завершение работы.

LINTER_EXIT_CODE_INCORR_VERSION 0x0001
Версия ядра ЛИНТЕР не соответствует версии базы данных, на которой его пытаются запустить.
Воспользуйтесь утилитой импорта-экспорта данных (migration), которая входит в состав СУБД ЛИНТЕР. Она позволяет перемещать базы данных между различными версиями СУБД ЛИНТЕР.

LINTER_EXIT_CODE_DATABASE_CRASH 0x0002
Повреждение структуры базы данных. Необходимо запустить утилиту тестирования и восстановления (testdb) для БД. А затем запустить на ней ядро ЛИНТЕР еще раз.

LINTER_EXIT_CODE_NO_MEMORY 0x0003
Не хватает оперативной памяти. Необходимо либо уменьшить размер пула ядра, либо увеличить количество памяти.

LINTER_EXIT_CODE_LOW_POOL 0x0004
Размер пула ядра не достаточен для работы. Необходимо увеличить размер пула ядра.

LINTER_EXIT_CODE_LOW_SRT_POOL 0x0005
Размер пула сортировки не достаточен для работы. Необходимо увеличить размер пула сортировки.

LINTER_EXIT_CODE_ALREADY_RUNNING 0x0006
Ядро ЛИНТЕР на выбранной базе данных уже запущено. Чтобы запустить несколько ядер ЛИНТЕР на одной БД, необходимо использовать параметр /NAME или переменную окружения LINTER_MBX.

LINTER_EXIT_CODE_DATABASE_PERM 0x0007
У пользователя нет прав для запуска ЛИНТЕР на указанной базе данных.

LINTER_EXIT_CODE_CANT_WRITE_TO_DISK 0x0008
Невозможно продолжать запись в системный журнал (ядро ЛИНТЕР запущено с ключом /JEXIT). Такая ситуация может возникнуть в двух случаях: либо не хватает места на диске, либо заполнены все файлы журнала.
Количество файлов журнала и размер каждого из них задаются в свойствах БД (либо с помощью утилиты «Администратор СУБД ЛИНТЕР», либо с помощью утилиты «Создание и конфигурирование базы данных» (gendb))

LINTER_EXIT_CODE_INVTIME 0x0009

Дата последнего запуска СУБД (в GMT) является «будущей» по сравнению с текущей датой операционной системы. Нужно запускать ядро ЛИНТЕР с параметром /TCORRECT до тех пор, пока даты не сравняются.

LINTER_EXIT_CODE_NO_ENV_TEMP 0x000A

Для БД, запущенной в режиме только для чтения (/RO), невозможно определить временный каталог или задано неверное значение переменной TEMP.

LINTER_EXIT_CODE_DATABASE_LOCKED 0x000B

СУБД не может заблокировать индексный файл системной таблицы \$\$\$SYSREL. Указанная БД уже заблокирована другим приложением (возможно, на ней уже запущена СУБД ЛИНТЕР).

LINTER_EXIT_CODE_USER_NAME_PASS 0x000C

Неверное имя пользователя или пароль в ключе -u ..

Коды завершения программы останова СУБД

Код завершения	Описание
0	Успешное завершение.
1	Неполные или неверные аргументы.
2	Неверные имя пользователя/пароль или нет привилегий для завершения работы ядра.
3	Ядро СУБД или сетевой драйвер не загружены, не найден указанный сервер.
4	Неизвестная ошибка при обращении к СУБД ЛИНТЕР.
5	Ядро СУБД выполняет другие запросы.

Приложение 1

Описание файла linter.log

В приложении рассматриваются примеры информации, которая заносится в файл протоколирования обработки SQL-запросов пользователя (файл linter.log), и приводится расшифровка данной информации.

Файл linter.log является перезаписываемым, т.е. при запуске ядра (с параметром, разрешающим ведение протокола) все сообщения о предыдущем сеансе работы СУБД будут удалены из файла.

Пример 1. Пример файла linter.log в режиме краткого протоколирования.

```
?DESC:L=92:
!:E=0:C=0:
?OPEN:U="SYSTE<":P=16384:R=0:
!:E=1025 @&#:C=0:
?OPEN:U=H:P=16384:R=0:
!:E=1025 @&#:C=0:
?OPEN:U="SYSTEM":P=16384:R=0:
!:E=0:C=3:
?DESC:L=92:
!:E=0:C=3:
?DESC:L=92:
!:E=0:C=3:
?OCUR:C=3:P=16384:R=0:
!:E=0:C=4:
?SLCT:C=4:L=65535:P=0:
select * from "SYSTEM"."$$$AUDIT";
!:E=0:C=4:A=1:
?GETA:C=4:L=0:
!:E=0:C=4:
?KILL:C=3:U=H:I=4:
!:E=0:C=3:
?CLOS:C=4:
!:E=1069 @&#:C=4:
OCUR:C=3:P=16384:R=0:
!:E=0:C=4:
?:C=4:L=65535:P=0:
execute "SYSTEM"."SAMPLE"(FALSE,FALSE,1);
#?OCUR:C=4:P=34:R=0:
#!:E=0:C=5:
#?:C=5:L=0:P=0:
drop table results;
#!:E=0:C=5:
#?:C=5:L=0:P=0:
create table results(lin char(100));
#!:E=0:C=5:
```

Пример 2. Пример файла linter.log в режиме полного протоколирования.

```
?SLCT:T=10:26:25.370:XPid=1732:XTid=1632:C=5:L=65535:P=0:
select * from auto;;;
!:E=0:T=10:26:25.450:XPid=1732:XTid=1632:C=5:A=1000:
?GETA:T=10:26:25.450:XPid=1732:XTid=1632:C=5:L=0:
!:E=0:T=10:26:25.450:XPid=1732:XTid=1632:C=5:
?GETA:T=10:26:25.450:XPid=1732:XTid=1632:C=5:L=3914:
!:E=0:T=10:26:25.460:XPid=1732:XTid=1632:C=5:
?OCUR:T=10:26:25.460:XPid=1732:XTid=1632:C=3:P=16384:R=0:
!:E=0:T=10:26:25.460:XPid=1732:XTid=1632:C=6:
?GETS:T=10:26:25.480:XPid=1732:XTid=1632:C=5:I=1:L=113:
!:E=0:T=10:26:25.480:XPid=1732:XTid=1632:C=5:
```

```
?GETS:T=10:26:25.480:XPid=1732:XTid=1632:C=5:I=1:L=113:
!:E=0:T=10:26:25.480:XPid=1732:XTid=1632:C=5:
```

Знак **?**, расположенный в начале строки, показывает, что информация относится к обработке запроса. Затем идёт команда из блока CBL интерфейса нижнего уровня (например, OCUR, SLCT, DESC и т.п.). Если ведётся полное протоколирование работы, то будет показана также дополнительная информация: время выполнения запроса, сетевой адрес, идентификаторы процесса и нити, пославшие запрос. Далее идут параметры команд из блока CBL, которые описаны ниже. Затем (на следующей строке) расположены данные, которые необходимы ядру СУБД для выполнения команды (если они требуются): текст запроса, данные для загрузки в BLOB и т.д.

Знак **!**, расположенный в начале строки, указывает на то, что это информация об ответе ядра на запрос пользователя. Она включает в себя код возврата (:E=«код возврата») и признак ошибки (константа @&# , а затем параметры блока CBL), если была ошибка. Например, строка, сообщающая об ошибке при выполнении запроса, может выглядеть так !:E=1025 @&#:C=0: (см. пример 1).

Знак **#** в начале строки (перед ? или !) – признак того, что запрос подан из триггера или хранимой процедуры.

Параметры блока CTBL:.

- C – номер канала
- L - длина буфера ответа (поле LnBufRow)
- P – режим обработки команды внутреннего интерфейса
- R – приоритет канала
- U – имя пользователя/пароль
- I – внутренний системный номер записи, которая была обработана последней (RowID)
- K – значение параметра, необходимого при обработке запроса. Например, номер BLOB поля.
- S - размер запроса или обработанных данных.
- O – код ошибки, переданный СУБД операционной/сетевой средой при обработке запроса (SysErr)
- A – количество реально обработанных записей (RowCount)

Приложение 2

Описание файла linter.out

В приложении приводится пример информации, содержащейся в файле сообщений ядра СУБД ЛИНТЕР (linter.out).

В этот файл заносится информация о таких параметрах запуска ядра, как размер очередей (Table, Column, Channel, File, User), размер пула ядра (POOL), количество процессов сортировок; а также значение переменной LINTER_MBX, состояние системных таблиц БД (Devices, Charset, Security, Procedures и т.д.). Сюда же вносятся предупреждения и сообщения об ошибках, возникающих в процессе работы СУБД.

Файл сообщений является дополняемым, т.е. при последующих запусках СУБД сообщения будут дописываться в конец файла linter.out.

При старте ядра СУБД ЛИНТЕР на уже созданной базе данных в файл linter.out заносятся сообщения о дате создания БД и датах последнего запуска и останова базы данных:

```
Database creation time:      22.10.2003 06:26:57.00
Last database startup time: 12.11.2003 08:59:02.05
Last database shutdown time: 12.11.2003 10:12:45.17
```

При завершении работы СУБД в файл linter.out заносится сообщение:

```
*** RDBMS Linter has been shut down ***
```

Пример файла linter.out.

```
22.10.2003 06:26:57 GENDB-I-CRFILE, Создается системная база данных ...
GENDB-I-SUCCRD,база данных создана
Database creation time:      22.10.2003 06:26:57.00
Last database startup time:  Unknown
Last database shutdown time: Unknown
22.10.2003 06:26:59Linter SQL v. 6.1.5.5 connected to data base  "DEMO DATABASE
"
22.10.2003 06:26:59 POOL holds 853 pages
22.10.2003 06:26:59 Table      queue size  : 100
22.10.2003 06:26:59 Column    queue size  : 500
22.10.2003 06:26:59 Channel   queue size  : 100
22.10.2003 06:26:59 File      queue size  : 30
22.10.2003 06:26:59 User      queue size  : 100
22.10.2003 06:26:59 Procedures table is absent
22.10.2003 06:26:59 Procedures Dictionary table is absent
22.10.2003 06:26:59 Security groups table is absent
22.10.2003 06:26:59 Security levels table is absent
22.10.2003 06:26:59 Devices table is absent
22.10.2003 06:26:59 Stations table is absent
22.10.2003 06:26:59 Replication rules table is absent
22.10.2003 06:26:59 Trigger table is absent
22.10.2003 06:26:59 Filter table is absent
22.10.2003 06:26:59 File extension table is absent
22.10.2003 06:26:59 Sequence table is absent
22.10.2003 06:26:59 Charset table is absent
22.10.2003 06:26:59 Translate table is absent
22.10.2003 06:26:59 Errors table is absent
22.10.2003 06:26:59 In-kernel backup table is absent
22.10.2003 06:26:59 Auditing disabled
22.10.2003 06:27:00 Transaction control is turned on
22.10.2003 06:27:00 Max concurrent sorting processes : 1
```

```

Existing control point(s) in database:
Control point list is empty.
22.10.2003 06:27:00 Kernel system parameters: MBX - "615", Pid - 364
22.10.2003 06:27:00 Copyright (C) 1995-2003 Relex, Inc. All rights
reserved.22.10.2003 06:27:00
22.10.2003 06:27:00 Fail! NO LICENSE FOUND!
22.10.2003 06:27:00 *** RDBMS Linter is running
22.10.2003 06:27:00 *** Press <ENTER> for shell prompt
22.10.2003 06:27:17 W-145 (OPENTH) : Invalid charset: CP437
22.10.2003 06:27:18 E-150 (CHARSET) : Charset #0 is used
22.10.2003 06:27:24 *** Linter is coming down ***
22.10.2003 06:27:24 *** PP processing ***
22.10.2003 06:27:24 *** Table queue processing ***
22.10.2003 06:27:24 *** Column queue processing ***
22.10.2003 06:27:24 *** User queue processing ***
22.10.2003 06:27:24 *** File queue processing ***
22.10.2003 06:27:24 *** RDBMS Linter has been shut down ***
Database creation time:      22.10.2003 06:26:57.00
Last database startup time:  12.11.2003 08:59:02.05
Last database shutdown time: 12.11.2003 10:12:45.17
12.11.2003 10:13:21 Linter SQL v. 6.1.5.5 connected to data base "DEMO DATABASE
"
12.11.2003 10:13:21 POOL holds 357 pages
12.11.2003 10:13:21 Table      queue size : 100
12.11.2003 10:13:21 Column    queue size : 500
12.11.2003 10:13:21 Channel  queue size : 100
12.11.2003 10:13:21 File      queue size : 29
12.11.2003 10:13:21 User      queue size : 100
12.11.2003 10:13:21 In-kernel backup table is absent
12.11.2003 10:13:21 Auditing disabled
12.11.2003 10:13:21 Transaction control is turned on
12.11.2003 10:13:21 Max concurrent sorting processes : 1
Existing control point(s) in database:
Control point list is empty.
12.11.2003 10:13:21 Kernel system parameters: MBX - "", Pid - 1676
12.11.2003 10:13:21 Copyright (C) 1995-2003 Relex, Inc. All rights
reserved.12.11.2003 10:13:21
12.11.2003 10:13:21 It's a DEMO license.
12.11.2003 10:13:21 *** RDBMS Linter is running
12.11.2003 10:13:21 *** Press <ENTER> for shell prompt
12.11.2003 15:08:23 WARNING: invalid channel number: 3
socket: 0 length = 1632 net = 0 cmd = RBAC. Continue work
12.11.2003 15:08:23 WARNING: invalid channel number: 3
socket: 0 length = 1632 net = 0 cmd = KILL. Continue work
12.11.2003 15:08:27 *** Linter is coming down ***
12.11.2003 15:08:27 *** PP processing ***
12.11.2003 15:08:27 *** Table queue processing ***
12.11.2003 15:08:27 *** Column queue processing ***
12.11.2003 15:08:27 *** User queue processing ***
12.11.2003 15:08:27 *** File queue processing ***
12.11.2003 15:08:28 *** RDBMS Linter has been shut down ***

```

Указатель ключей

ANALYZE, 12
AUTOINDEX, 12
BASE, 6
CHECK, 8
DEBUG, 9
INMEMPOOL, 6
JDBCP, 7
JDBCS, 8
JEXIT, 7
K, 7
KILL, 10
LOCK, 11
LOG, 9
LOGALL, 9
LOGFLIMIT, 9
LOGQUERY, 9
MBX, 6
NAME, 6
NOJEXIT, 7
NOLARGE, 8
NOLOG, 9
NOOUTFILE, 7
NOOUTPUT, 8
NOSQL, 10
NOTSP, 10
OUTFLIMIT, 8
PASS, 12
PIDFILE, 7
POOL, 6
PPOOL, 6
RAPID, 13
RO, 11
SETPASS, 12
SPOOL, 6
SPX, 8
TCORRECT, 7
TCP, 8
TMPDIR, 7
TRACE, 10, 11
TRACEFLIMIT, 9
TRACELOG, 11
U, 13
VERSION, 13
WLHB, 6

