

**СИСТЕМА  
УПРАВЛЕНИЯ  
БАЗАМИ  
ДАнных**

**ЛИНТЕР®**

**ЛИНТЕР БАСТИОН  
ЛИНТЕР СТАНДАРТ**

**Установка СУБД ЛИНТЕР в среде  
ОС UNIX, QNX**

**НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ**

** РЕЛЭКС®**

## Товарные знаки

РЕЛЭКС™, ЛИНТЕР® являются товарными знаками, принадлежащими ЗАО 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 (1990-2019). 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: [www.relex.ru](http://www.relex.ru) и [www.linter.ru](http://www.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: [www.relex.ru](http://www.relex.ru) и [www.linter.ru](http://www.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: [market@relex.ru](mailto:market@relex.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

<b>Предисловие</b> .....	2
Назначение документа .....	2
Для кого предназначен документ .....	2
Необходимые предварительные знания .....	2
Дополнительные документы .....	2
<b>Общие сведения</b> .....	3
Необходимые ресурсы .....	3
Виды дистрибутивов .....	3
Порядок установки СУБД ЛИНТЕР .....	3
<b>Процедура установки</b> .....	5
Подготовительные операции .....	5
Выбор языка установки .....	5
Ознакомление с лицензионным соглашением .....	6
Выбор установочного каталога .....	7
Распаковка архива .....	7
Настройка программных интерфейсов .....	8
Си .....	9
ЛИНТЕР Perl .....	9
DBI Perl .....	9
PHP3 .....	10
PHP4 .....	11
Python .....	11
Ruby .....	12
TCL .....	13
TK .....	13
Java .....	14
unixODBC .....	14
Выбор текущей базы данных .....	15
Создание вспомогательных файлов .....	15
Командный файл для запуска сетевого драйвера сервера .....	16
Командный файл для запуска сетевого драйвера клиента .....	16
Командный файл для работы Java-приложений .....	16
Командный файл для удалённого управления компонентами .....	17
Командный файл для архивирования БД .....	17
Командный файл для асинхронной репликации данных .....	17
Кодировка словаря базы данных .....	18
Кодировка для таблиц по умолчанию .....	19
Файл сетевой конфигурации .....	20
Запуск ядра СУБД ЛИНТЕР .....	20
Интеграция командных файлов СУБД ЛИНТЕР и ОС .....	21
Создание символических ссылок к исполняемым файлам .....	21
Автоматический старт ядра и утилит СУБД ЛИНТЕР при запуске ОС .....	22
<b>Проверка установки</b> .....	23
<b>Процедура ввода серийного номера СУБД ЛИНТЕР</b> .....	24
<b>Процедура удаления СУБД ЛИНТЕР</b> .....	25
<b>Примеры работы с СУБД ЛИНТЕР</b> .....	26
<b>Краткий список утилит и динамических библиотек СУБД ЛИНТЕР</b> .....	28
<b>Приложение. Объекты БД DEMO</b> .....	30

---

# Предисловие

## Назначение документа

Документ содержит описание процедуры установки СУБД ЛИНТЕР в операционных системах типа UNIX или QNX. Приводится типовая процедура установки. Особенности установки в каждой конкретной операционной системе отражены в файле `install.txt` соответствующего дистрибутива.

Документ предназначен для СУБД ЛИНТЕР СТАНДАРТ 6.0 сборка 17.48, далее по тексту СУБД ЛИНТЕР.

## Для кого предназначен документ

Документ предназначен для пользователей, устанавливающих СУБД ЛИНТЕР, и для разработчиков программного обеспечения, использующих различные программные интерфейсы СУБД ЛИНТЕР.

## Необходимые предварительные знания

Для успешной установки СУБД необходимо иметь навыки работы в соответствующей ОС.

## Дополнительные документы

- СУБД ЛИНТЕР. Архитектура СУБД.
- СУБД ЛИНТЕР. Репликация данных.
- СУБД ЛИНТЕР. Perl-интерфейсы.
- СУБД ЛИНТЕР. PHP-интерфейсы.
- СУБД ЛИНТЕР. Python-интерфейс.
- СУБД ЛИНТЕР. TCL/TK-интерфейс.
- СУБД ЛИНТЕР. ODBC-драйвер.
- СУБД ЛИНТЕР. Архивирование и восстановление базы данных.
- СУБД ЛИНТЕР. Сетевые средства.
- СУБД ЛИНТЕР. Запуск и останов СУБД ЛИНТЕР в среде ОС UNIX, QNX.
- СУБД ЛИНТЕР. Интерфейс нижнего уровня.

---

# Общие сведения

## Необходимые ресурсы

Для установки СУБД ЛИНТЕР необходимо иметь:

- дистрибутив СУБД ЛИНТЕР;
- интерактивный доступ в ОС с возможностью установки программ (для полной конфигурации СУБД требуются права `root`);
- 200 Мбайт свободной памяти на жёстком диске сервера СУБД.

## Виды дистрибутивов

СУБД ЛИНТЕР может поставляться в следующем виде:

- 1) стандартный `rpm`-пакет;
- 2) стандартный `pkg`-пакет;
- 3) собственный пакет установки;
- 4) установочный `shell`-скрипт.

При поставке дистрибутива в виде `rpm(pkg)`-пакета используется стандартная для соответствующего пакета процедура установки (с помощью программ `rpm` или `pkgadd`). Данный документ описывает процедуру установки дистрибутива, распространяемого в собственном формате установки, но вся информация, относящаяся к конфигурированию СУБД или интеграции её с ОС справедлива для `rpm(pkg)`-пакета и установочного `shell`-скрипта.

Стандартный установочный пакет СУБД ЛИНТЕР состоит из 4 файлов:

- 1) `install.sh` – основной установочный скрипт;
- 2) `install.txt` – краткая информация, необходимая для успешной установки данного конкретного дистрибутива;
- 3) `license.txt` – стандартный текст лицензии для СУБД ЛИНТЕР;
- 4) `linter.tar.Z` (расширение может быть и другим – например, `.tar.gz` или `.tgz`) – собственно дистрибутив.

Установочный `shell`-скрипт представляет собой программу для распаковки дистрибутива СУБД ЛИНТЕР и упакованный набор файлов дистрибутива. Скрипт распаковывает стандартный пакет во временный каталог и запускает из него скрипт `install.sh`. Преимущество такой поставки заключается в том, что все необходимые для установки и настройки СУБД средства расположены в одном файле.

## Порядок установки СУБД ЛИНТЕР

Установка СУБД ЛИНТЕР включает в себя выполнение следующих фаз:

- 1) подготовительные операции;
- 2) выбор языка установки;

- 3) ознакомление с условиями лицензии;
- 4) выбор каталога для установки;
- 5) распаковка архива;
- 6) настройка программных интерфейсов;
- 7) создание БД;
- 8) создание командных файлов для запуска и останова ядра и вспомогательных утилит СУБД;
- 9) запуск ядра СУБД;
- 10) интеграция процедур запуска и останова СУБД в стандартные процедуры запуска и останова ОС, интеграция компонентов СУБД ЛИНТЕР с графической оболочкой ОС.

Все фазы, за исключением 1-4, являются необязательными. Любую из них можно выполнить впоследствии после прерывания процедуры установки. Опытный пользователь может обойтись без конфигурации, настроив все необходимые параметры и переменные самостоятельно (см. настоящий документ и документ «СУБД ЛИНТЕР. Архитектура СУБД»).

В случае если не требуется интеграция со стандартными процедурами запуска и останова ОС или с графической оболочкой, специальные привилегии не требуются – СУБД может быть установлена обычным пользователем ОС.

Если СУБД устанавливается из rpm(pkg)-пакета, то необходимые привилегии будут запрошены соответствующей программой установки. Кроме того, в этом случае будут проверяться зависимости для всех входящих в дистрибутив компонентов. Многие из этих зависимостей могут не потребоваться (используются в специфических интерфейсах), поэтому отсутствие некоторых компонентов можно проигнорировать. Программа установки ведёт диалог на английском языке вне зависимости от операционной системы и locale.

Для установки дистрибутива из установочного shell-скрипта достаточно запустить этот shell-скрипт. При отсутствии прав на выполнение скрипта его запуск может быть осуществлен с помощью sh.

**Внимание!** При передаче файла по ftp-протоколу копирование установочного shell-скрипта должно производиться в binary-режиме, поскольку он содержит не только текстовую информацию, но и упакованные данные.

---

# Процедура установки

## Подготовительные операции

Файлы дистрибутива (один или несколько) могут находиться на разных носителях данных:

- локальный жёсткий диск;
- оптический или магнитный диск;
- flash-память и др.

Для некоторых устройств необходимо предварительно смонтировать их (командой `mount`).

В случае копирования дистрибутива по сети может потребоваться установка признака исполнения для файла `install.sh` (командой `chmod +x install.sh`). Для копирования двоичных файлов по протоколу `ftp` необходимо использовать режим `bin`.

Для установки с помощью `rpm(pkg)`-пакета смотрите документацию на соответствующие утилиты установки (`rpm` или `pkgadd`).

Для установки с помощью скрипта `install.sh` необходимо, чтобы в каталоге запуска скрипта присутствовали все остальные файлы дистрибутива (или текущий каталог должен быть каталогом с дистрибутивом – например, на CD-ROM).

Для установки СУБД ЛИНТЕР с помощью скрипта `install.sh` или `linter-<версия СУБД>-6.0.<номер подверсии>.<номер сборки>-linux-<разрядность ОС>.sh`, где:

- `<версия СУБД>` – `bastion` или `standard`;
- `<разрядность ОС>` – 32 или 64;

(например, `linter-bastion-6.0.17.42-linux-x64.sh`) необходимо выполнить следующие шаги (обозначены римскими цифрами):

I. Установить в качестве текущего каталога каталог с дистрибутивом СУБД ЛИНТЕР (например, на CD-ROM) либо скопировать в каталог со скриптом `install.sh` файлы дистрибутива и установить его текущим каталогом.

II. Запустить на выполнение скрипт:

```
sh install.sh
```

или

```
sh linter-bastion-6.0.17.42-linux-x64.sh
```



### Примечание

Имя установочного shell-скрипта `linter-bastion-6.0.17.42-linux-x64.sh` может отличаться от приведенного, т. к. оно зависит от операционной системы и версии дистрибутива. В данном случае в качестве примера приведено имя скрипта для ОС Linux версии 6.0.17.42.

## Выбор языка установки

III. Сразу же после запуска скрипта будет выдан запрос на выбор языка диалога в процессе установки:

```
Untar distributive files. Please wait ...
Done
1 - english
2 - russian_utf8
3 - russian_koi8
4 - russian_cp866
Select language please (from 1 to 4):
```

Необходимо ввести число от 1 до 4, соответствующее желаемому языку диалога и его кодировке:

- 1 – английский;
- 2 – русский в кодировке utf-8;
- 3 – русский в кодировке koi8-г;
- 4 – русский в кодировке cp866.

Если в ответ на запрос ввести пустое значение, то по умолчанию будет использоваться английский язык.

## Ознакомление с лицензионным соглашением

- IV. После выбора языка установки будет выдан запрос с инструкцией по просмотру лицензии (для просмотра используется программа `more`) и предложено нажать клавишу **<Enter>** для начала ознакомления с лицензионным соглашением.

Пожалуйста, нажмите клавишу "Enter" для начала ознакомления с текстом лицензионного соглашения.

Для прокрутки текста используйте клавишу "ПРОБЕЛ" ("Space").

Установка СУБД ЛИНТЕР предполагает обязательное ознакомление с условиями лицензионного соглашения. В случае если в дистрибутиве по какой-либо причине отсутствует лицензионное соглашение, необходимо связаться с поставщиком или с разработчиком СУБД.

Типовой дистрибутив СУБД ЛИНТЕР содержит стандартное лицензионное соглашение. В случае поставки для конкретного заказчика лицензионное соглашение может быть изменено; в этом случае стандартное лицензионное соглашение можно игнорировать.

При стандартной поставке СУБД ЛИНТЕР текст лицензионного соглашения находится в файле `license.txt`, входящем в дистрибутивный пакет.

При установке СУБД ЛИНТЕР самостоятельно (без использования установочного скрипта `install.sh`) или с помощью `rpm(pkg)`-пакета необходимо ознакомиться с файлом `license.txt` и согласиться (или не согласиться) с предоставленной лицензией. В случае несогласия с её условиями следует отказаться от процедуры установки и связаться с поставщиком или разработчиком СУБД.

- V. После завершения просмотра лицензионного соглашения будет предложено подтвердить согласие с его условиями.



Согласны ли Вы с условиями приведенного выше лицензионного соглашения? (Y/N) :

Ответить **Y** в случае согласия с предложенной лицензией или другой имеющейся лицензией на данную копию СУБД.

Любой другой ответ прекращает установку СУБД ЛИНТЕР.

## Выбор установочного каталога

При установке с помощью `rpm(pkg)`-пакета установочный каталог выбирается автоматически.

При установке с помощью скрипта `install.sh` есть возможность выбора установочного каталога.

### VI. Выбор местоположения установочного каталога

СУБД ЛИНТЕР может быть установлена в любом каталоге по выбору администратора или в текущем каталоге (`.`).

Для установки СУБД Вы должны иметь привилегии записи в каталог установки.

Выбрать каталог для установки СУБД? (Y/N) [Y] :

Ответить **N**, если установка должна выполняться в текущий каталог (если установка выполняется с CD-ROM, то она завершится с ошибкой).

Ответить **Y** для выбора другого установочного каталога.

В обоих случаях установка СУБД ЛИНТЕР будет выполняться в подкаталог `linter` текущего или выбранного каталога.

### VII. Указание пути к установочному каталогу

Если был выбран вариант выбора установочного каталога, будет предложено указать путь к нему:

Пожалуйста, введите каталог установки СУБД ЛИНТЕР:

Ввести полный путь к каталогу, в котором будет создан подкаталог `linter` с установленной СУБД ЛИНТЕР.

Если указанный каталог не существует, он будет создан автоматически (при наличии прав на его создание).

## Распаковка архива

При установке с помощью `rpm(pkg)`-пакета соответствующая программа установки автоматически распакует дистрибутив.

При использовании установочного скрипта `install.sh` распаковка дистрибутива выполняется в выбранный каталог.

Распаковка дистрибутива СУБД ЛИНТЕР. Пожалуйста, подождите ...

Через несколько секунд или минут (в зависимости от ЭВМ, ОС и дистрибутива) и в случае отсутствия сообщений об ошибках или аварийном прекращении установки архив дистрибутива будет распакован в указанный каталог.



### Примечание

При самостоятельной установке (без использования rpm(pkg)-пакета или установочного скрипта `install.sh`) выполнить распаковку архива можно в любой установочный каталог, например, командой:

```
compress -dc linter.tar.Z | tar xf -
```

или другим способом – в зависимости от вида пакета и доступных архиваторов.

На этом обязательные фазы установки заканчиваются. Дальнейшие фазы являются необязательными и могут неоднократно выполняться в будущем при изменении конфигурации программных средств.

## Настройка программных интерфейсов

СУБД ЛИНТЕР включает программные интерфейсы к группе языков программирования. Функционирование этих программных интерфейсов зачастую зависит от установленных на данной ЭВМ версий языков программирования и их исполняющих систем.

Настройка программных интерфейсов обеспечивает:

- совместимость СУБД ЛИНТЕР с языками программирования и их версиями;
- возможность сборки загружаемых программных модулей, зависящих от версии программы, с которой они будут функционировать. Эти модули поставляются полностью или частично в исходных текстах и могут быть собраны самостоятельно при помощи встроенных `makefile`. Дистрибутив обычно включает уже готовые собранные модули, но может потребоваться модуль для другой (например, более новой) версии языка.

В процессе конфигурации программных интерфейсов сама сборка интерфейса не происходит – формируются только параметры этой сборки. Информация о сборке конкретного программного интерфейса содержится в документации на этот программный интерфейс (см. документы СУБД ЛИНТЕР по программным интерфейсам).

Программа установки СУБД ЛИНТЕР обеспечивает конфигурирование следующих программных компонент:

- языка программирования Си;
- Perl-интерфейсов (в т.ч. и DBI);
- PHP-интерфейсов (версий 3, 4, 5, 6 и 7);
- Python-интерфейса;
- TCL/TK-интерфейса;
- Java-интерфейса;
- unixODBC-интерфейса.

Параметры конфигурации сохраняются в файле `Definition` в корневом каталоге установленного дистрибутива.

Конфигурация программных интерфейсов может быть выполнена вручную с помощью скрипта `scripts/configure` в любое удобное время. В этом случае файл `Definition` будет создан заново.

При установке СУБД ЛИНТЕР самостоятельно или с помощью rpm(pkg)-пакета конфигурация программных интерфейсов выполняется вручную.

---

При использовании скрипта `install.sh` настройку программных интерфейсов предлагается выполнить в процессе установки СУБД ЛИНТЕР.

## Си

### VIII. Настройка Си-интерфейса

Если есть необходимость сборки демонстрационных приложений С или Java, то можно запустить специальную процедуру конфигурации для их последующей сборки.

Если необходимо перестроить интерфейсы для Tcl/Tk, Perl, PHP, то для этого также может быть запущена эта же процедура конфигурации.

Процедура конфигурации может быть пропущена и выполнена в любой момент после установки СУБД.

Запустить процедуру конфигурации? (Y/N) [Y]:

При ответе Y программа конфигурации выполняет поиск компилятора языка программирования Си и компоновщика, необходимых для сборки демонстрационных примеров, расположенных в каталоге `linter/samples`. В случае если в ОС будет обнаружено несколько Си-компиляторов, будет предложено выбрать один из них. Установленные параметры конфигурации будут использоваться также при сборке модулей программного интерфейса.

В процессе конфигурации Си-интерфейса определяются возможности Си-компилятора и наличие библиотек (например, `posix threads`).

## ЛИНТЕР Perl

### IX. Настройка ЛИНТЕР Perl-интерфейса

Программа конфигурации выполняет поиск интерпретатора языка программирования Perl в ОС. При удачном поиске на экран выводится запрос на конфигурацию СУБД ЛИНТЕР для построения программного интерфейса:

Конфигурировать дистрибутив для построения Perl интерфейса? (Y/N) [Y]:

В случае положительного ответа в файле `Definition` сохраняются параметры конфигурирования, необходимые для построения ЛИНТЕР Perl-интерфейса. В дальнейшем эти параметры будут использоваться при построении Perl-интерфейса согласно инструкции, приведенной в файле `perl/readme.txt` и документе «СУБД ЛИНТЕР. Perl-интерфейсы».

Если необходимость в построении ЛИНТЕР Perl-интерфейса отсутствует, ответить N.

## DBI Perl

### X. Настройка DBI Perl-интерфейса

Кроме собственного Perl-интерфейса СУБД ЛИНТЕР поддерживает и стандартный DBI-интерфейс для работы с БД реляционного типа. В составе СУБД ЛИНТЕР для DBI-интерфейса имеется специальный драйвер. Если была осуществлена настройка для построения Perl-интерфейса (см. пункт [«Настройка ЛИНТЕР Perl-интерфейса»](#)),

## Процедура установки

и Perl в ОС содержит поддержку DBI-интерфейса, то на экран выводится запрос на построение DBI-драйвера СУБД ЛИНТЕР:

Конфигурировать дистрибутив для построения DBI драйвера ЛИНТЕР?  
(Y/N) [Y]:

В случае положительного ответа сборка DBI-драйвера может быть осуществлена согласно инструкции, приведенной в файле `perl-dbi/readme.txt` и документе «СУБД ЛИНТЕР. Perl-интерфейсы».

Если необходимость в построении DBI Perl-интерфейса отсутствует, ответить **N**.

## RНРЗ

### XI. Настройка RНРЗ-интерфейса

Хотите ли Вы конфигурировать дистрибутив для построения интерфейса RНР v.3.x.x? (Y/N) [N]:



### Примечание

Эта версия RНР устарела, поэтому ответ по умолчанию – **N**.

Ответить **Y**, если есть необходимость в конфигурировании данного интерфейса.

Для настройки данного интерфейса программе установки необходима информация о местоположении каталога заголовочных файлов RНР-интерфейса. Эти файлы могут располагаться на любом диске файловой системы ОС, поэтому для их поиска может потребоваться длительное время. Сократить время поиска можно при помощи локализации области поиска. Для этого программа установки выводит на экран список возможных вариантов поиска. Необходимо выбрать один из вариантов (число из диапазона 1-4), по умолчанию поиск выполняется по 2 варианту:

Пожалуйста, выберите метод поиска каталога заголовочных файлов:

- 1 - Ввод полного пути к заголовочным файлам
- 2 - Поиск в стандартных системных каталогах
- 3 - Поиск во всех файловых системах
- 4 - Пропустить конфигурацию RНР v.3.x.x

Введите число в диапазоне от 1 до 4 [Поиск в стандартных системных каталогах]:

Дальнейшее поведение программы установки определяется выбранным вариантом:

Номер ответа	Реакция программы установки
1	Выдает запрос на ввод пути к каталогу заголовочных файлов RНР-интерфейса: Enter the RНР v.3.x.x include files directory path: Обычно (но не обязательно) заголовочные файлы устанавливаются в каталог <code>/usr/include/php</code> или <code>/usr/local/include/php</code> .
2	Осуществляет поиск необходимых файлов в стандартных каталогах заголовочных файлов ОС и в домашнем каталоге пользователя ОС

Номер ответа	Реакция программы установки
3	Осуществляет поиск заголовочных файлов во всех файловых системах, смонтированных в данный момент
4	Отказ от дальнейшего конфигурирования РНР-интерфейса версии 3.x.x.
Клавиша <Enter>	Аналогично варианту 2 (выбор по умолчанию)

Выбор любого варианта должен быть подтвержден.

В случае выбора вариантов № 2 и 3 будет предложен список из всех найденных каталогов, в которых содержатся заголовочные файлы РНР-интерфейса, а также вариант, позволяющий отказаться от поиска.

Выбранный каталог проверяется на наличие необходимых файлов. В случае успешной проверки параметры конфигурации РНР-интерфейса записываются в файл `Definition`. В дальнейшем построение РНР-интерфейса может быть осуществлено согласно инструкции, приведенной в файле `linter/php/readme.txt` или в документе «СУБД ЛИНТЕР. РНР-интерфейсы».

## РНР4

### ХII. Настройка РНР4-интерфейса



#### Примечание

Необходимость выделения этой процедуры в качестве отдельного шага вызвана тем, что, начиная с версии 4, изменилась схема расположения и названия заголовочных файлов.

Хотите ли Вы конфигурировать дистрибутив для построения интерфейса РНР v.4.x.x? (Y/N) [Y] :

Ответить **Y**, если есть необходимость в конфигурировании данного интерфейса.

Дальнейшие действия по конфигурированию интерфейса аналогичны действиям по конфигурированию РНР-интерфейса версии 3.x.x (см. пункт [«Настройка РНР3-интерфейса»](#)).



#### Примечание

Для РНР-интерфейсов очень важно точное соответствие версии языка и версии, с которой собран модуль. В случае их несовпадения РНР-модуль просто не будет загружен.

## Python

### ХIII. Настройка Python-интерфейса

Конфигурировать дистрибутив для построения интерфейса Python? (Y/N) [Y] :

Ответить **Y**, если есть необходимость в конфигурировании данного интерфейса.

Настройка СУБД ЛИНТЕР для Python-интерфейса полностью совпадает с настройкой для РНР-интерфейса (см. пункт [«Настройка РНР3-интерфейса»](#) и

пункт «[Настройка RНР4-интерфейса](#)»), как в особенностях загрузки модулей, так и в произвольном расположении заголовочных файлов в файловой системе.

Выбранный каталог проверяется на наличие необходимых файлов. В случае успешной проверки результаты настройки Python-интерфейса записываются в файл `Definition` и в дальнейшем могут использоваться при построении Python-интерфейса командой `make` из каталога `linter/python` согласно инструкции `linter/python/readme.txt` или документу «СУБД ЛИНТЕР. Python-интерфейс».

## Ruby

### XIV. Настройка Ruby-интерфейса

Конфигурировать дистрибутив для построения интерфейса Ruby? (Y/N)  
[Y] :

Ответить Y, если есть необходимость в конфигурировании данного интерфейса.

Для настройки интерфейса программе установке необходимо указать местоположение заголовочных файлов Ruby. В процессе настройки интерфейса программа установки выдаст следующие варианты поиска заголовочных файлов Ruby:

- 1 - Быстрый поиск в стандартных системных каталогах
- 2 - Ввод полного пути к заголовочным файлам
- 3 - Поиск в стандартных системных каталогах
- 4 - Поиск во всех файловых системах
- 5 - Пропустить построение Ruby

Введите число в диапазоне от 1 до 5 [Быстрый поиск в стандартных системных каталогах]

В зависимости от выбранного варианта будет определяться дальнейшее поведение программы установки:

<u>Номер ответа</u>	<u>Реакция программы установки</u>
1	Осуществляет быстрый поиск необходимых файлов Ruby в стандартных каталогах заголовочных файлов ОС. Сканирование каталогов осуществляется с поиском по маске, которая задает потенциальное имя каталога местоположения заголовочных файлов ( <code>/ruby/*/*</code> , <code>/include/ruby*</code> , <code>/include/ruby*/*</code> )
2	Выдает запрос на ввод пути к каталогу заголовочных файлов Ruby. Данные файлы по умолчанию располагаются в каталоге <code>/usr/include/ruby</code> или <code>/usr/local/include/ruby</code> , но также могут присутствовать в любом месте файловой системы, например, в случае ручной установки Ruby в систему
3	Осуществляет поиск необходимых файлов Ruby в стандартных каталогах заголовочных файлов ОС и в домашнем каталоге пользователя ОС
4	Осуществляет поиск заголовочных файлов во всех файловых системах, смонтированных в данный момент

Номер ответа	Реакция программы установки
5	Отказ от дальнейшей настройки Ruby-интерфейса
Клавиша <Enter>	Аналогично варианту 1 (выбор по умолчанию)

Любой выбор из предложенных вариантов должен быть подтвержден. В результате поиска файлов будет выведен список каталогов, в которых были обнаружены заголовочные файлы Ruby, а также вариант для отмены процесса настройки интерфейса. После успешного завершения настройки интерфейса все необходимые параметры конфигурации будут занесены в файл `Definition`. Далее из каталога `linter/ruby` необходимо произвести сборку библиотеки, подав команду:

```
make clean all
```

В результате успешной сборки в каталоге `linter/bin` будет располагаться собранная библиотека `LinRuby.so`.

## TCL

### XV. Настройка TCL-интерфейса

Конфигурировать дистрибутив для построения TCL интерфейса? (Y/N)

[Y]:

Ответить **Y**, если есть необходимость в конфигурировании данного интерфейса.

Для настройки данного интерфейса программа установки выполняет поиск конфигурационного файла `tclConfig.sh` в стандартных каталогах ОС.

В случае успешного поиска на экран выводится запрос на подтверждение правильности пути к найденному конфигурационному файлу:

Конфигурационный скрипт TCL найден в (`/path/to/tclConfig.sh`).

Путь к конфигурационному скрипту TCL верен (Y/N) [Y]:

В случае утвердительного ответа настройка считается выполненной. Если файл конфигурации не найден или ответ на запрос о корректности этого файла отрицательный, то выводится запрос на указание метода поиска конфигурационного файла (текст запроса аналогичен запросу при настройке РНР-интерфейса, см. пункт [«Настройка РНР3-интерфейса»](#)).

Если конфигурационный файл найден, то необходимая для конфигурации TCL-интерфейса информация переписывается из конфигурационного файла в файл `Definition`. В дальнейшем построение TCL-интерфейса может быть осуществлено согласно инструкции `tcl/readme.txt` или документу «СУБД ЛИНТЕР. TCL/ТК-интерфейс».

Драйвер может быть построен в виде TCL-пакета или скомпилирован совместно с оболочкой `tcl linsh`.

## ТК

### XVI. Настройка ТК-интерфейса

Конфигурировать дистрибутив для построения ТК с интерфейсом

ЛИНТЕР? (Y/N) [Y]:

Ответить **Y**, если есть необходимость сборки оболочки ТК со встроенным интерфейсом СУБД ЛИНТЕР.

Дальнейшее конфигурирование СУБД ЛИНТЕР выполняется аналогично конфигурированию ТСЛ-интерфейса (см. пункт [«Настройка ТСЛ-интерфейса»](#)), за исключением того, что конфигурационный файл имеет имя `tkConfig.sh`, а оболочка – `linwi.sh`.

## Java

### XVII. Настройка Java-интерфейса

В начале настройки программа конфигурации выводит на экран запрос на необходимость конфигурирования СУБД ЛИНТЕР для сборки примеров, написанных на языке программирования Java:

Конфигурировать дистрибутив для построения демонстрационных приложений Java? (Y/N) [Y]:

Ответить **Y**, если есть необходимость в конфигурировании СУБД ЛИНТЕР для работы с Java-программами и/или для сборки примеров, написанных на языке программирования Java. В этом случае программа установки выполняет поиск установленных в ОС Java-компиляторов. В случае успешного поиска информация о них записывается в файл `Definition`. СУБД ЛИНТЕР поддерживает различные стандарты JDBC – 1, 2, 3, 4, 4.1 и 4.2. Примеры написаны с использованием JDBC 1. Однако для их сборки и функционирования может использоваться любая версия JDK.

## unixODBC

### XVIII. Настройка unixODBC-интерфейса

В составе дистрибутива СУБД ЛИНТЕР поставляется готовый к использованию ODBC драйвер в виде статической библиотеки и разделяемой библиотеки. Если для доступа к БД используется ODBC драйвер-менеджер типа unixODBC, в котором присутствуют графические средства конфигурирования DSN, то программа установки позволяет собрать специальный драйвер конфигурации DSN ЛИНТЕР. Для этого на экран выводится запрос о необходимости установки библиотеки, используемой для построения драйвера-конфигуратора:

Будете ли вы строить модуль конфигурирования DSN unixODBC (Y/N) [N]:

Ответить **Y**, если есть необходимость в данном конфигурировании.

Для сборки драйвера-конфигуратора необходимы исходные файлы драйвера-менеджера unixODBC, поэтому программа установки предложит выбрать метод поиска каталога этих исходных файлов: автоматический поиск или ручной ввод пути к исходным файлам. После проверки правильности введенного пути необходимая для сборки информация сохраняется в файле `Definition`. В дальнейшем построение библиотеки конфигурации может быть осуществлено согласно инструкции `linter/odbc/readme.txt` или документу «СУБД ЛИНТЕР. ODBC-драйвер».

На этом фаза конфигурации программных интерфейсов завершается.



## Выбор текущей базы данных

При установке СУБД ЛИНТЕР любым способом автоматически создаётся демонстрационная БД (в подкаталоге /db установочного каталога). Состав объектов демонстрационной БД (ДЕМО) приведен в [приложении](#). Если установка выполняется с помощью конфигурационного файла `install.sh`, то можно дополнительно создать БД, аналогичную демонстрационной БД, но в любом другом каталоге.

XIX. Для этого на экран выводится запрос:

Будет использована база данных из дистрибутива или создать новую:

- 1 - Использовать демонстрационную базу данных из дистрибутива
- 2 - Использовать существующую базу данных
- 3 - Создать новую базу данных аналогичную демонстрационной

Введите число в диапазоне от 1 до 3 [Использовать демонстрационную базу данных из дистрибутива]:

Дальнейшее поведение программы установки определяется выбранным вариантом.

В первом случае будет использоваться БД, созданная разработчиками СУБД и встроенная в дистрибутив (демонстрационная БД).

Вы уверены в выборе "Использовать демонстрационную базу данных из дистрибутива"? (Y/N) [N]:

Во втором случае будет использоваться существующая БД, местоположение которой будет запрошено программой установки.

В третьем случае дополнительно будет создана ещё одна демонстрационная БД (в каталоге, запрошенном программой установки). Если заданный каталог не существует, программа установки выведет запрос на подтверждение создания каталога. С практической точки зрения стандартная демонстрационная БД и созданная БД одинаковы, за исключением своего расположения.

Выбранная БД становится *текущей* БД и используется во всех дальнейших шагах процесса установки (создание командных файлов, запуск ядра СУБД ЛИНТЕР и др.).

Если в состав дистрибутива СУБД ЛИНТЕР включена система репликации данных, то в подкаталоге /storage установочного каталога создаётся БД системы репликации. Аналогично ситуации с демонстрационной БД при выборе второго варианта ответа можно создать ещё одну БД системы репликации в другом каталоге.

Используя скрипт `scripts/make_database`, можно создать собственную демонстрационную БД.

## Создание вспомогательных файлов

Создание вспомогательных командных файлов, автоматизирующих некоторые стандартные операции при работе с СУБД ЛИНТЕР, выполняется скриптом `scripts/make_scripts`. При установке СУБД ЛИНТЕР с помощью файла `install.sh` этот скрипт вызывается автоматически программой установки, а при самостоятельной

установке или установке с использованием rpm(pkg)-пакета его надо запускать вручную.

В результате работы этого скрипта в подкаталоге bin установочного каталога будут созданы командные файлы start\_linter.sh и stop\_linter.sh, которые являются процедурами автоматического управления компонентами СУБД ЛИНТЕР. Эти скрипты впоследствии, при необходимости, могут быть модифицированы администратором СУБД для подстройки под собственные требования.

Для подтверждения создания командных файлов программа установки выводит запрос: Создать скрипты запуска и останова для работы СУБД с данной БД? (Y/N) [Y]:

В случае утвердительного ответа программа установки будет запрашивать дополнительные сведения о запуске необходимых компонентов СУБД ЛИНТЕР.

## Командный файл для запуска сетевого драйвера сервера

XX. Командный файл для запуска сетевого драйвера сервера

Создание командного файла для управления запуском сетевого драйвера сервера: Сетевой драйвер сервера (dbs\_tcp) позволяет обслуживать запросы с удаленных клиентских машин через TCP/IP сеть. Стартовать сетевой драйвер сервера из скрипта запуска? (Y/N) [Y]:

Ответить Y, если совместно с запуском ядра СУБД ЛИНТЕР надо автоматически запускать сетевой драйвер сервера (dbs\_tcp). Сетевой драйвер сервера позволяет принимать запросы к ЛИНТЕР-серверу не только с локальной машины, но и с любой машины в сети TCP/IP.

## Командный файл для запуска сетевого драйвера клиента

XXI. Командный файл для запуска сетевого драйвера клиента

Создание командного файла для управления запуском сетевого драйвера клиента: Для работы с удаленными серверами ЛИНТЕР на клиентской машине должен быть запущен сетевой драйвер клиента (dbc\_tcp). Стартовать сетевой драйвер клиента из скрипта запуска? (Y/N) [N]:

Ответить Y, если на компьютере, в котором выполняется текущая установка СУБД ЛИНТЕР, надо автоматически запускать сетевой драйвер клиента (dbc\_tcp). Сетевой драйвер клиента позволяет обмениваться информацией данному компьютеру с удаленным ЛИНТЕР-сервером.

## Командный файл для работы Java-приложений

XXII. Командный файл для работы Java-приложений

Создание командного файла для управления работой Java-приложений: JDBC сервер позволяет JAVA приложениям работать с СУБД ЛИНТЕР через JDBC интерфейс.

---

Стартовать JDBC сервер из скрипта запуска? (Y/N) [Y]:

Ответить Y, если необходимо запускать JDBC-драйвер для работы с СУБД ЛИНТЕР локальных и/или удаленных Java-приложений.

## Командный файл для удалённого управления компонентами

### XXIII. Командный файл для удалённого управления компонентами

Создание командного файла для удалённого управления компонентами:

SNMP сервер позволяет удаленно управлять работой ядра и утилит СУБД ЛИНТЕР

Стартовать SNMP сервер из скрипта запуска? (Y/N) [N]:

Ответить Y, если необходимо автоматически запускать SNMP-сервер, с помощью которого можно удаленно управлять запуском/остановом ядра СУБД ЛИНТЕР и его компонентов.

## Командный файл для архивирования БД

### XXIV. Командный файл для архивирования БД

Создание командного файла для управления архивированием БД:

Для повышения надежности рекомендуется выполнять архивирование базы данных.

Данная операция может быть выполнена без прерывания работы ядра СУБД с помощью утилиты lhb.

Рекомендуется выполнять эту операцию ежедневно или чаще.

Запускать lhb из скрипта запуска? (Y/N) [Y]:

Ответить Y, если необходимо создать командный файл для запуска утилиты архивирования lhb в фоновом режиме для ежедневного инкрементного архивирования БД. В ходе дальнейшего диалога необходимо указать каталог, в котором будут храниться файлы архива, время архивирования (по GMT), количество файлов архива. Ежедневно файл архива БД будет пополняться накопленными в БД за день изменениями. Новый файл архива создается один раз в неделю в воскресенье. Таким образом, количество файлов архива соответствует количеству недель хранения архива (см. документ «СУБД ЛИНТЕР. Архивирование и восстановление базы данных»).

## Командный файл для асинхронной репликации данных

### XXV. Командный файл для асинхронной репликации данных

Создание командного файла для управления асинхронной репликацией данных:

Данный дистрибутив ЛИНТЕР поддерживает асинхронную пересылку (репликацию) данных на удаленные сервера ЛИНТЕР.

Активизировать поддержку асинхронной репликации? (Y/N) [Y]:



### Примечание

Запрос выдается в случае, если в состав дистрибутива включена подсистема репликации данных.

Ответить **Y**, если необходимо автоматически запускать сервер асинхронной репликации и ядро СУБД ЛИНТЕР для вспомогательной БД. Затем в ходе интерактивного диалога выполнить настройку списка удаленных серверов – указать имя и IP-адрес каждого удаленного сервера. Программа установки запрашивает имена удаленных серверов до тех пор, пока не будет введено пустое имя. После добавления каждого сервера выводится полный список уже добавленных серверов и их IP-адресов. По окончании ввода списка серверов программа установки сконфигурирует демонстрационную БД для работы в режиме асинхронной репликации.

Впоследствии между этими серверами может быть настроена асинхронная репликация данных. Если система репликации должна быть активизирована из стартового скрипта, то из него же автоматически будет запускаться и сетевой драйвер клиента `dbc_tcp`. Для этого программа установки создаст конфигурационный файл сетевого драйвера клиента `dbc_tcp (nodetab)`, включив туда все перечисленные удаленные сервера (см. документы «СУБД ЛИНТЕР. Сетевые средства» и «СУБД ЛИНТЕР. Репликация данных»).

## Кодировка словаря базы данных

### XXVI. Кодировка словаря базы данных

Выбор кодировки словаря базы данных:

Пожалуйста, выберите кодировку словаря базы данных:

- 1 - CP1251:Cyrillic (CP1251, windows-1251)
- 2 - CP1252:Western (CP1252, windows-1252)
- 3 - CP437:Western (CP437, cp437)
- 4 - CP850:Western (CP850, cp850, ibm850)
- 5 - CP866:Cyrillic (CP866, ibm866)
- 6 - CP8859-10:Nordic (CP8859-10, ISO-8859-10, latin6)
- 7 - CP8859-13:Baltic (CP8859-13, ISO-8859-13)
- 8 - CP8859-14:Celtic (CP8859-14, ISO-8859-14, Latin8, iso-celtic)
- 9 - CP8859-15:Western (CP8859-15, ISO-8859-15, latin-9)
- 10 - CP8859-1:Western (CP8859-1, ISO-8859-1, latin1)
- 11 - CP8859-2:Central European (CP8859-2, ISO-8859-2, latin2)
- 12 - CP8859-3:South European (CP8859-3, ISO-8859-3, latin3)
- 13 - CP8859-4:Baltic (CP8859-4, ISO-8859-4, latin4)
- 14 - CP8859-5:Cyrillic (CP8859-5, ISO-8859-5, cyrillic)
- 15 - CP8859-6:Arabic (CP8859-6, ISO-8859-6, arabic)
- 16 - CP8859-7:Greek (CP8859-7, ISO-8859-7, greek, greek8)
- 17 - CP8859-8:Hebrew Visual (CP8859-8, ISO-8859-8, hebrew)
- 18 - CP8859-9:Turkish (CP8859-9, ISO-8859-9, latin5)
- 19 - DEFAULT:Default (ASCII-7, US-ASCII, IBM367, cp367, ISO646-US)
- 20 - KOI8-R:Cyrillic (KOI8-R)
- 21 - Cancel

Введите число в диапазоне от 1 до 21  
 [CP1252:Western(CP1252, windows-1252)] :

После ввода кода кодировки будет задан вопрос:

Вы уверены в выборе "KOI8-R:Cyrillic(KOI8-R)"? (Y/N) [N] :

Ответить **Y**, если выбрана требуемая кодировка, иначе ответить **N** и выбрать другую требуемую кодировку словаря базы данных.

## Кодировка для таблиц по умолчанию

### XXVII. Кодировка для таблиц по умолчанию

Выбор кодировки для таблиц по умолчанию:

Пожалуйста, выберите кодировку для создаваемых таблиц по умолчанию:

- 1 - CP1251:Cyrillic(CP1251, windows-1251)
  - 2 - CP1252:Western(CP1252, windows-1252)
  - 3 - CP437:Western(CP437, cp437)
  - 4 - CP850:Western(CP850, cp850, ibm850)
  - 5 - CP866:Cyrillic(CP866, ibm866)
  - 6 - CP8859-10:Nordic(CP8859-10, ISO-8859-10, latin6)
  - 7 - CP8859-13:Baltic(CP8859-13, ISO-8859-13)
  - 8 - CP8859-14:Celtic(CP8859-14, ISO-8859-14, Latin8, iso-celtic)
  - 9 - CP8859-15:Western(CP8859-15, ISO-8859-15, latin-9)
  - 10 - CP8859-1:Western(CP8859-1, ISO-8859-1, latin1)
  - 11 - CP8859-2:Central European(CP8859-2, ISO-8859-2, latin2)
  - 12 - CP8859-3:South European(CP8859-3, ISO-8859-3, latin3)
  - 13 - CP8859-4:Baltic(CP8859-4, ISO-8859-4, latin4)
  - 14 - CP8859-5:Cyrillic(CP8859-5, ISO-8859-5, cyrillic)
  - 15 - CP8859-6:Arabic(CP8859-6, ISO-8859-6, arabic)
  - 16 - CP8859-7:Greek(CP8859-7, ISO-8859-7, greek, greek8)
  - 17 - CP8859-8:Hebrew Visual(CP8859-8, ISO-8859-8, hebrew)
  - 18 - CP8859-9:Turkish(CP8859-9, ISO-8859-9, latin5)
  - 19 - CP932:Japanese(CP932, Shift\_JIS, x-sjis, x-shift-jis, x-ms-cp932)
  - 20 - CP936:Chinese Simplified(CP936, GB2312, csGB2312)
- Press "Enter" to continue

- 21 - CP949:Korean(CP949, ks\_c\_5601-1987, windows-949)
- 22 - CP950:Chinese Traditional(CP950, big5, windows-950, csBig5, CN-Big5)
- 23 - DEFAULT:Default(ASCII-7, US-ASCII, IBM367, cp367, ISO646-US)
- 24 - EUC\_JP:Japanese(EUC-JP, X-EUC-JP, x-euc)
- 25 - KOI8-R:Cyrillic(KOI8-R)
- 26 - Cancel

Введите число в диапазоне от 1 до 26  
 [CP1252:Western(CP1252, windows-1252)] :

После ввода кода кодировки будет задан вопрос:

Вы уверены в выборе "KOI8-R:Cyrillic (KOI8-R)"? (Y/N) [N]:

Ответить **Y**, если выбрана требуемая кодировка, иначе ответить **N** и выбрать другую требуемую кодировку для таблиц по умолчанию.

## Файл сетевой конфигурации

### XXVIII. Файл сетевой конфигурации

Создание файла сетевой конфигурации:

Для доступа к удаленным базам данных используется сетевой драйвер клиента `dbc_tcp`.

Он имеет файл настройки `nodetab`, в котором приведен список удаленных СУБД ЛИНТЕР.

Редактировать список удаленных СУБД ЛИНТЕР? (Y/N) [N]:

Ответить **Y**, если необходимо создать файл сетевой конфигурации (`nodetab`) на компьютере, в котором устанавливается СУБД ЛИНТЕР. В случае положительного ответа программа установки для каждого удаленного узла запрашивает следующую информацию:

- имя ЛИНТЕР-сервера;
- протокол;
- сетевой адрес ЛИНТЕР-сервера;
- параметры соединения (номер порта, тайм-ауты сервера, клиента, соединения).

Диалог прекращается при вводе пустого имени ЛИНТЕР-сервера.



### Примечание

После установки администратору ОС следует проверить привилегии, с которыми ОС будет запускать СУБД ЛИНТЕР. Настройка должна проводиться в соответствии с политикой информационной безопасности, применяемой в АС. Настройка осуществляется в соответствии с документацией на конкретную ОС.

При установке скрипт запуска СУБД ЛИНТЕР будет зарегистрирован в ОС, например (зависит от ОС):

```
/etc/init.d/linter.sh
```

Скрипт `linter.sh` запускает СУБД, путем выполнения скрипта `start_linter.sh` из подкаталога `bin` каталога установки.

Скрипт `linter.sh` осуществляет запуск командой "start-stop-daemon"

Для повышения или понижения привилегий необходимо скорректировать опцию "--chuid" для данной команды (смотри документацию для команды "start-stop-daemon" используемой ОС).

## Запуск ядра СУБД ЛИНТЕР

Программа установки `install.sh` может запустить ядро СУБД ЛИНТЕР для выбранной БД.

XXIX. Запуск ядра СУБД ЛИНТЕР:

Запускать ЛИНТЕР автоматически при старте системы? (Y/N) [Y]:

Ответить **Y**, если необходимо сразу (в процессе установки) запустить ядро СУБД ЛИНТЕР.

По умолчанию предлагается ядро не запускать. Запуск ядра можно выполнить позднее с использованием скрипта `bin/start_linter.sh` или вручную.

В случае ручного запуска необходимо запустить программу `linter` из каталога `bin` (можно прописать этот каталог в переменную окружения `PATH`, но это не обязательно). Местоположение БД задается либо с помощью переменной окружения `SY00`, либо в командной строке запуска программы `linter` (см. документ «СУБД ЛИНТЕР. Запуск и останов СУБД ЛИНТЕР в среде ОС UNIX, QNX»).

### **Примечание**

При использовании РАМ-аутентификации на некоторых ОС возможна деинициализация механизмов обмена ядра СУБД ЛИНТЕР при первом же использовании РАМ-аутентификации. Это проявляется в невозможности присоединения РАМ-аутентификации к ядру СУБД ЛИНТЕР (возвращается код завершения 1001) при работающем процессе `linter`. Такое поведение связано с ошибкой в РАМ. Эта ошибка исправлена в версии РАМ 1.1. Для устранения ее проявлений в других версиях необходимо с помощью настроек РАМ-аутентификации исключить создание отдельного процесса для аутентификации (убрать опцию `fork`).

## Интеграция командных файлов СУБД ЛИНТЕР и ОС

### **Примечание**

Нижеследующие операции рекомендуется выполнять при неактивном ядре СУБД ЛИНТЕР.

Желательно полный путь до каталога `linter/bin` добавить к переменной окружения `PATH` или, при наличии прав пользователя `root`, с помощью программы установки создать символические ссылки на исполняемые файлы СУБД ЛИНТЕР в каталоге `/usr/bin`.

## Создание символических ссылок к исполняемым файлам

XXX. Создание символических ссылок к исполняемым файлам

Создать символические ссылки на исполняемые файлы ЛИНТЕР в каталоге `/usr/bin`? (Y/N) [Y]:

Рекомендуется ответить **Y**.

Если установка производится от имени другого пользователя, то для выполнения данной операции будет запрошен пароль пользователя `root`.

После этого будет произведена попытка настроить меню `kde` и `gnome` для запуска утилит СУБД ЛИНТЕР. В случае обнаружения соответствующих WM их

настройки будут обновлены для запуска графических утилит СУБД ЛИНТЕР и процедур запуска и останова ядра СУБД.

## Автоматический старт ядра и утилит СУБД ЛИНТЕР при запуске ОС

### XXXI. Автоматический старт ядра и утилит СУБД ЛИНТЕР при запуске ОС

Для ОС Linux с набором командных файлов в стиле SYSTEM V может быть осуществлена настройка автоматического старта ядра и утилит СУБД ЛИНТЕР при запуске ОС.

Запускать ЛИНТЕР автоматически при старте системы? (Y/N) [Y]:

Ответить Y, если необходим запуск ядра СУБД ЛИНТЕР одновременно с запуском ОС.

Для выполнения данной настройки требуется привилегия пользователя root. Пароль пользователя root будет запрошен при необходимости.

На этом установка СУБД ЛИНТЕР завершается. При работе необходимо учитывать, что по умолчанию для созданных в процессе установки БД администратор БД имеет следующие регистрационные данные: имя SYSTEM, пароль MANAGER. В дальнейшем пароль администратора БД можно сменить. При этом необходимо изменить пароль администратора и в файле `linter/bin/private_passwd` (этот файл используется скриптом на запуск СУБД ЛИНТЕР). Вы можете отказаться от его использования, а можете придумать собственную процедуру запуска.



---

## Проверка установки

Проверка установки СУБД ЛИНТЕР зависит от ряда факторов:

- способа установки СУБД;
- наличия вспомогательных командных файлов;
- состояния ядра СУБД (активно или нет) и др.

Простейшая проверка работоспособности локальной СУБД ЛИНТЕР, не зависящая от перечисленных выше факторов, выполняется следующим образом:

1) перейти в каталог `linter/bin`;

2) указать местоположение БД одним из способов:

- установить в переменной окружения `SY00` путь к демонстрационной БД (находится в подкаталоге `\db` установочного каталога СУБД ЛИНТЕР):

```
export SY00=/linter/db
```

и запустить ядро СУБД ЛИНТЕР командой:

```
./linter
```

- указать путь к демонстрационной БД в значении ключа запуска ядра СУБД `/base`:

```
./linter /base=..\db
```

3) запустить программу командного интерфейса СУБД ЛИНТЕР:

```
./inl -u SYSTEM/MANAGER
```

4) на приглашение программы ввести SQL-запрос:

```
SQL>select count (*) from auto;
```

5) если установка СУБД ЛИНТЕР выполнена удачно и СУБД работоспособна, то на экран будет выдано сообщение:

```
|      1000|
```

```
INL : выдано строк      :1
```

6) выйти из программы `inl` командой `exit`;

7) завершить работу СУБД ЛИНТЕР с помощью команды:

```
./shut SYSTEM/MANAGER
```

Работа СУБД ЛИНТЕР будет завершена.

---

# Процедура ввода серийного номера СУБД ЛИНТЕР

Серийный номер содержит четыре части: Serial Number – серийный номер (8 цифр), License Type – номер типа лицензии (3 цифры), Additional Info – дата (8 цифр), License Key – контрольная комбинация из 12 символов.

Для ввода серийного номера СУБД ЛИНТЕР служит специальная утилита: `setlic`.

Способы использования утилиты `setlic`:

1) С помощью файла с информацией о серийном номере СУБД ЛИНТЕР (`license.key`):

- перейти в каталог `bin` дистрибутива СУБД ЛИНТЕР, остановить все запущенные ядра СУБД ЛИНТЕР;
- скопировать файл `license.key` в каталог `bin` дистрибутива СУБД ЛИНТЕР и подать команду:

```
setlic file license.key
```

2) При отсутствии файла `license.key` серийный номер можно ввести в диалоговом режиме:

- перейти в каталог `bin` дистрибутива СУБД ЛИНТЕР, остановить все запущенные ядра СУБД ЛИНТЕР;
- подать команду `setlic` и ввести необходимые сведения – составные части серийного номера СУБД ЛИНТЕР из поставляемого вместе с дистрибутивом СУБД ЛИНТЕР документа «Паспорт программного продукта», пункт 1.11.

При успешном вводе серийного номера СУБД ЛИНТЕР выдается сообщение `Checking license in progress... OK`.

При неправильном вводе серийного номера СУБД ЛИНТЕР выдается сообщение `Invalid license information!`.

---

## Процедура удаления СУБД ЛИНТЕР

Для удаления СУБД необходимо из подкаталога /bin установочного каталога СУБД выполнить команду:

```
sh uninstall.sh
```

В появившемся меню необходимо выбрать одну из кодировок интерфейса:

```
1 - english
```

```
2 - russian_utf8
```

```
3 - russian_koi8
```

```
Select language please ( from 1 to 3 ):
```

После ввода значения кодировки будет выведен вопрос:

Внимание! Все каталоги дистрибутива ЛИНТЕР будут удалены.

Сохраните, пожалуйста, Ваши данные.

```
Деинсталлировать СУБД ЛИНТЕР?(Y/N) [N]:
```

Ответить Y для деинсталляции СУБД.

После подтверждения выполнения деинсталляции СУБД будут выведены сообщения о ходе ее выполнения:

```
Модификация стартовых меню.
```

```
Пожалуйста, подождите ...
```

```
Ядро ЛИНТЕР не запущено
```

```
Ядро ЛИНТЕР не запущено
```

```
Деинсталляция... Выполнено
```

---

# Примеры работы с СУБД ЛИНТЕР

После установки СУБД ЛИНТЕР примеры работы с ней будут находиться в подкаталоге `\samples\` установочного каталога.

Краткое описание структуры и содержания подкаталогов каталога `\samples\`:

- 1) C – примеры программ на языке программирования Си, использующих интерфейс нижнего уровня (call-интерфейс) СУБД ЛИНТЕР. В файле `\samples\C\readme.txt` содержится инструкция по сборке исполняемых модулей;
- 2) CALL – примеры работы с интерфейсом нижнего уровня (call-интерфейсом) СУБД ЛИНТЕР программ на языке программирования Си. Все примеры, содержащиеся в этом каталоге, подробно описаны в документе «СУБД ЛИНТЕР. Интерфейс нижнего уровня»;
- 3) DB – примеры, демонстрирующие загрузку данных в БД ЛИНТЕР различными способами:
  - подкаталог DBF – пример конвертации из DBF-формата (с использованием утилиты `dbf2lin`);
  - подкаталог LOD – пример конвертации из текстового файла (с использованием утилиты `loarel`);
- 4) DBI – примеры работы с СУБД ЛИНТЕР из программ, написанных на языке программирования Perl DBI;
- 5) JDBC – примеры работы с СУБД ЛИНТЕР из программ, написанных на языке Java. В файле `\samples\JDBC\readme.txt` содержится подробное описание каждого класса-примера;
- 6) LAPI – примеры работы с прикладным интерфейсом СУБД ЛИНТЕР (высокоуровневый API к СУБД ЛИНТЕР). В файле `\samples\LAPI\readme.txt` содержится описание каждого примера;
- 7) LINAPI – примеры приложений, написанных на языке программирования Си с использованием LinAPI-интерфейса (высокоуровневый API к СУБД ЛИНТЕР). В файле `\samples\LINAPI\readme.txt` содержится описание каждого примера;
- 8) LINCPPAPI – примеры приложений, написанных на языке программирования Си++ с использованием LinCPPAPI-интерфейса (высокоуровневый API к СУБД ЛИНТЕР). В файле `\samples\LINCPPAPI\readme.txt` содержится описание каждого примера;
- 9) LINQT – примеры приложений, написанных с использованием QtLinter-драйвера, выполняющего доступ к СУБД ЛИНТЕР из приложений, разработанных с использованием среды разработки Qt;
- 10) ODBC – примеры приложений, написанных на языке программирования Си с использованием ODBC-интерфейса:
  - приложение `odbctest` осуществляет соединение с базой данных и выполнение SQL-запроса, поданного из командной строки;
  - приложение `tpcb` реализует тест производительности SQL-сервера;
- 11) PCC – примеры программ, иллюстрирующих применение встроенного SQL-интерфейса СУБД ЛИНТЕР;
- 12) PERL – примеры работы с СУБД ЛИНТЕР из программ, написанных на языке программирования Perl. В файле `\samples\PERL\readme.txt` содержится информация об условиях, необходимых для работы примера;

- 
- 13) PHP – примеры работы с СУБД ЛИНТЕР из программ, написанных на языке программирования PHP. В файле `\samples\PHP\readme.txt` содержится информация об условиях, необходимых для работы примера;
  - 14) PROC, SP – примеры работы с хранимыми процедурами. В файлах `\samples\PROC\readme.txt` и `\samples\SP\readme.txt` содержатся инструкция по запуску примеров и описание используемых в примерах хранимых процедур;
  - 15) SQL – примеры SQL-запросов к СУБД. Все примеры выполняются на демонстрационной БД;
  - 16) TCL – примеры работы с СУБД ЛИНТЕР из программ, написанных на языке программирования TCL/TK;
  - 17) TRIGGERS – примеры работы с триггерами. В файле `\samples\triggers\readme.txt` содержится инструкция по запуску примеров и описание используемых в примерах триггеров.

---

# Краткий список утилит и динамических библиотек СУБД ЛИНТЕР

Наименование	Описание
dbc_tcp	Командный сетевой драйвер клиента
dbcgui	Графический сетевой драйвер клиента
dbcapi	Динамическая библиотека сетевого драйвера клиента
dbf2lin	Конвертер DBF-файлов
dbs_tcp	Командный сетевой драйвер сервера
dbsgui	Графический сетевой драйвер сервера
dbstore	Экспорт структур и данных
decimals, tick	Динамические библиотеки специальных типов данных
gendb	Создание и конфигурирование базы данных
inl	Командный интерфейс
inter/int64	Динамическая библиотека CALL-интерфейса
ldba	Администрирование базы данных
lhb	Архивирование и восстановление базы данных
lhbх	Графическая утилита архивирования и восстановления базы данных
libpng, libjpeg	Динамические библиотеки, используемые графическими утилитами СУБД ЛИНТЕР
lin2dbf	Программа конвертирования текстового файла в DBF-формат
linapid	JDBC-сервер
lindesk	Рабочий стол СУБД ЛИНТЕР
LinPerl.so	Динамическая библиотека для программ на языке Perl
LinPerl.pm	Программный интерфейс для языка Perl
linsnmpd	Агент SNMP-протокола
Linter.so	Динамическая библиотека для языка Perl
Linter.pm	Интерфейс DBI для языка Perl
linter	Головной модуль ядра СУБД ЛИНТЕР
loarel	Импорт данных
loltp	Управление распределенными транзакциями
lreplserver	Сервер репликации
migration	Миграция базы данных
pcc	Препроцессор встроенного SQL
libphp_linter.so	Динамическая библиотека для языка PHP
server	Сервер системы горячего резервирования
setlic	Программа установки лицензии
shut	Останов ядра СУБД ЛИНТЕР

---

<b>Наименование</b>	<b>Описание</b>
spc	Компилятор хранимых процедур
spman	Псевдографический отладчик триггеров и хранимых процедур
stproc	Пример программы слежения системы горячего резервирования
tdbx	Графическая утилита тестирования базы данных
testdb	Тестирование базы данных
ChangeLog.txt	Файл журнализации изменений проекта

# Приложение

## Объекты БД DEMO

Состав и взаимосвязи таблиц БД DEMO приведены на рисунке П.1.

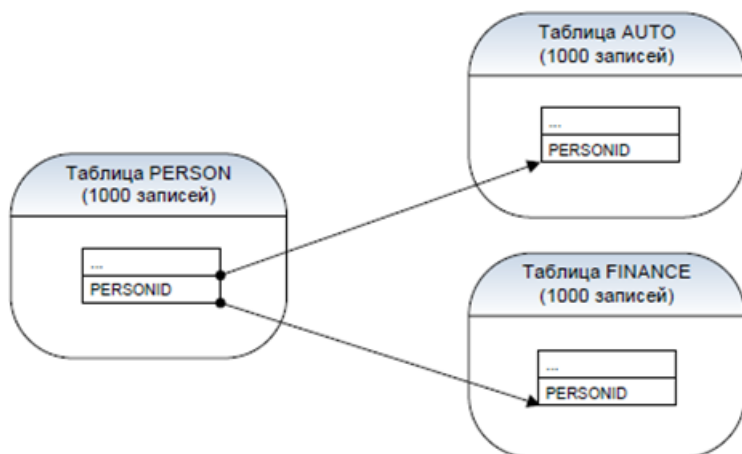


Рисунок П.1. Состав и взаимосвязи таблиц БД DEMO

### Таблица PERSON

```
create table person( /*Данные о людях */
NAME char(20), /*Фамилия */
FIRSTNAM char(15), /*Имя */
INITIALS char(1), /*Первая буква отчества */
SEX char(1), /*Пол */
AGE integer, /*Возраст */
FMLYSTAT char(1), /*Семейное положение */
DEPNDNITS integer, /*Количество иждивенцев */
NUMB integer, /*Номер дома */
STREET char(20), /*Улица */
CITY char(15), /*Город */
STATE char(2), /*Штат */
ZIP integer, /*Почтовый индекс */
PHONE char(8), /*Телефон */
JOB char(20), /*Место работы */
SALARY integer, /*Зарплата */
CMISSION integer, /*Процент с продаж */
YRSEUDC integer, /*Общая длительность учёбы */
YRSCMPNY integer, /*Стаж работы */
VACDAYS integer, /*Продолжительность отпуска */
SICKDAYS integer, /*Количество больничных дней*/
HOBBY char(3), /*Хобби */
PERSONID integer primary key) /*Идентификатор человека */
maxrow 2;
```



---

### Таблица AUTO

```
create table auto( /*Характеристики автомобилей */
MAKE char(20), /*Производитель */
MODEL char(20), /*Модель */
BODYTYPE char(15), /*Тип кузова */
CYLNDERS integer, /*Количество цилиндров */
HORSEPOWER integer, /*Мощность в лошадиных силах */
DISPLACEMENT integer, /*Рабочий объем двигателя */
WEIGHT integer, /*Вес */
COLOR char(10), /*Цвет */
YEAR integer, /*Год выпуска */
SERIALNO char(16), /*Серийный номер */
CHKDATE integer, /*Дата техосмотра */
CHKMILE integer, /*Пробег */
PERSONID integer primary key) /*Идентификатор владельца */
maxrow 2;
```

### Таблица FINANCE

```
create table finance( /*Характеристики банковских карт */
CRDITCRD char(18), /*Платежная система */
CRDITLIM integer, /*Кредитный лимит */
CURNTBAL integer, /*Текущий баланс */
OILCRD char(7), /*Топливная карта */
NETWORTH integer, /*Суммарная стоимость активов */
CRDITRAT integer, /*Кредитный рейтинг */
INSCMPNY char(25), /*Страховая компания */
PLCYAMNT integer, /*Поправка к страховому полису */
COLLEGE char(16), /*Учебное заведение */
ONVACTN char(1), /*Карта заблокирована */
INVSTMNT char(1), /*Инвестор */
BANK char(30), /*Банк карты */
PERSONID integer primary key) /*Идентификатор владельца */
maxrow 2;
```