

**МОБИЛЬНАЯ
РЕЛЯЦИОННАЯ
СУБД**

ЛИНТЕР®

Linter Standard
Linter Bastion
Linter RealTime
Linter Multiversion

Сетевые средства

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ

РЕЛЭКС®

Товарные знаки

РЕЛЭКСТМ, 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
Дополнительные документы	3
Общие сведения	4
Модель сетевой архитектуры СУБД ЛИНТЕР	5
ЛИНТЕР-сервер	5
Клиентское приложение	5
Программные сетевые средства	6
Сетевой драйвер клиента	6
Сетевой драйвер сервера	6
Файл сетевой конфигурации клиента	6
Переменные окружения ОС	7
Протоколы обмена данными	7
Механизм взаимодействия клиент-сервер	8
Сетевая прозрачность	8
Независимость от протоколов	8
Прозрачность местоположения	8
Организация сетевого доступа	9
Операции соединения/отсоединения	9
Операции обмена данными	9
Аварийное прекращение соединения	10
Разъединение по тайм-ауту	10
Варианты сетевых соединений	11
Локальный несетевой доступ	11
Локальный сетевой доступ	11
Удаленный сетевой доступ	12
Установка сетевых средств	13
Файл сетевой конфигурации	14
Структура файла	14
Подготовка файла	14

Протоколы обмена данными.....	16
TCP/IP.....	16
TCP/IPS.....	17
LOCAL.....	18
LOCALI, LOCALS.....	19
ATCP/IP.....	20
ATCP/IPS.....	21
IPX/SPX.....	21
Именованный канал.....	22
DECNET.....	23
TSR.....	23
NETBIOS.....	24
LASSP.....	25
Сетевые средства для Unix, QNX-платформ.....	27
Драйвер сервера.....	27
Назначение драйвера.....	27
Условия выполнения.....	27
Запуск драйвера.....	27
Ключи командной строки.....	27
Выполнение драйвера.....	30
Настройка защищенного соединения со стороны сервера.....	30
Драйвер клиента.....	32
Назначение драйвера.....	32
Условия выполнения.....	32
Запуск драйвера.....	32
Ключи командной строки.....	32
Выполнение драйвера.....	36
Настройка защищенного соединения со стороны клиента.....	37
Сетевые средства для Win32-платформ.....	39
Драйвер сервера.....	39
Запуск драйвера.....	39
Ключи командной строки.....	39
Функции драйвера.....	41
Настройка драйвера.....	41
Статистическая информация.....	41
Драйвер клиента.....	41
Условия выполнения.....	41
Запуск драйвера.....	42
Ключи командной строки.....	42
Функции драйвера.....	43
Выбор файла сетевой конфигурации.....	43
Сетевое конфигурирование.....	45
Настройка драйвера.....	48
Статистическая информация.....	50
Тестирование конфигурации сети.....	53
Тестирование на Unix-платформе.....	53
Тестирование на Win32-платформе.....	53

Коды завершения сетевых средств	54
Управление сетевым доступом.....	55
Активизация клиента.....	56
Удаленный клиент.....	56
Локальный клиент	57
Активизация сервера	57
Локальный сервер.....	57
Удаленный сервер	57
Способы соединения с ЛИНТЕР-сервером	60
Соединение из командной строки	60
Соединение с помощью экранной формы	61
Соединение через программный интерфейс.....	61
Приложение 1. Пример создания файла сетевой конфигурации	62
Приложение 2. Варианты сетевых конфигураций СУБД ЛИНТЕР	65
Приложение 3. Проверка работы сетевого клиента в среде ОС Unix	69

Предисловие

Назначение документа

Документ содержит описание сетевой инфраструктуры СУБД ЛИНТЕР.

Приведены сведения о конфигурировании, настройке и порядке запуска сетевых средств для обеспечения локального и/или удаленного доступа из клиентских приложений к одной или нескольким БД с использованием различных протоколов обмена данными.

Описаны возможные способы организации взаимодействия клиентского приложения и СУБД ЛИНТЕР.

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

Для кого предназначен документ

Документ предназначен для администраторов СУБД ЛИНТЕР и разработчиков клиентских ЛИНТЕР–приложений.

Необходимые предварительные знания



Для работы с сетевыми средствами необходимо:

- знать основы функционирования локальных вычислительных сетей (ЛВС);
- уметь работать в соответствующей операционной системе на уровне продвинутого пользователя (устанавливать переменные окружения, контролировать состояние процессов ОС и т. п.).

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.

Предисловие

<u>Обозначение</u>	<u>Пример</u>	<u>Значение</u>
Мелкий моноширинный текст	Page Time Count	Текст программы.
Заглавные буквы	BROWSE	Названия команд, слова, зарезервированные в SQL, ключевые слова.
Курсив в < >	<return statement>	Определяемый элемент синтаксической конструкции.
Символ ::=		Равенство по определению. Слева от знака стоит определяемое понятие, справа – собственно определение понятия.
Квадратные скобки []	DBSTORE [-d -n -o -p -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.

Дополнительные документы

- СУБД ЛИНТЕР. Архитектура СУБД.
- СУБД ЛИНТЕР. Справочник кодов завершения.
- СУБД ЛИНТЕР. Запуск и останов СУБД на платформе Unix, QNX.
- СУБД ЛИНТЕР. Запуск и останов СУБД на платформе Win32.

Общие сведения

Сетевые средства СУБД ЛИНТЕР предназначены для обеспечения «прозрачного» доступа клиентских приложений к базам данных (БД), расположенных как на удаленных, так и на локальных компьютерах вычислительной сети. Прозрачность доступа обеспечивается с помощью следующих механизмов:

- доступ к локальным или удаленным БД выполняется посредством одинакового интерфейса;
- отсутствует необходимость знать местонахождение БД.

Активизация сетевых средств может выполняться одним из следующих способов:

- автоматически при запуске операционной системы (ОС);
- автоматически при запуске ядра СУБД ЛИНТЕР;
- вручную;
- из клиентских приложений.

Модель сетевой архитектуры СУБД ЛИНТЕР

В основу сетевой архитектуры СУБД ЛИНТЕР положена модель «клиент-сервер», которая предполагает наличие трех программных компонентов:

- 1) ЛИНТЕР-сервер;
- 2) клиентское приложение;
- 3) программные сетевые средства.

ЛИНТЕР-сервер

ЛИНТЕР-сервер отвечает за прием запросов клиентских приложений, их обработку и передачу результатов выполнения запроса клиентскому приложению. В общем случае ЛИНТЕР-сервер представляет собой ядро СУБД ЛИНТЕР и сетевой драйвер сервера. Список доступных клиентскому приложению ЛИНТЕР-серверов хранится в файле сетевой конфигурации клиентского приложения. Клиентское приложение обращается к ЛИНТЕР-серверу либо по его имени (в случае доступа к конкретному ЛИНТЕР-серверу), либо без указания имени (в случае доступа к ЛИНТЕР-серверу по умолчанию).

Следует различать имя ЛИНТЕР-сервера и собственно имя БД, задаваемое при её создании с помощью утилиты `gendb`. Имя БД является ее атрибутом, и для всех клиентских приложений оно одинаково, в то время как в локальной сети одна и та же БД может быть видна разным клиентским приложениям под различными логическими (ЛИНТЕР-сервер) именами.

Для соединения клиентского приложения с сервером БД сетевые средства СУБД ЛИНТЕР используют прозрачные сетевые драйверы (клиента и сервера), промышленные и собственные стандарты сетевых протоколов.

Сетевые драйверы выполняют коммуникационные задачи через общие точки входа, которые не зависят от специфики используемых сетевых протоколов (будь то TCP/IP, DECNET, разделяемая память и т.п.).

Связь между клиентом и сервером обрабатывается по принципу стека.

Логической единицей обмена являются SQL-запросы и строки данных. На уровне CALL-интерфейса (интерфейс нижнего уровня) логические единицы обмена транслируются в серии команд к ядру СУБД.

Клиентское приложение

Клиентское приложение обеспечивает взаимодействие пользователя с ЛИНТЕР-сервером. Оно определяет SQL-операции, которые должен выполнить ЛИНТЕР-сервер, передает их через соответствующий программный интерфейс (программный интерфейс зависит от языка программирования, на котором разработано приложение – C/C++, PERL, PHP QT и др.) и обрабатывает полученные от ЛИНТЕР-сервера результаты.

По отношению к ЛИНТЕР-серверу клиентское приложение может быть локальным или удаленным.

Локальное клиентское приложение функционирует на том же компьютере, что и ЛИНТЕР-сервер. В этом случае необходимость в использовании протоколов обмена данными между клиентским приложением и ЛИНТЕР-сервером может отсутствовать.

Модель сетевой архитектуры

Удалённое клиентское приложение функционирует на отдельном от ЛИНТЕР-сервера компьютере. В этом случае клиентское приложение и ЛИНТЕР-сервер обмениваются информацией между собой с помощью сетевого протокола обмена данными.

 Чтобы разрешить удаленный доступ клиентских приложений к БД необходимо на ЛИНТЕР-сервере с этой БД:

1) выполнить команду:

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

(разрешить доступ к этой БД со всех компьютеров);

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

Программные сетевые средства

Программные сетевые средства обеспечивают обмен данными между клиентским приложением и ЛИНТЕР-сервером. В их состав входят:

- сетевой драйвер клиента;
- сетевой драйвер сервера;
- файл сетевой конфигурации клиента;
- переменные окружения ОС;
- протоколы обмена данными.

Сетевой драйвер клиента

Сетевой драйвер клиента – программное средство, устанавливаемое вместе с клиентским приложением и выполняющее следующие функции:

- прием запросов клиентских приложений к ЛИНТЕР-серверу;
- установление соответствия между именем ЛИНТЕР-сервера и его сетевым адресом;
- осуществление сетевого или локального соединения с ЛИНТЕР-сервером;
- передачу запросов клиентских приложений ЛИНТЕР-серверу через сеть или локально;
- прием результатов обработки запросов от ЛИНТЕР-сервера;
- передачу результатов обработки запросов клиентскому приложению.

Сетевой драйвер сервера

Сетевой драйвер сервера – программное средство, устанавливаемое в составе ЛИНТЕР-сервера и выполняющее следующие функции:

- осуществление локального соединения с ядром СУБД ЛИНТЕР;
- осуществление сетевого соединения с сетевыми драйверами клиентов;
- получение от сетевого драйвера клиента запросов клиентских приложений и передачу их ядру СУБД ЛИНТЕР;
- получение от ядра СУБД ЛИНТЕР результатов выполнения запросов и передачу их драйверу клиента.

Файл сетевой конфигурации клиента

Файл сетевой конфигурации клиента – текстовый файл (со стандартным именем nodetab), размещаемый на том же компьютере, где запускается клиентское приложение,

и описывающий сетевую конфигурацию СУБД ЛИНТЕР с точки зрения этого клиентского приложения (имена доступных ЛИНТЕР-серверов, их сетевые адреса и другие параметры, необходимые сетевому драйверу клиента для установления сетевого соединения с конкретным ЛИНТЕР-сервером).

Переменные окружения ОС

Переменные окружения ОС – это переменные окружения операционной системы, используемые для настройки работы сетевых средств СУБД ЛИНТЕР и клиентских приложений.

Протоколы обмена данными

Протоколы обмена данными – программные средства операционной системы, используемые для обмена данными между сетевым драйвером клиента и сетевым драйвером сервера либо непосредственно между клиентским приложением и ядром СУБД ЛИНТЕР. Настраиваемые характеристики протоколов обмена задаются с помощью *параметров сетевого доступа* (например, величина тайм-аута сетевого соединения), которые хранятся в файле сетевой конфигурации `nodetab` и задаются индивидуально для каждого ЛИНТЕР-сервера. Формат представления параметров сетевого доступа определяется используемым сетевым протоколом.

Механизм взаимодействия клиент-сервер

Сетевые средства СУБД ЛИНТЕР обеспечивают *прозрачный* механизм доступа клиентских приложений к ЛИНТЕР-серверам. Это означает, что клиентское приложение не обязано содержать внутри себя или запрашивать извне информацию о местоположении ЛИНТЕР-сервера в локальной сети и его сетевых параметрах доступа, кроме, возможно, самого имени ЛИНТЕР-сервера, к которому необходим доступ. (В ряде случаев клиентские приложения могут работать с ЛИНТЕР-сервером, даже не зная его имени, т.е. с «ЛИНТЕР-сервером по умолчанию»).

Сетевая прозрачность

Клиентские приложения, функционирующие на локальном ЛИНТЕР-сервере, могут быть развернуты в сети для доступа к аналогичному ЛИНТЕР-серверу без каких-либо изменений в тексте приложений. С точки зрения разработчика или пользователя приложения, весь обмен данными через сетевые средства невидим для пользователя приложения. Кроме того, можно изменять структуру сети для приложения, не изменяя самого приложения. Эта способность оставаться невидимым известна как сетевая прозрачность.

Независимость от протоколов

Сетевые средства СУБД ЛИНТЕР обеспечивают клиентским приложениям независимость от протоколов обмена данными. Сетевые средства могут работать с большинством сетевых протоколов, поддерживаемых операционными системами. Любое приложение, разработанное на любом компьютере, использовавшем любой протокол, может быть перенесено без изменений на другие компьютеры, использующие иные протоколы.

Прозрачность местоположения

Сетевые средства СУБД ЛИНТЕР обеспечивают прозрачность местоположения. Это означает, что доступ из клиентского приложения к ЛИНТЕР-серверу осуществляется только по имени сервера, никакой дополнительной информации о физическом местоположении сервера приложению не требуется.

Организация сетевого доступа

Операции соединения/отсоединения

Операция соединения между сетевыми драйверами клиента и сервера инициируется во время передачи любого запроса к ЛИНТЕР-серверу. Для соединения с ЛИНТЕР-сервером необходимо указать его имя в соответствующих полях программных интерфейсов или в опциях утилит. Если имя ЛИНТЕР-сервера не указывается, то выполняется соединение с ЛИНТЕР-сервером *по умолчанию*. ЛИНТЕР-сервер по умолчанию – это локальный ЛИНТЕР-сервер или, если локальный ЛИНТЕР-сервер отсутствует, один из ЛИНТЕР-серверов, перечисленный в файле сетевой конфигурации.

Разрыв соединения между сетевыми драйверами клиента и сервера происходит в следующие моменты:

- завершение работы всех клиентских приложений;
- завершение работы сетевого драйвера сервера или клиента;
- системная проблема с сетевым соединением (аппаратная или программная);
- завершение работы удалённого ЛИНТЕР-сервера.

При проблемах с сетевым соединением или при завершении работы удаленного ЛИНТЕР-сервера клиентскому приложению передаются соответствующие коды завершения в диапазоне 4000 – 4999.

О проблемах с сетевым соединением сообщает ОС в коде завершения соответствующего системного вызова, или же эти проблемы обнаруживаются сетевыми драйверами самостоятельно по истечении тайм-аутов получения тестовых посылок.

Следует различать сетевое соединение между сетевыми драйверами клиента и сервера и соединение клиентского приложения с ЛИНТЕР-сервером. По одному сетевому соединению может быть открыто несколько соединений (каналов) к ЛИНТЕР-серверу, возможно, даже разными клиентскими приложениями.

При разрыве сетевого соединения происходит закрытие всех каналов, по которым происходила работа приложений на данном компьютере. Каналы приложения также закрываются в случае завершения работы самого приложения.

Операции обмена данными

Сетевые средства СУБД ЛИНТЕР реализуют следующий механизм обмена данными:

- клиентское приложение передает SQL-запрос пользовательскому интерфейсу, встроенному в это приложение;
- пользовательский интерфейс направляет запрос непосредственно ядру СУБД ЛИНТЕР или сетевому драйверу клиента;
- сетевой драйвер клиента передает по сети запрос сетевому драйверу сервера;
- сетевой драйвер сервера переадресует запрос ядру СУБД ЛИНТЕР;
- ядро СУБД ЛИНТЕР обрабатывает запрос и результаты его выполнения возвращает сетевому драйверу сервера;
- сетевой драйвер сервера передает по сети результаты выполнения запроса сетевому драйверу клиента;
- сетевой драйвер клиента передает полученные данные клиентскому приложению.

Запросы клиентского приложения при работе через сеть могут быть как синхронными, так и асинхронными (аналогично работе с локальным ЛИНТЕР-сервером).

Аварийное прекращение соединения

Аварийное прекращение соединения возникает в ситуации, когда клиентское приложение аварийно завершает свою работу, не успев проинформировать сетевые средства о необходимости закончить соединение с ЛИНТЕР-сервером. В этом случае сетевые средства сбрасывают операции клиента и сервера, что приведет к разъединению текущей операции.

Разъединение по тайм-ауту

Разъединение по тайм-ауту возникает при обнаружении зависшего соединения. Сетевой драйвер сервера периодически (с заданным при конфигурировании сетевого доступа интервалом) посылает тестовый пакет. Если соединение не работает (обычно из-за недоступности клиентского процесса или компьютера), оно закрывается, клиентскому приложению передается соответствующий код завершения.

Варианты сетевых соединений

Локальный несетевой доступ

Конфигурация локального несетевого доступа приведена на Рис. 1. В этом случае клиентское приложение и ЛИНТЕР-сервер (БД и ядро СУБД ЛИНТЕР) размещаются на одном компьютере.

Конфигурационный файл nodetab не требуется (доступ осуществляется по умолчанию к установленному на компьютере ЛИНТЕР-серверу).

Сетевые средства (сетевые драйверы клиента и сервера) не используются.

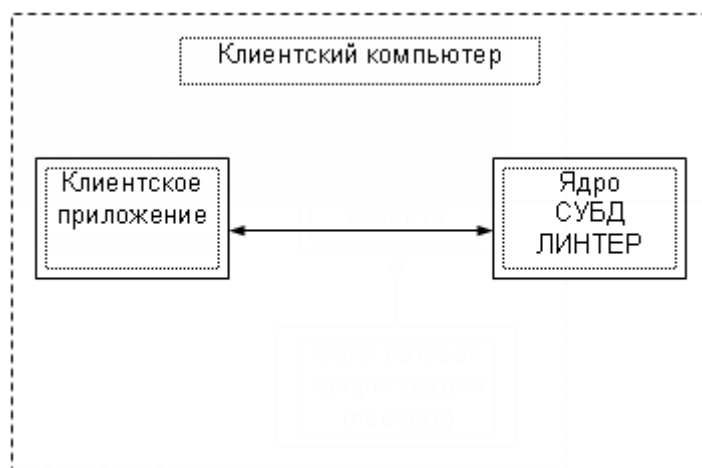


Рис. 1. Конфигурация локального несетевого доступа

Локальный сетевой доступ

Конфигурация локального сетевого доступа приведена на Рис. 2. В этом случае клиентское приложение и один или несколько ЛИНТЕР-серверов (БД и экземпляры ядра СУБД ЛИНТЕР) размещаются на одном компьютере.

Используется один сетевой драйвер клиента и столько экземпляров сетевого драйвера сервера, сколько установлено на компьютере БД, и экземпляров СУБД ЛИНТЕР (одной и той же или разных версий).

Конфигурационный файл nodetab требуется.

Пример конфигурационного файла:

ИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера	Тайм-аут клиента	Тайм-аут соединения
А	ТСРІР	195.98.69.200				
В	ТСРІРС	195.98.69.200	1061	2		

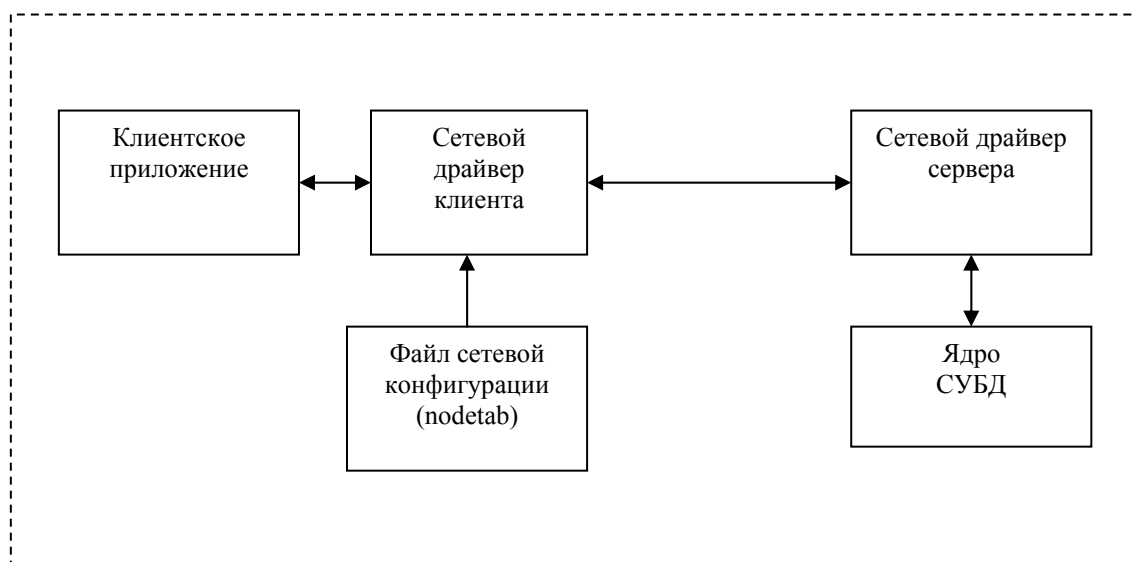


Рис. 2. Конфигурация локального сетевого доступа

Удаленный сетевой доступ

Конфигурация удаленного сетевого доступа приведена на Рис. 3. В этом случае клиентское приложение и один или несколько ЛИНТЕР-серверов (БД и экземпляры ядра СУБД ЛИНТЕР) размещаются на разных компьютерах.

Конфигурационный файл nodetab требуется.

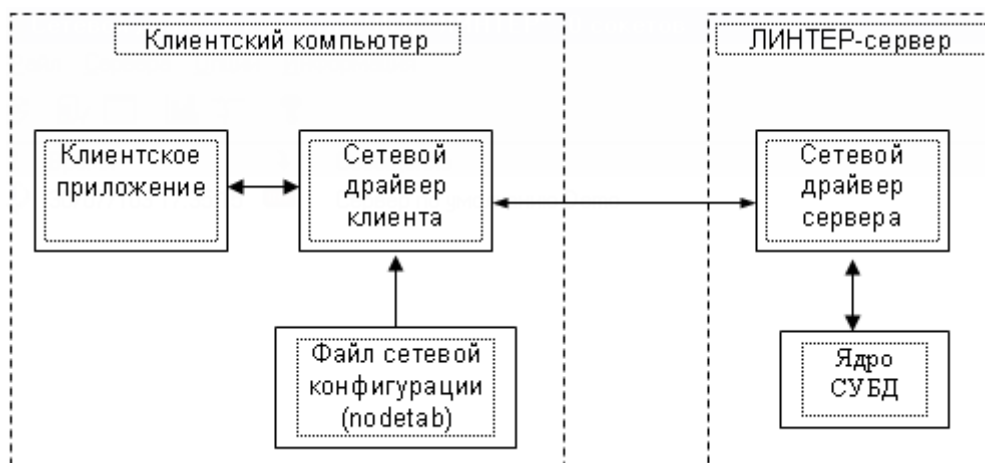


Рис. 3. Конфигурация удаленного сетевого доступа

Установка сетевых средств

На платформе ОС типа Win32 установка сетевых средств выполняется выборочно в процессе установки СУБД ЛИНТЕР путем выставления флажка Сетевые драйверы в окне «Выбор компонентов» (см. документ «СУБД ЛИНТЕР. Установка на платформе Win32»).

На платформе ОС типа Unix сетевые средства устанавливаются всегда при установке СУБД ЛИНТЕР (независимо от того, будут или нет они использоваться в дальнейшем) (см. документ «СУБД ЛИНТЕР. Установка на платформе типа Unix, QNX»).

Файл сетевой конфигурации

Структура файла

Для управления доступом клиентских приложений к БД используется файл сетевой конфигурации клиента – текстовый файл с именем `nodetab`. Строки файла описывают сетевые параметры ЛИНТЕР-серверов и, при необходимости, системы горячего резервирования СУБД ЛИНТЕР. Каждая строка разделяется на поля (Рис. 4). Разделительным символом полей являются пробел или символ табуляции. Каждое поле имеет свое назначение и устанавливает один из параметров сетевого соединения ЛИНТЕР-сервера.

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера	Тайм-аут клиента	Тайм-аут соединения
---------------	------------------	---------------	--------------	------------------	------------------	---------------------

Рис. 4. Поля строки файла `nodetab`

Обязательными полями являются:

- ЛИНТЕР-сервер;
- сетевой протокол;
- сетевой адрес.

В строке могут присутствовать и другие поля, необходимые для работы сетевых компонентов СУБД ЛИНТЕР:

- сетевой порт;
- тайм-аут сервера;
- тайм-аут клиента;
- тайм-аут соединения.

Формат и значение этих полей зависит от типа используемого протокола обмена данными (см. раздел «Протоколы обмена данными»).

Поле Тайм-аут сервера задает максимально допустимое время (в мин.), после которого сетевой драйвер сервера будет считать, что сетевое соединение с драйвером клиента разорвано. При сетевой работе драйвер клиента посылает сетевому драйверу сервера тестовые послылки несколько чаще, чем этот интервал. В случае, если драйвер сервера не получит тестового пакета от драйвера клиента в течение времени, указанного в поле Тайм-аут сервера, соединение будет закрыто драйвером сервера с генерацией соответствующего кода завершения. Значение тайм-аута по умолчанию: 1 минута. При явном задании нулевого тайм-аута тестовые послылки драйвером клиента посылаться не будут, и в случае разрыва соединения длительность обнаружения разрыва будет определяться реализацией протокола в ОС.

Поле Тайм-аут клиента задает интервал времени (в сек.), через который драйвер клиента считает соединение с драйвером сервера разорванным в случае неполучения от драйвера сервера тестового пакета. Значение по умолчанию: 0, т. е. тестовые пакеты не посылаются, и состояние разрыва соединения драйвером клиентом не обнаруживается.

Поле Тайм-аут соединения задает время (в сек.), по истечении которого будет фиксироваться ошибка, если сетевое соединение между драйверами клиента и сервера установить не удалось. По умолчанию значение равно 0, т.е. определяется ОС.

Подготовка файла

Для Unix-платформ файл `nodetab` вводится вручную с помощью любого текстового редактора, имеющегося в операционной системе.

На платформе Win32 он может быть подготовлен следующими способами:

- вручную;
- в процессе установки СУБД ЛИНТЕР на клиентском компьютере (см. документ «СУБД ЛИНТЕР. Установка СУБД на Win32-платформе»);
- для сетевой Win32 – с помощью утилиты linadm (см. документ «СУБД ЛИНТЕР. Сетевой администратор»);
- с помощью сетевого драйвера клиента с графическим интерфейсом (см. раздел «Драйвер клиента»).

Правила ввода файла nodetab:

- 1) обязательными полями являются:
 - имя ЛИНТЕР-сервера (не более 8 символов в латинской кодировке);
 - имя сетевого протокола;
 - сетевой адрес.
- 2) необязательным полям с незаполненными значениями присваиваются значения по умолчанию;
- 3) неправильно введенное значение (например, вместо цифрового значения буквенное) игнорируется, сообщение об ошибке не выдается, и неправильное значение заменяется значением по умолчанию;
- 4) каждая строка файла должна заканчиваться символом «переход на новую строку»;
- 5) разделителем столбцов файла могут быть пробел и знаки табуляции (вследствие этого задаваемые в таблице имена не должны содержать пробелов);
- 6) файл допускает ввод комментариев. Признаком начала комментария является символ #. Комментарием считаются все символы от начала комментария до конца строки файла.

Протоколы обмена данными

С функциональной точки зрения протоколы обмена можно разделить на следующие группы:

- локальные протоколы, предназначенные для обмена данными между клиентским приложением и локальным ЛИНТЕР-сервером. В эту группу входят протоколы: LOCAL, LOBALI, LOCALS, NPIPE;
- сетевые протоколы, предназначенные для обмена данными между клиентским приложением и удалённым ЛИНТЕР-сервером. В частном случае сетевые протоколы могут использоваться для организации одновременного доступа к нескольким локальным ЛИНТЕР-серверам. В эту группу входят протоколы: TCP/IP, TCP/IPS, IPX/SPX, DECNET, TSR, NETBIOS, NPIPE;
- протоколы системы репликации (тиражирования) данных, предназначенные для обмена данными между реплицируемыми серверами. Клиентскому приложению эти протоколы недоступны. В эту группу входят протоколы: ATCP/IP, ATCP/IPS;
- протоколы резервного доступа, предназначенные для доступа к одному из нескольких ЛИНТЕР-серверов именованной группы. В эту группу входит протокол LASSP.

TCP/IP

Сетевой протокол Internet.

Описание в конфигурационном файле

ЛИНТЕР-сервер

Имя удалённого ЛИНТЕР-сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: TCP/IP .

Поле обязательное.

Сетевой адрес

Сетевой IP-адрес или имя, определяемое через службу DNS (Domain Name Service). Например, 100.101.102.103 – числовой IP-адрес, mycomp.myorg.mydomen – полное DNS-имя, mycomp – краткое DNS-имя.

Поле обязательное.

Сетевой порт

Номер порта TCP/IP удаленного сервера. Задаётся в десятичном или шестнадцатеричном виде.

Поле необязательное. По умолчанию используется номер порта 1060 (0x424).

Если на удалённом компьютере запущено несколько сетевых драйверов сервера, использующих несколько портов, то необходимо явно указывать каждый порт, за исключением, может быть, одного.

Поле обязательно для заполнения, если сетевой драйвер сервера на удалённом компьютере «слушает» порт, отличный от порта 1060.

В ОС Unix для получения более наглядной информации о портах TCP/IP по команде netstat можно внести в файл /etc/services строку:

1060 tcip Linter

Тайм-аут сервера

Тайм-аут обнаружения разрыва соединения сетевым драйвером сервера (мин.).

Поле необязательное. По умолчанию используется значение 1 мин.

При указании значения 0 тайм-аут игнорируется, и разрыв соединения диагностируется средствами ОС.

Тайм-аут клиента

Тайм-аут обнаружения разрыва соединения сетевым драйвером клиента (сек.).

Поле необязательное. По умолчанию используется значение 0, т. е. тайм-аут определяется ОС.

Величина этого тайм-аута должна быть больше 2 секунд.

Тайм-аут соединения

Тайм-аут ожидания установки соединения сетевым драйвером клиента (сек.).

Поле необязательное. По умолчанию используется значение 0, т. е. тайм-аут определяется ОС.

Применимость протокола для программных платформ

Все операционные системы, поддерживающие TCP/IP.

Пример

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера, мин.	Тайм-аут клиента, сек.	Тайм-аут соединения, сек.
SERV1	TCPIP	192.168.69.226	1060	1	4	
SERV2	TCPIP	195.98.69.227	0x425			
SERV3	TCPIP	Accountant	1062	2	20	4
SERV4	TCPIP	Accountant.com				

TCP/IPS

Протокол TCP/IP, защищенный SSL.

Описание в конфигурационном файле

ЛИНТЕР-сервер

Имя ЛИНТЕР-сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: TCP/IPS .

Поле обязательное.

Сетевой порт

Значение идентично протоколу TCP/IP.

Тайм-аут сервера

Значение идентично протоколу TCP/IP.

Протоколы обмена данными

Тайм-аут клиента

Значение идентично протоколу TCP/IP.

Тайм-аут соединения

Значение идентично протоколу TCP/IP.

Применимость протокола для программных платформ

Возможно применение для всех Unix-подобных ОС. В некоторых поставках может не включаться. Нецелесообразно применение во встроенных системах из-за значительных накладных расходов на реализацию SSL.

Пример

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера, мин.	Тайм-аут клиента, сек.	Тайм-аут соединения, сек.
SERV1	TCPIPS	195.98.69.226	1060	1		
SERV2	TCPIPS	195.98.69.227	0x425			
SERV3	TCPIPS	Accountant	1062	2	20	4

LOCAL

Протокол для доступа к нескольким локальным ЛИНТЕР-серверам из одного клиентского приложения. С помощью протокола LOCAL клиентское приложение может обращаться напрямую к одному ЛИНТЕР-серверу, а к остальным – через сетевой драйвер клиента без участия сетевого драйвера сервера. Возможна также работа со всеми ЛИНТЕР-серверами через протокол LOCAL.

Описание в конфигурационном файле

ЛИНТЕР-сервер

Имя ЛИНТЕР-сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: LOCAL.

Поле обязательное.

Сетевой адрес

Задает идентификатор механизма межпроцессного взаимодействия. Аналог переменной окружения LINTER_MBX. Осуществляет доступ к СУБД, при запуске которой LINTER_MBX принимала точно такое же значение, как в этом поле (см. документ «Запуск и останов СУБД ЛИНТЕР»).

Поле обязательное.

Сетевой порт

Не задается.

Тайм-аут сервера

Не задается.

Тайм-аут клиента

Не задается.

Тайм-аут соединения

Не задается.

Применяемость протоколов для программных платформ

ОС Unix, QNX. Не поддерживается ОС VxWorks и ОС PV, на которых может выполняться только один экземпляр СУБД.

Пример

1. На компьютере запускаются два экземпляра СУБД.

Файл nodetab:

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера, мин.	Тайм-аут клиента, сек.	Тайм-аут соединения, сек.
SERV1	LOCAL	1234				

Запуск СУБД и сетевого драйвера клиента:

```
>linter /base=/first/base
>LINTER_MBX=1234
>export LINTER_MBX
>linter /base=/second/base
>unset LINTER_MBX
>dbc_tcp
```

2. Обращение к СУБД для БД /first/base

```
>inl
```

3. Обращение к СУБД для БД /second/base

```
>inl -n SERV1
```

LOCALI, LOCALS

Протокол, позволяющий клиентскому приложению обращаться к нескольким локальным ядрам СУБД ЛИНТЕР, использующим IPC-механизмы межпроцессного обмена для LOCALI или Unix domain sockets для LOCALS.

Если для межпроцессного обмена драйвер клиента и клиентское приложение используют механизм IPC SYSTEMV, то протокол LOCALI является синонимом протокола LOCAL.

Если для межпроцессного обмена драйвер клиента и клиентское приложение используют Unix domain sockets, то протокол LOCALS является синонимом протокола LOCAL.

Все сказанное о протоколе LOCAL применимо к LOCALI и LOCALS. Протоколы LOCALI и LOCALS выделены в отдельные группы для возможности одновременного доступа к СУБД с разными механизмами обмена.

Описание в конфигурационном файле

См. описание протокола LOCAL.

Применяемость протоколов для программных платформ

Только Unix-платформы и QNX.

АТСР/IP


Протокол АТСР/IP предназначен для использования системой репликации (асинхронного тиражирования), поскольку сервер репликации и сетевой драйвер клиента имеют общий файл конфигурации.

Описание в конфигурационном файле

ЛИНТЕР–сервер

Имя ЛИНТЕР–сервера, в БД которого должны тиражироваться данные. Это имя должно совпадать с именем сервера, создаваемым в БД ЛИНТЕР посредством запроса вида `create server`.

Поле обязательное.

 Количество строк с протоколом АТСР/IP в файле `nodetab` должно соответствовать количеству ЛИНТЕР–серверов, на которые выполняется тиражирование данных (см. документ «СУБД ЛИНТЕР. Репликация данных»).

Сетевой протокол

Обозначение протокола: АТСРIP.

Поле обязательное.

Сетевой адрес

Сетевой IPv4-адрес или имя, определяемое через службу DNS (Domain Name Service). Например, `100.101.102.103` – числовой IP-адрес, `myscomp.myorg.mydomain` – полное DNS-имя, `myscomp` – короткое DNS-имя.

Поле обязательное.

Сетевой порт

Номер TCP/IP порта удаленного сервера репликации. Задается в десятичном или шестнадцатеричном виде.

Поле необязательное. По умолчанию используется номер порта 1080 (0x438).

Номер должен совпадать с номером порта, указываемым при запуске сервера репликации на принимающей стороне.

Тайм-аут сервера

Поле не используется.

Тайм-аут клиента

Поле не используется.

Тайм-аут соединения

Поле не используется.

Применяемость протокола для программных платформ

Все операционные системы, на которых устанавливается система репликации СУБД ЛИНТЕР.

Примеры

См. документ «СУБД ЛИНТЕР. Репликация данных».

ATCP/IP

Протокол TCP/IP для использования в системе репликации (асинхронного тиражирования), защищенный SSL. Полностью аналогичен протоколу ATCP/IP за исключением преобразования данных, передающихся по сети.

Описание в конфигурационном файле

См. описание протокола ATCP/IP.

IPX/SPX

Сетевой протокол IPX/SPX.

Описание в конфигурационном файле

ЛИНТЕР–сервер

Имя ЛИНТЕР–сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: IPXSPX.

Поле обязательное.

Сетевой адрес

Задаётся в виде шестнадцатеричных цифр, из которых первые 8 соответствуют адресу сети, остальные 12 – адресу сетевой карты. Если в адресе сети указан 0, то происходит обращение к сети по умолчанию, если 1, то обращение будет к NetWare–серверу, который контролирует эту сеть.

Поле обязательное.

Сетевой порт

Номер сетевого порта сервера в диапазоне от 1 до 65535.

Поле необязательное. По умолчанию - 0x4848.

Тайм-аут сервера

Поле необязательное. По умолчанию используется значение 1 мин.

Тайм-аут клиента

Поле необязательное. По умолчанию используется значение 0.

Тайм-аут соединения

Поле необязательное. По умолчанию используется значение 0.

Применяемость протокола для программных платформ

Только в среде Win32 и MS DOS.

Пример

Файл nodetab:

ЛИНТЕР–сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм–аут сервера, мин.	Тайм–аут клиента, сек.	Тайм–аут соединения, сек.
serv1	IPXSPX	0000001500000000000001	0x4848	1		

Именованный канал

Протокол межпроцессного обмена.

Именованные каналы (named pipes) используются для гарантированной потоковой передачи данных по сети (можно сказать, что это выделенная линия для обмена данными между процессами). Идентификация именованного канала выполняется с помощью присвоения ему имени (аналогично имени файла).

Именованные каналы можно использовать как для локального, так и для удалённого доступа к ЛИНТЕР-серверу.

Описание в конфигурационном файле

ЛИНТЕР-сервер

Имя ЛИНТЕР-сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: NPIPE

Поле обязательное.

Сетевой адрес

Имя канала. Символьная строка, сформированная по правилам задания объектов Win32.

Поле обязательное.

Сетевой порт

Не задаётся.

Тайм-аут сервера

Не задаётся. Используется значение ОС по умолчанию.

Тайм-аут клиента

Не задаётся. Используется значение ОС по умолчанию.

Тайм-аут соединения

Не задаётся. Используется значение ОС по умолчанию.

Применяемость протокола для программных платформ

Только в среде Win32.

Примеры

Файл nodetab:

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера, мин.	Тайм-аут клиента, сек.	Тайм-аут соединения, сек.
SERV1	NPIPE	DEMO				
SERV2	NPIPE	BANK				

DECNET

Сетевой протокол DECNET.

Описание в конфигурационном файле

ЛИНТЕР–сервер

Имя ЛИНТЕР–сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: DECNET.

Поле обязательное.

Сетевой адрес

Комбинация из двух 16-битных чисел, разделённых точкой.

Сетевой порт

Шестнадцатибитное число.

Тайм-аут сервера

Поле необязательное. По умолчанию используется значение 1 мин.

Тайм-аут клиента

Поле необязательное. По умолчанию используется значение 0.

Тайм-аут соединения

Поле необязательное. По умолчанию используется значение 0.

Применяемость протокола для программных платформ

Только в среде ОС VMS.

TSR

Протокол обращения из программ Win 3.x к DOS-ядру ЛИНТЕР. Протокол реализован на основе резидентных программ (TSR).

Описание в конфигурационном файле

ЛИНТЕР–сервер

Имя ЛИНТЕР–сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: TSR.

Поле обязательное.

Сетевой адрес

Не задаётся.

Сетевой порт

Не задаётся.

Тайм-аут сервера

Не задаётся. Используется значение ОС по умолчанию.

Протоколы обмена данными

Тайм-аут клиента

Не задаётся. Используется значение ОС по умолчанию.

Тайм-аут соединения

Не задаётся. Используется значение ОС по умолчанию.

Применяемость протокола для программных платформ

Только в среде ОС Win 3.x.

Пример

Файл nodetab:

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера, мин.	Тайм-аут клиента, сек.	Тайм-аут соединения, сек.
SERV1	TSR					

NETBIOS

Сетевой протокол NETBIOS.

Описание в конфигурационном файле

ЛИНТЕР-сервер

Имя ЛИНТЕР-сервера.

Поле обязательное.

Сетевой протокол

Обозначение протокола: NETBIOS.

Поле обязательное.

Сетевой адрес

16-ти символьное имя, которое не должно начинаться с символа *.

Сетевой порт

Номер порта для сессий – 139.

Тайм-аут сервера

Поле необязательное. По умолчанию используется значение 1 мин.

Тайм-аут клиента

Поле необязательное. По умолчанию используется значение 0.

Тайм-аут соединения

Поле необязательное. По умолчанию используется значение 0.

Применяемость протокола для программных платформ

Только в среде ОС Win 3.x, Win32.

Примеры

Файл nodetab:

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера,	Тайм-аут клиента,	Тайм-аут соединения,
				сервера,	клиента,	соединения,

				мин.	сек.	сек.
SERV1	NETBIOS	DEMO<00>	139			
SERV2	NETBIOS	COMP<20>				
SERV3	NETBIOS	GROUP<1E>				

LASSP

Протокол доступа к одному из нескольких ЛИНТЕР-серверов в заданной группе.

По протоколу LASSP (Linter Automated Standby Server Protocol) осуществляется попытка соединения со всеми ЛИНТЕР-серверами, перечисленными в строке файла nodetab для этого протокола. При успешном соединении с первым ЛИНТЕР-сервером группы обмен данными выполняется через это соединение. При разрыве соединения, через которое в данный момент выполняется обмен данными, осуществляется попытка автоматического переключения на следующий (по порядку) ЛИНТЕР-сервер группы, принадлежащий данному протоколу LASSP. Если попытки соединения со всеми ЛИНТЕР-серверами группы оказались неудачными, клиентскому приложению возвращается соответствующий код завершения.

Описание в конфигурационном файле

ЛИНТЕР-сервер

Условное имя группы ЛИНТЕР-серверов.

Поле обязательное.

Сетевой протокол

Обозначение протокола: LASSP.

Поле обязательное.

Сетевой адрес

Список имен ЛИНТЕР-серверов, разделенных пробелами или символами табуляции. Перечисленные в данном списке имена должны быть описаны в этом же файле nodetab в строках, предшествующих строке с описанием протокола LASSP. Не допускается включение в группу ЛИНТЕР-серверов для одного LASSP-протокола ЛИНТЕР-серверов из других групп LASSP-протоколов: каждый ЛИНТЕР-сервер должен включаться только в одну группу.

Сетевой порт

Не задается.

Тайм-аут сервера

Не задается.

Данный протокол не имеет собственных средств ограничения времени соединения и обнаружения разрыва соединения, поэтому необходимо обязательно устанавливать индивидуальные тайм-ауты каждого из ЛИНТЕР-серверов, входящих в группу. Если хотя бы один ЛИНТЕР-сервер не будет иметь тайм-аутов, то до возврата кода завершения клиентскому приложению и обнаружения разрыва соединения может пройти большой интервал времени.

Тайм-аут клиента

Не задается. См. пункт Тайм-аут сервера данного протокола.

Тайм-аут соединения

Не задается. См. пункт Тайм-аут сервера данного протокола.

Применяемость протокола для программных платформ

Только в среде ОС семейства Unix.

Примеры

Файл nodetab:

ЛИНТЕР-сервер	Сетевой протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера, мин.	Тайм-аут клиента, сек.	Тайм-аут соединения, сек.
SERV1	TCPIP	195.200.56.40	1060	1	20	10
SERV2	LOCAL	1234				
SERV3	TCPIPS	195.200.56.50	1060	1	20	10
MAIN	LASSP	SERV1 SERV2 SERV3				

Сетевые средства для Unix, QNX-платформ

Драйвер сервера

Назначение драйвера

Сетевой драйвер сервера предназначен для обслуживания запросов клиентов на ЛИНТЕР-сервере.

Условия выполнения

Для работы драйвера в ОС должен быть установлен протокол TCP/IP.

Запуск драйвера

Команда запуска драйвера:

```
dbb_tcp [<ключ> ...]  
<ключ> ::= <вариант 1> | <вариант 2>  
<вариант 1> ::= -<код ключа>{<пробел> | = }<значение ключа>  
<вариант 2> ::= /<код ключа>=<значение ключа>
```

Ключи командной строки

Драйвер обрабатывает следующие ключи:

-h | -?

Подсказка по всем ключам драйвера.

-P= [<сетевой адрес>:]<номер порта>

Номер порта для протокола TCP/IP. Через этот порт удаленные клиенты будут соединяться с ЛИНТЕР-сервером с помощью данного драйвера. По умолчанию <номер порта> равен 1060. Драйвер сервера можно привязывать не только к порту, но и к сетевому адресу (адресу интерфейса). В этом случае адрес должен задаваться только в числовом виде.



Разрешена работа только с одним адресом интерфейса.

Пример

```
dbb_tcp -P 127.0.0.1:1060  
(драйвер будет работать только с локальным интерфейсом).
```

-M=<идентификатор очереди>

Номер идентификатора очереди сообщений СУБД ЛИНТЕР. Если ключ не задан, то значение идентификатора очереди берется из переменной окружения LINTER_MBX, а если не определена и эта переменная, по умолчанию принимается значение 20561.

-D

Флаг автоматического запуска драйвера из системной сетевой программы `inetd`. Ключ задается в файле настройки `inetd.conf`. Структура таблицы `inetd.conf` приведена в документации на ОС Unix.

-K=<сигнал>

Заставляет драйвер посылать указанный <сигнал> родительскому процессу по окончании своей инициализации. Значение <сигнала> должно быть целым положительным числом. В случае отсутствия аргумента или его неверного значения по умолчанию посылается сигнал `SIGTERM`.

-C

Запрещает переводить процесс драйвера в фоновый режим. Драйвер в этом случае не освобождает консоль до своего завершения.

-S


Заставляет использовать дочерний процесс при каждом новом соединении.

-W

Заставляет отслеживать активность ядра СУБД ЛИНТЕР. Если ядро стало неактивным, драйвер также завершает свою работу.

-I=<спецификация файла>

Заставляет записывать идентификатор процесса (PID) драйвера сервера в текстовый файл. Если заданный файл существует, он будет перезаписан. Файл будет уничтожен при нормальном завершении работы драйвера.

 Информация о PID драйвера в текстовом файле требуется, как правило, при обработке командных файлов (например, для того, чтобы завершить работу драйвера).

-N

Запрещает сообщать ядру СУБД ЛИНТЕР о разрыве сетевого соединения. При последующем восстановлении соединения драйвер продолжит работу, будто разрыва соединения не было.

-VERSION

Вывод на консоль информации о версии драйвера.


-SSLONLY

Запрет на передачу данных по сети в немаскируемом режиме (по незащищенному соединению). В режиме маскирования исключается возможность получения информации путем прослушивания сетевого соединения. Если будет предпринята попытка переслать данные по незащищенному соединению, то это соединение будет закрыто.

-SSLKEY=<длина ключа>

Генерировать и запоминать ключи маскирования (пара «ключ-сертификат») сетевого сервера в файлах `db_s_tcp.key` и `db_s_tcp.crt` текущего каталога. Если ключи маскирования не были сохранены (файлы `db_s_tcp.key` и `db_s_tcp.crt` отсутствуют), то они будут генерироваться заново при каждом запуске драйвера. Рекомендуемое значение длины ключа – в диапазоне 2024-4096. В случае копирования файла `db_s_tcp.crt` на сторону клиента под другим именем идентификация сетевого сервера сетевым клиентом возможна во время установки соединения. Для этого необходимо, чтобы приведенное к верхнему регистру имя файла с расширением `.crt` на стороне клиента совпадало с именем ЛИНТЕР-сервера, с которым устанавливается соединение, из файла сетевой конфигурации `nodetab`.


Длина генерируемого ключа указывается в битах. Если значение не задано, по умолчанию генерируется ключ длиной 512 бит (минимально возможная длина).

 Файл `db_s_tcp.key` необходимо защитить средствами ОС Unix от просмотра и изменения другими пользователями, тем более нельзя перемещать его на другую машину.

-SSLAUTH

Режим идентификации клиента. В нем при выполнении очередного (не первого) соединения от клиента принимается используемый им ключ маскирования и сравнивается с запомненным ранее. Файл маскирования клиента, сохраняемый на диске, имеет имя, соответствующее IP-адресу клиента в точечной нотации и расширение `.crt`. В случае отсутствия в рабочем каталоге сетевого сервера файла маскирования клиента, он создается автоматически с сохранением в нем полученного ключа маскирования. В дальнейшем сравнение передаваемого при соединении ключа маскирования будет производиться с ключом, запомненным в первый раз.

Файл ключа маскирования клиента `db_s_tcp.crt` может быть перемещен на ЛИНТЕР-сервер администратором СУБД и соответствующим образом переименован. Т.к. клиентов может быть довольно много, целесообразно производить запуск сетевого сервера из отдельного каталога. Данный режим поддерживается только в случае наличия в текущем каталоге сетевого драйвера сервера, файлов ключа и сертификата.

 Этот механизм не обеспечивает 100% надежность идентификации клиента, т.к. в принципе ключ клиента может быть получен при подмене адреса ЛИНТЕР-сервера.

-SSLNOCREAT

Запрещает создавать файл ключа маскирования клиента при работе в режиме идентификации клиента. Это исключает добавление новых клиентских компьютеров без ведома администратора СУБД. Ключ может применяться только совместно с ключом `SSLAUTH`.

-T=<тайм-аут>

Задаёт длительность задержки (сек.) ответа от ядра СУБД ЛИНТЕР. По истечении половины заданного тайм-аута ядру СУБД посылается тестовый пакет. Если до окончания заданного тайм-аута от ядра не придёт ни одного сообщения, то произойдет разрыв всех сетевых соединений. Если задан только ключ `-T`, то по умолчанию величина тайм-аута принимается равной 60 сек. Величина тайм-аута не может быть меньше 10 сек.

-WD

Устанавливает режим контроля над работой драйвера. В этом режиме создается специальный («следающий») процесс, который контролирует основной процесс `dbstcp`. При зависании основного процесса производится его автоматический перезапуск. Обнаружение зависания происходит с запаздыванием в 10 сек. после того, как оно фактически произошло.

После окончания работы основного процесса `dbstcp` «следающий» процесс также завершает свою работу в течение 10 сек.

При завершении «следающего» процесса основной процесс тоже завершается.

Выполнение драйвера

Драйвер работает в автоматическом режиме.

Настройка защищенного соединения со стороны сервера

Чтобы установить защищенное соединение между сетевым клиентом и сетевым ЛИНТЕР-сервером необходимо в файле сетевой конфигурации `nodetab` вместо протокола обмена TCP/IP указать протокол TCP/IPS. Все остальные поля файла `nodetab` для протокола TCP/IPS имеют ту же функциональность, что и для протокола TCP/IP. Если используется протокол защищенного соединения, то сначала будет установлено обычное соединение по протоколу TCP/IP, а затем по специальному запросу сетевого драйвера клиента это соединение переведется в защищенный режим. По умолчанию сетевой драйвер сервера одновременно может работать как с обычными TCP/IP протоколами, так и с защищенными.

В файле `nodetab` можно задавать две строки с различными именами ЛИНТЕР-сервера и типами соединений, но с идентичными значениями сетевого адреса и порта. В этом случае возможно попеременное как защищенное, так и незащищенное соединение с одним и тем же сетевым ЛИНТЕР-сервером. На стороне сетевого сервера для обеспечения работоспособности данного режима нет необходимости предпринимать какие-либо дополнительные действия.

По умолчанию для защищенного режима используются ключи длиной 512 бит.

Ключ и сертификат по умолчанию автоматически генерируются при запуске сетевого драйвера ЛИНТЕР-сервера (во время генерации на экране отображаются точки) и сохраняются в оперативной памяти до завершения работы сетевого драйвера сервера. В случае наличия в текущем каталоге запуска сетевого драйвера сервера файлов `dbstcp.key` и `dbstcp.crt` (см. стр. 27, «Запуск драйвера») стадия генерации ключей пропускается, ключ и сертификат будут взяты из указанных файлов. Файлы ключа и сертификата должны соответствовать друг другу.

Сетевой ЛИНТЕР-сервер позволяет сгенерировать пару «ключ-сертификат». Для этого сетевой драйвер сервера необходимо запустить с ключом `-SSLKE` (одновременно можно задать длину ключа в битах). Продолжительность генерации ключа и сертификата лежит в пределах от нескольких секунд до нескольких минут (в зависимости от заданной длины ключа и мощности процессора). После окончания генерации ключ и сертификат будут сохранены в текущем каталоге в файлах `dbstcp.key` и `dbstcp.crt` соответственно. При последующем запуске из этого же каталога сетевого драйвера сервера будут использованы ключи и сертификат, запомненные в указанных файлах. Файлы

`dbb_tcp.key` и `dbb_tcp.crt` будут доступны для чтения и записи только их владельцу. Можно увеличить длину используемого ключа до 1024 или 2048 бит для повышения степени защиты.

Используя ключ `-SSLONLY`, можно отказаться от обмена с сетевым ЛИНТЕР-сервером по незащищенному каналу. В этом случае все попытки обращения к сетевому ЛИНТЕР-серверу без предварительного перевода соединения в защищенный режим приведут к разрыву соединения. При выборе ключа достаточной длины это обеспечивает практически полную защиту сетевого соединения от постороннего доступа, необходимо только тщательно контролировать доступ к файлам сертификата и (особенно) ключа.

Если файл сертификата будет:

- скопирован с серверного компьютера на клиентский в текущий каталог сетевого драйвера клиента;
- переименован таким образом, чтобы его имя (в верхнем регистре) совпало с именем ЛИНТЕР-сервера, определенным в файле `nodetab`;
- иметь расширение `.crt`

то сетевой драйвер клиента при установке защищенного соединения сравнит сохраненный сертификат и сертификат сервера и, в случае их несоответствия, закроет соединение. Это обеспечит невозможность соединения при подмене адреса сетевого ЛИНТЕР-сервера. Злоумышленник также должен будет получить файл ключа, хранящийся на сервере, поэтому этот файл должен быть защищен от доступа посторонних лиц средствами операционной системы. В данном режиме также невозможно будет установить соединение с сетевым ЛИНТЕР-сервером, который сгенерирует ключ и сертификат при запуске, или ключ которого не будет соответствовать сертификату, хранимому на стороне клиента. Клиент, не имеющий сертификата, сохраняет возможность открыть защищенное соединение с ЛИНТЕР-сервером.

Существует возможность проверки у клиента наличия соответствующего сертификата (для этого сетевой драйвер сервера необходимо запустить с ключами `SSLAUT` и `SSLONLY`).

Проверка выполняется следующим образом:

- если задан ключ `SSLAUT` и существуют файлы ключа и сертификата (`dbb_tcp.key` и `dbb_tcp.crt` соответственно) в текущем каталоге, то сетевой драйвер сервера (`dbb_tcp`) при установке с ним сетевого соединения запрашивает сертификат сетевого драйвера клиента;
- сетевой драйвер клиента читает свой ключ и сертификат из файлов `dbb_tcp.key` и `dbb_tcp.crt` текущего каталога драйвера сервера;
- сетевой драйвер клиента передает свой сертификат драйверу сервера по его запросу;
- драйвер сервера пытается прочитать сертификат данного клиента из файла, имеющего имя, соответствующее сетевому адресу данного клиента в точечной нотации (например, для адреса клиента 127.0.0.1 имя файла должно быть `127.0.0.1.crt`);
- если сертификат сетевого клиента отсутствует или не соответствует образцу, хранимому на сервере, защищенное сетевое соединение установлено не будет;
- в случае если соединение устанавливается в первый раз, и данный драйвер клиента имеет файлы ключа и сертификата, то соединение будет разрешено. При этом автоматически будет создан файл сертификата, соответствующий сетевому адресу данного клиента;

- если другой клиент попытается соединиться с того же сетевого адреса, или кто-то подменит сетевой адрес, то выявится несовпадение сертификатов, и такое соединение будет отвергнуто.

Если необходимо исключить подключение к ЛИНТЕР-серверу неизвестных клиентов, то следует добавить ключ SSLNOCREAT при запуске сетевого драйвера сервера. Это исключит возможность соединения с ЛИНТЕР-сервером тех клиентов, для которых у сетевого драйвера сервера нет файлов сертификата. Таким образом, администратор СУБД путем запрета создания файлов сертификатов может обеспечить защищенные соединения с ограниченным кругом клиентов. Файлы сертификатов служат «ключами» открытия соединения и должны тщательно оберегаться как на серверной, так и на клиентской стороне.

Драйвер клиента

Назначение драйвера

Драйвер клиента предназначен для обслуживания запросов клиентских приложений к локальному/удаленному ЛИНТЕР-серверу.

Условия выполнения

Для работы драйвера в ОС должен быть установлен сетевой протокол TCP/IP.

Запуск драйвера

Команда запуска драйвера:

```
dbc_tcp [<ключ> ...]  
<ключ> ::= <вариант 1> | <вариант 2>  
<вариант 1> ::= -<код ключа>{<пробел> | = }<значение ключа>  
<вариант 2> ::= /<код ключа>=<значение ключа>
```

Драйвер клиента должен запускаться на компьютере, где будет функционировать клиентское приложение. Драйвер нужно запускать обязательно до запуска приложения.

Ключи командной строки

```
-h | -HELP |?
```


Подсказка по всем ключам драйвера.

Пример

```
dbc_tcp -h  
[-H] [-HELP] [-?] - this help screen  
[-N Path]  
[-NODETAB Path] - path to \"nodetab\" file - configuration file of dbc_tcp  
[-S] [-DEFAULT] - set default server name  
[-M ExchangeId] - set interprocess communication identifier  
...
```

-S | -DEFAULT=<имя сервера>

Задает имя ЛИНТЕР-сервера по умолчанию. Если ключ не задан, то ЛИНТЕР-сервером по умолчанию считается первый из ЛИНТЕР-серверов, определенный в файле сетевой конфигурации `nodetab`.

 Ключ `DEFAULT` является синонимом ключа `S`.

Примеры

```
dbc_tcp -S Serv1
dbc_tcp -S=Serv2
dbc_tcp -default "Bank"
```

-N | -NODETAB=<спецификация файла nodetab>

Задает местоположение файла `nodetab` на клиентском компьютере.

Поиск драйвером файла `nodetab` выполняется по следующему алгоритму:

- если в командной строке запуска драйвера задан параметр `/N` (см. ниже), то файл берется из указанного каталога;
- в текущем каталоге драйвера клиента;
- в каталоге исполняемого модуля драйвера клиента;
- в домашнем каталоге пользователя;
- в каталогах, задаваемых переменной окружения `PATH`;

Пример

```
>dbc_tcp -n /path/to/file/nodetab
```

где `/path/to/file/nodetab` – путь к файлу `nodetab`.

-M=<параметр>

Задает идентификатор межпроцессного обмена (аналог переменной окружения `NET_MBX`).

-K | -PKILL=<сигнал>

Заставляет драйвер посылать указанный `<сигнал>` родительскому процессу по окончании своей инициализации. Значение `<сигнала>` должно быть целым положительным числом. В случае отсутствия аргумента или его неверного значения по умолчанию посылается сигнал `SIGTERM`.

-C | -PCHECK

Заставляет драйвер периодически проверять существование родительского процесса и прекращать свою работу после его завершения. Максимальный промежуток времени между завершением работы родительского процесса и завершением работы самого драйвера в этом случае составляет 10 сек.

-NODAEEMON


Запрещает переводить процесс драйвера в фоновый режим. Драйвер в этом случае не освобождает консоль до своего завершения.

-A | -AUTOCHANGE

Задает режим автоматического выбора другого ЛИНТЕР-сервера по умолчанию с уведомлением об этом клиентского приложения.

Если попытка соединения с явно указанным ЛИНТЕР-сервером по умолчанию (ключ `-S` или `-DEFAULT`) была неудачной, то клиентскому приложению возвращается соответствующий код завершения (ошибка установления соединения или истечение тайм-аута соединения), после чего драйвер автоматически выберет в качестве ЛИНТЕР-сервера по умолчанию следующий ЛИНТЕР-сервер из файла сетевой конфигурации `nodetab`. При последующем запросе приложения к ЛИНТЕР-серверу по умолчанию будет осуществлена попытка соединиться с вновь выбранным ЛИНТЕР-сервером. Смена ЛИНТЕР-сервера по умолчанию будет производиться до осуществления удачного сетевого соединения с одним из ЛИНТЕР-серверов из файла `nodetab`.

В случае если кандидатом на ЛИНТЕР-сервер по умолчанию был выбран последний ЛИНТЕР-сервер из файла `nodetab`, и соединение с ним также оказалось неудачным, то следующим будет назначен первый ЛИНТЕР-сервер файла `nodetab`.


 Если клиентское приложение при установке соединения с БД посылает несколько запросов (как, например, утилита `inl` – команды `OPEN` и `DESC`), то в случае невозможности соединения с ЛИНТЕР-сервером по умолчанию в режиме `AUTOCHANGE` эти запросы будут посланы *разным* ЛИНТЕР-серверам.

-G | -AUTOCONNECT

Задает режим автоматического выбора другого ЛИНТЕР-сервера по умолчанию без уведомления об этом клиентского приложения.

Если попытка соединения с явно указанным ЛИНТЕР-сервером по умолчанию (ключ `-S` или `-DEFAULT`) была неудачной в течение тайм-аута, то клиентскому приложению *не будет* возвращен соответствующий код завершения (ошибка установления соединения или истечение тайм-аута соединения). Драйвер автоматически выберет в качестве нового ЛИНТЕР-сервера по умолчанию следующий ЛИНТЕР-сервер из файла сетевой конфигурации `nodetab` и инициирует соединение с ним. Если соединение с очередным ЛИНТЕР-сервером окажется неудачным, процесс продолжится до исчерпания списка ЛИНТЕР-серверов в файле `nodetab`.

Клиентское приложение получит соответствующий код завершения только в случае невозможности установить соединение со *всеми* ЛИНТЕР-серверами, перечисленными в файле `nodetab`. Если значение переменной окружения `DBC_LOG` установлено в 1, то в стандартный поток ошибок будет выводиться информация о процессах установления и разрыва соединений, о смене соединения по умолчанию.

 Параметры `AUTOCONNECT` и `AUTOCHANGE` являются взаимно исключающими. При одновременном задании обоих параметров будет использован `AUTOCONNECT`.

-ERR1001

Заставляет драйвер в режиме `AUTOCONNECT` возвращать код завершения 1001 («ЛИНТЕР-сервер не найден») в случае, если все попытки соединения со всеми ЛИНТЕР-серверами из файла `nodetab` закончились неудачно.

В обычном режиме в этом случае возвращается ошибка соединения.

-AUTOCONNECTTIMEOUT=<тайм-аут циклического соединения>

Задаёт тайм-аут циклического соединения, т.е. интервал времени (сек.), в течение которого драйвер будет самостоятельно пытаться установить соединение с хотя бы одним ЛИНТЕР-сервером, перечисленным в файле `nodetab`.

Ключ `AUTOCONNECTTIMEOUT` имеет смысл только при совместном использовании с ключом `AUTOCONNECT`.

В режиме `AUTOCONNECT` (без ключа `AUTOCONNECTTIMEOUT`) клиентское приложение, использующее ЛИНТЕР-сервер по умолчанию, получит код завершения неудачного соединения с ЛИНТЕР-сервером *сразу* после каждой неудачной попытки установить соединение с ЛИНТЕР-серверами, перечисленными в файле `nodetab`.

В случае использования ключа `AUTOCONNECT` одновременно с ключом `AUTOCONNECTTIMEOUT` код завершения неудачного соединения с ЛИНТЕР-сервером будет возвращен клиентскому приложению только *после* указанного в ключе интервала времени, в течение которого драйвер будет производить периодические попытки соединиться с каким-либо ЛИНТЕР-сервером из перечисленных в файле `nodetab`.

Если в течение заданного интервала не будет установлено ни одно соединение, драйвер вернет клиентскому приложению код завершения 4006 (или 1001, если задан ключ `ERR1001`).

Если установить тайм-аут циклического соединения небольшим, то будет произведен только один цикл соединений, и задержка будет определяться суммарным временем попыток соединения с каждым из ЛИНТЕР-серверов (так же, как и при работе без ключа `AUTOCONNECTTIMEOUT`).

Если тайм-аут для конкретного соединения с каким-либо ЛИНТЕР-сервером (в файле `nodetab`) задать небольшим, то при разрыве сетевого соединения общее время от передачи клиентским приложением запроса сетевому серверу до возврата ошибки приложению будет определяться ключом `AUTOCONNECTTIMEOUT` или суммой времен тайм-аутов соединений всех ЛИНТЕР-серверов файла `nodetab`. Если же тайм-аут соединения в `nodetab` не задан, то скорость реакции на неудачное соединение будет определяться особенностями сетевого протокола ОС, но все равно будет суммой времен, потраченных на попытку соединения с каждым из ЛИНТЕР-серверов. Также необходимо учитывать, что тайм-аут не может быть менее двух секунд (реально минимальный тайм-аут устанавливается в интервале от 2 до 4 сек.). Это относится ко всем тайм-аутам `nodetab` и к ключу `AUTOCONNECTTIMEOUT`.

Таким образом, при задании минимальных тайм-аутов соединения (как в `nodetab`, так и в ключе `AUTOCONNECTTIMEOUT`) в 2 сек. и ключа `-G` реальный тайм-аут будет равен 2 - 4 сек., умноженным на количество ЛИНТЕР-серверов в файле `nodetab`. Задать тайм-аут 0 сек. нельзя, так как это значение зарезервировано для исключения тайм-аута клиента и сервера.

-CONNECTONLOAD

Заставляет драйвер начать немедленное установление соединения со всеми ЛИНТЕР-серверами сразу после запуска (даже если запросов на соединение от клиентских приложений не поступало) и держать открытыми установленные ранее соединения. Первый из ЛИНТЕР-серверов, с которым будет установлено соединение, назначается сервером по умолчанию.

В обычном режиме установление соединения с ЛИНТЕР-сервером (по умолчанию или указанным явно) инициируется только после получения от клиентского приложения запроса на соединение, и, соответственно, все ранее установленные соединения закрываются, если от клиентских приложений в течение примерно 5 мин. не поступило ни одного запроса к СУБД.

Если задан ключ **CONNECTONLOAD**, то сетевые соединения будут держаться открытыми до выгрузки драйвера.

-CONNERROR


Заставляет драйвер возвращать немедленно код завершения в случае отсутствия соединения с заданным ЛИНТЕР-сервером. В обычном режиме драйвер ожидает установления соединения и приема ответа от ЛИНТЕР-сервера или ошибки соединения, либо истечения тайм-аута соединения прежде, чем вернуть соответствующий код завершения клиентскому приложению. Если задан ключ **CONNERROR**, то код завершения будет возвращен клиентскому приложению немедленно. Это может быть полезно, например, в режиме автоматического соединения с ЛИНТЕР-сервером по умолчанию, когда приложение может ждать ответа достаточно долго. Если же запрос от клиентского приложения поступил после установления соединения, то работа с приложением будет проходить в обычном режиме.

-P | -PIDFILE=<спецификация файла>

Записывает идентификатор процесса (PID) драйвера клиента в текстовый файл.

Если заданный файл существует, он будет перезаписан.

Файл будет уничтожен при нормальном завершении работы драйвера.

 Информация о PID драйвера в текстовом файле требуется, как правило, при обработке командных файлов (например, для того, чтобы завершить работу драйвера).

-U

Завершение работы драйвера, функционирующего в фоновом режиме (драйверу передается сигнал **TERM**, инициирующий завершение работы).


- PING

Проверка активности удаленного ЛИНТЕР-сервера по умолчанию.

Драйвер клиента выводит на консоль сообщение об активности или неактивности ЛИНТЕР-сервера по умолчанию и завершает свою работу.

При активном ЛИНТЕР-сервере по умолчанию код возврата равен 0, в противном случае возвращается код завершения, который получило бы клиентское приложение в данной ситуации. Коды возврата могут быть использованы для проверки активности удаленного ЛИНТЕР-сервера в командном файле.

Для проверки активности произвольного ЛИНТЕР-сервера данный ключ должен применяться совместно с ключом **S**.

 Если в файле конфигурации **nodetab** для протокола TCP/IP тайм-аут соединения не задан, то он будет определяться операционной системой и может оказаться очень большим, что может повлиять на длительность работы драйвера, запущенного с ключом **PING**.

-VERSION

Вывод на консоль информации о версии драйвера клиента.

Выполнение драйвера

Драйвер работает в автоматическом режиме. В процессе работы драйвер использует управляющую информацию из указанного при запуске файла сетевой конфигурации **nodetab**.

Настройка защищенного соединения со стороны клиента

При указании в файле сетевой конфигурации `nodetab` протокола `TCP/IP` соединение будет осуществляться в защищенном режиме. Этот режим исключает получение информации, хранящейся в БД ЛИНТЕР, путем прослушивания сетевого соединения. Все остальные параметры соединения аналогичны соответствующим параметрам протокола `TCP/IP`.

Если в каталоге запуска драйвера присутствуют файлы `dbc_tcp.key` и `dbc_tcp.crt`, то происходит сравнение ключа маскирования сервера, считанного из файла и принятого по сети от сетевого драйвера сервера СУБД ЛИНТЕР. В случае несовпадения значений соединение не устанавливается. Если файл ключа маскирования сервера в текущем каталоге не найден, он будет создан автоматически. Файл ключа маскирования сервера имеет имя, совпадающее с именем ЛИНТЕР-сервера из файла `nodetab`, и расширение `.crt`. Имя ЛИНТЕР-сервера в файле приводится к верхнему регистру. Администратор СУБД может перенести файл `dbc_tcp.crt` на клиентскую сторону и соответствующим образом переименовать его до установления первого соединения с сервером. Наличие файлов `dbc_tcp.crt` и `dbc_tcp.key` (ключей маскирования клиента) позволяет сетевому драйверу сервера получить ключ маскирования клиента и произвести сравнение с запомненным ранее ключом.

Создать файлы ключей маскирования клиента можно, запустив `dbc_tcp` с ключом `SSLKEY` и переименовав полученные файлы `dbc_tcp.key` и `dbc_tcp.crt` в `dbc_tcp.key` и `dbc_tcp.crt` соответственно. При этом файлы должны быть защищены средствами ОС.

При инициализации защищенного соединения сначала устанавливается обычное соединение по протоколу `TCP/IP`, затем оно переводится в защищенный режим. Существуют три режима защищенного соединения:

- 1) без аутентификации сервера;
- 2) аутентификация с сохранением ключей в автоматическом режиме;
- 3) аутентификация с сохранением ключей в ручном режиме.

Если драйвер клиента запускается стандартным образом, без манипулирования файлами `dbc_tcp.key` и `dbc_tcp.crt`, аутентификация сервера не проводится, но защищенное соединение будет установлено.

Если перенести сертификаты с серверного компьютера на клиентский и сохранить их в каталоге запуска драйвера клиента с именем, аналогичным имени ЛИНТЕР-сервера из файла `nodetab`, и с расширением `.crt`, то при установке соединения будет проверено соответствие запомненного сертификата клиента и сертификата сервера. В случае их несоответствия соединение установлено не будет.

Если в каталоге запуска драйвера клиента создать файлы ключа и сертификата клиента с именами `dbc_tcp.key` и `dbc_tcp.crt` соответственно, это позволит:

- передать сертификат клиента серверу для аутентификации клиента сервером;
- включить режим автоматического сохранения сертификата сервера в текущем каталоге при установке соединения с данным ЛИНТЕР-сервером в первый раз.

При установке последующих соединений с данным ЛИНТЕР-сервером будет производиться сверка сертификатов с запомненным при первом соединении.

Файлы ключей и сертификатов должны быть защищены от доступа посторонних лиц. Это достигается сменой прав доступа на 0600 средствами ОС Unix. Сертификат и ключ могут быть созданы утилитой `openssl` (в состав дистрибутива СУБД ЛИНТЕР не входит) с именами `dbc_tcp.crt` и `dbc_tcp.key` соответственно. Также сертификат и ключ можно создать утилитой `dbstcp` с опцией `SSLKEY`. В этом случае файлы должны быть переименованы из `dbstcp.crt` в `dbc_tcp.crt` и `dbstcp.key` в `dbc_tcp.key`.

Сетевые средства для Win32-платформ

Драйвер сервера

Запуск драйвера

Исполняемый файл драйвера для семейства ОС MS Windows NT/2000/XP – `dbw_wnt.exe`, для семейства ОС MS Windows 95/98/Millennium – `dbw_w95.exe`.

Команда запуска драйвера:

```
dbw_wnt [ {/| -} <ключ>...]  
dbw_w95 [ {/| -} <ключ>...]
```

Ключи командной строки

`/LOCAL`

Задаёт режим выполнения драйвера как приложения с графическим интерфейсом. Если ключ не задан, по умолчанию драйвер предпримет попытку запуститься с ключом `/LOCAL` через непродолжительную паузу.

Пример

```
dbw_wnt /local
```

```
/L=[<уровень трассировки>  
[:[<размер файла>[<число файлов>]]<спецификация файла>]
```

Управляет выводом трассировочной информации. Трассируется установка и потеря соединения. В случае потери соединения трассируется также причина разрыва.

```
<уровень трассировки> ::=  
<шестнадцатичное значение> | <предопределенное значение>
```

Каждый двоичный разряд этого числа включает один уровень трассировки, соответственно, максимально возможный уровень трассировки `0xffffffff`.

```
<предопределенное значение> ::=  
ERROR | SYSERROR | INIT | INFO | CLIENT | LINTER  
| SYSINFO | PACKET | DBG | TRACE
```

Несколько предопределенных уровней задаются через запятую.



По умолчанию выполняется трассировка только ошибок.

```
<размер файла> ::= целочисленное положительное значение
```

Задаёт ограничение в байтах на размер файла трассировки.

При превышении указанного размера создается новый файл трассировки. Старый файл переименовывается с добавлением числового постфикса после названия.

По умолчанию ограничение на размер файла равно 1 Мбайт.

<число файлов> ::= целочисленное положительное значение

Задаёт количество файлов трассировки. Отделяется от параметра <размер файла> символом «х» (латиница).

Если задано несколько файлов трассировки, то при превышении текущим файлом трассировки заданного размера происходит переименование всех файлов трассировки с увеличением числового постфикса, добавляемого к названию файла, на 1.

Если количество файлов трассировки превысило <число файлов>, то происходит удаление самого старого файла трассировки. При этом место, занимаемое файлами трассировки на файловой системе, ограничивается.

<спецификация файла> ::= имя и местоположение файла трассировки

Если не спецификация файла не задана, по умолчанию создается файл `dbb_tcp.log` в текущем каталоге.

Пример

```
-L 0xFFFFFFFF:1000000x5/var/log/dbb_tcp.log
```

Ключ устанавливает следующие параметры трассировки:

- уровень трассировки – максимальный;
- размер файлов ограничен 10 Мбайт;
- количество файлов – 5;
- местоположение и имя файла трассировки `/var/log/dbb_tcp.log`.

В результате на файловой системе в каталоге `/var/log` будут созданы файлы `dbb_tcp.log`, `dbb_tcp.log1`, `dbb_tcp.log2`, `dbb_tcp.log3`, `dbb_tcp.log4` `dbb_tcp.log5`.

```
/DEBUG [<уровень трассировки>]  
[:[<размер файла>[х<число файлов>]]<спецификация файла>]
```

Управляет выводом трассировочной информации.

Параметры и результат работы ключа `-DEBUG` аналогичны параметрам и результату работы ключа `-L` с учетом следующих дополнений:

- указывается место в файле, где произошел вызов;
- выводятся системные ошибки, возникающие при нормальной работе.

Пример

```
dbb_wnt /local /debug
```

/NAME=<параметр>

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

/VERSION=<номер>

Задаёт номер версии СУБД ЛИНТЕР, с которой должен работать драйвер. Значение <номера>:

- 4 – для версии 4 и младше;
- 5 – для версии 5 и старше.

Если параметр VERSION или аргумент <номер> не заданы, используется значение по умолчанию, равное версии дистрибутива СУБД ЛИНТЕР, в состав которого входил драйвер.

/TCP=<адрес>

Задаёт сетевой адрес порта при работе по протоколу TCP/IP.

/SPX=<адрес>

Задаёт сетевой адрес порта при работе по протоколу IPX/SPX..

/NBNAME=<адрес>

Задаёт сетевой адрес при работе по протоколу NETBIOS.

/NB=<номер>

Задаёт номер сетевого порта NETBIOS.

/NP=<имя>

Задаёт имя именованного канала (Named Pipe).

Функции драйвера

Настройка драйвера

Пользовательский интерфейс

Настройка пользовательского интерфейса выполняется аналогично настройке в сетевом драйвере клиента (см. раздел «Настройка драйвера»).

Статистическая информация

Процедуры управления сбором статистической информации и её просмотр выполняются так же, как и в сетевом драйвере клиента (см. раздел «Статистическая информация»).

Драйвер клиента

Условия выполнения

Для работы драйвера должен быть установлен соответствующий протокол обмена.

Запуск драйвера

Исполняемый файл драйвера для семейства ОС MS Windows NT/2000/XP – `dbc_wnt.exe`, для семейства ОС MS Windows 95/98/Millennium – `dbc_w95.exe`.

Команда запуска драйвера:

```
dbc_wnt [ {/| -} <ключ> ...]  
dbc_w95 [ {/| -} <ключ> ...]
```

Ключи командной строки

`/LOCAL`

Задаёт режим выполнения драйвера как приложения с графическим интерфейсом. Если ключ не задан, по умолчанию драйвер предпримет попытку запуститься с ключом `/LOCAL` через непродолжительную паузу.

Пример

```
dbc_wnt /local
```

`/DEBUG`

Задаёт вывод на консоль отладочной информации.

Пример

```
dbc_wnt /local /debug
```

`/CRYPT`

Задаёт защищённый режим приёма/передачи данных по каналам сети.

Пример

```
dbc_wnt /crypt
```

`/REPLY`

Разрешает разрыв соединения по тайм-ауту клиента.

Пример

```
dbc_wnt /crypt /reply
```

`/S|-DEFAULT=<имя сервера>`

Задаёт имя ЛИНТЕР-сервера по умолчанию (см. описание ключа `S` в драйвере `db_s_tcp`, стр. 33).

`/N=<спецификация файла nodetab>`

Задаёт местоположение файла `nodetab` на клиентском компьютере (см. описание ключа `N` в драйвере `db_s_tcp`, стр. 33).

Поиск драйвером файла `nodetab` выполняется по следующему алгоритму:

- если в командной строке запуска драйвера задан ключ `/N`, то файл берётся из указанного каталога;
- в каталоге, который использовался при предыдущем сеансе работы драйвера;

- в подкаталоге /bin установочного каталога СУБД ЛИНТЕР;
- в каталоге запуска драйвера клиента.

/MBX=<параметр>

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

Если значение <параметра> не равно значению по умолчанию, межпроцессный обмен доступен только тем клиентским приложениям, для которых переменная окружения NET_MBX задана, и ее значение совпадает с MBX драйвера клиента.

На одном компьютере возможен запуск нескольких драйверов клиентов (например, работающих с разными файлами nodetab), которые функционируют с различными значениями MBX.

При попытке запустить второй драйвер клиента с одинаковым значением NET_MBX будет зафиксирована ошибка создания почтового ящика.



Символьное значение переменной окружения NET_MBX – регистронезависимое.

/VERSION=<номер>

Задает номер версии СУБД ЛИНТЕР, с которой должен работать драйвер. Значение <номера>:

- 4 – для версии 4 и младше;
- 5 – для версии 5 и старше.

Если ключ VERSION не задан, используется значение по умолчанию, равное версии дистрибутива СУБД ЛИНТЕР, в состав которого входил драйвер.

Функции драйвера

Выбор файла сетевой конфигурации

В среде ОС Win32 информация о местоположении файла сетевой конфигурации nodetab записывается драйвером при первоначальном запуске или при изменении nodetab (как содержимого, так и местоположения) в реестр операционной системы.

При первоначальном запуске драйвера местоположение файла nodetab указывается при помощи ключа \N или, если он не задан, определяется по умолчанию (см. стр. 33). Определить, какой именно файл nodetab будет использовать драйвер после запуска, можно следующим образом:

1. Запустить драйвер (драйвер не должен быть запущен как сервис ОС).
2. Активизировать главное окно драйвера (см. Рис. 5).

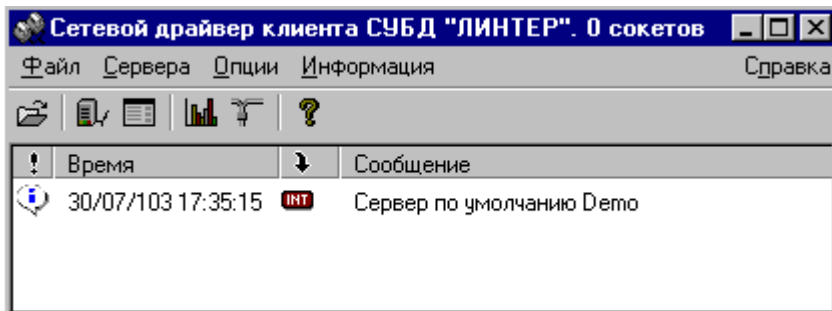



Рис. 5. Сетевой драйвер клиента

3. Выполнить команду **Файл ⇒ Список серверов...** или нажать клавиши <Ctrl>+<O>, или кнопку **Nodetab**  на панели инструментов. Появится стандартное окно поиска и выбора файла, которое содержит путь к файлу **nodetab**.


Настройку драйвера для работы с другим файлом **nodetab** или с тем же самым, но с измененными данными, можно выполнить двумя методами: статическим и динамическим.

При статическом методе работа драйвера должна быть завершена, после чего он снова запускается на выполнение с указанием нового местоположения файла **nodetab** (ключ /N), если местоположение действительно изменилось, либо без ключа /N, если изменилось только содержимое файла **nodetab**, а не его местоположение.

Динамический метод настройки драйвера для работы с другим файлом **nodetab** выполняется без завершения работы драйвера, т.е. параллельно с обслуживанием клиентских запросов к ЛИНТЕР-серверам. При этом необходимо учитывать следующие особенности функционирования драйвера:

- при запуске драйвер считывает информацию о параметрах соединения из файла **nodetab** в свои внутренние структуры данных и при дальнейшей работе уже не обращается к файлу **nodetab**. Фактически это означает, что если после запуска драйвера сетевые параметры в файле **nodetab** были изменены извне, то для драйвера данный факт останется незамеченным. Он продолжит работать с теми параметрами, которые запомнил и сохранил в своих внутренних структурах. Следовательно, в обычном режиме факт изменения **nodetab** драйвер и клиентское приложение, соответственно, обнаружат только в том случае, если повторно иницируют сетевое соединение с этим же ЛИНТЕР-сервером;
- при редактировании файла **nodetab** средствами данного драйвера или утилитой «Сетевой администратор СУБД ЛИНТЕР» осуществляется реальное обновление «внутреннего» списка сетевых серверов драйвера клиента. Следующие попытки установить соединение будут производиться с учетом сделанных изменений. Если произведено изменение параметров ЛИНТЕР-сервера, с которым в данный момент имеется активное соединение, то эти изменения вступят в силу только при перезапуске соответствующего сетевого драйвера сервера;
- можно свободно удалять или модифицировать параметры сетевого соединения тех ЛИНТЕР-серверов, с которыми в данный момент нет активного сетевого соединения в локальной сети. Для просмотра всех активных соединений сети можно воспользоваться утилитой «Сетевой администратор СУБД ЛИНТЕР»;
- можно свободно добавлять новые ЛИНТЕР-серверы.


Для динамического изменения настройки файла `nodetab` драйвера клиента необходимо:

1. Убедиться, что новая версия файла `nodetab` не конфликтует с текущим состоянием локальной сети (т.е. информация о всех текущих сетевых соединениях в новом файле `nodetab` осталась неизменной). В противном случае закрыть, по возможности, все клиентские приложения, работающие с данным ЛИНТЕР-сервером, и перезапустить их после обновления файла `nodetab`.
2. Выполнить команду **Файл** ⇒ **Список серверов...** или нажать клавиши `<Ctrl>+<O>`, или кнопку **Nodetab**  на панели инструментов. Появится стандартное окно поиска и выбора файла, которое содержит путь к файлу `nodetab`.
3. Выбрать новое местоположение файла `nodetab` или подтвердить существующее. Информация о новом местоположении `nodetab` будет сохранена в реестре ОС Windows.

Сетевое конфигурирование

Добавление ЛИНТЕР-сервера

Для добавления нового ЛИНТЕР-сервера в файл сетевой конфигурации `nodetab` следует:

1. Активизировать главное окно драйвера (см. Рис. 6).
2. Выполнить команду **Сервера** ⇒ **Список...** или нажать клавиши `<Ctrl>+<L>` либо кнопку **Список**  на панели инструментов. Появится диалоговое окно для работы с файлом `nodetab` (рис. 6).

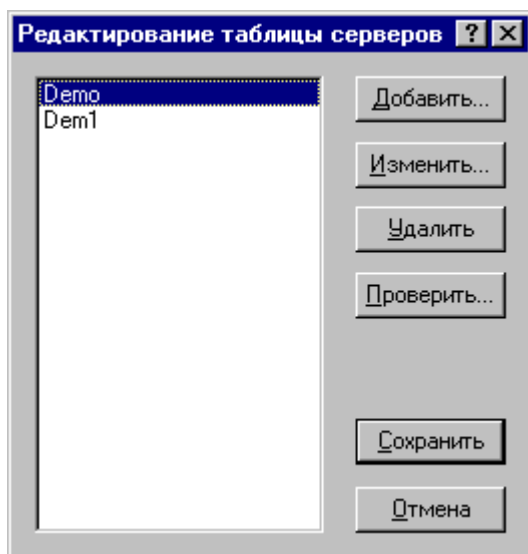


Рис. 6. Список серверов

3. Нажать кнопку **Добавить**. Появится диалоговое окно для установки параметров сетевого соединения (рис. 7).

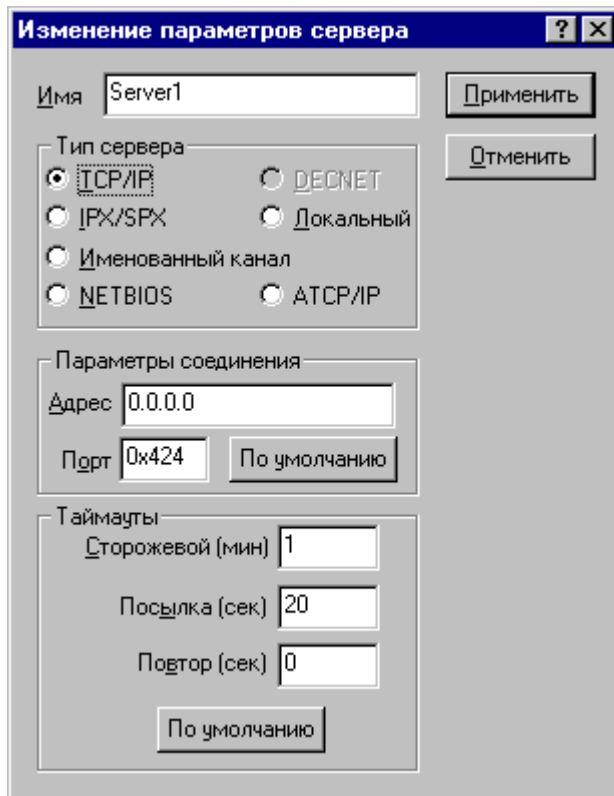


Рис. 7. Изменение параметров сервера

4. Заполнить поля диалогового окна.

Имя

Имя ЛИНТЕР-сервера (длиной до 8 символов). Введенное имя на дубликат значения не проверяется, поэтому файл `nodetab` может содержать любое количество идентичных имен ЛИНТЕР-серверов, однако драйвером всегда будет использоваться первый (в смысле номера записи в файле) ЛИНТЕР-сервер из этого списка.

Тип протокола

В группе переключателя Тип протокола выбрать протокол, по которому должен выполняться обмен данными драйвера клиента с ЛИНТЕР-сервером.

Параметры соединения

В группе Параметры соединения указать в поле Адрес сетевой адрес (сетевое имя) и в поле Порт номер порта сетевого соединения. Поля Адрес и Порт могут быть недоступны, если для выбранного типа протокола некоторые из них (или оба одновременно) не имеют смысла. При выборе протокола соединения в полях Адрес и Порт предлагаются для ввода принятые значения по умолчанию.

Если принятые по умолчанию значения были заменены другими, для их восстановления нужно нажать кнопку По умолчанию.

Тайм-ауты


В группе Тайм-ауты указать временные параметры соединения (см. раздел «Файл сетевой конфигурации»). Все или часть полей ввода этой группы могут быть недоступны, если для выбранного типа протокола некоторые из них (или все одновременно) не имеют смысла. Для выбранного протокола соединения в этих полях предлагаются принятые значения по умолчанию.

Если принятые по умолчанию значения были заменены другими, для их восстановления нужно нажать кнопку По умолчанию.

5. Для подтверждения (но еще не сохранения в файле `nodetab`!) введенных данных нажать кнопку **Применить**, для отказа от ввода – кнопку **Отменить**.
6. Повторить, если необходимо, шаги 3-5 для добавления следующего ЛИНТЕР-сервера.
7. Для включения информации о новых ЛИНТЕР-серверах в файл `nodetab` нажать кнопку **Сохранить** (рис. 6).


Изменение параметров соединения с ЛИНТЕР-сервером

Для изменения параметров существующего ЛИНТЕР-сервера необходимо:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выполнить команду **Сервера ⇒ Список...** или нажать клавиши `<Ctrl>+<L>` либо кнопку **Список**  на панели инструментов. Появится диалоговое окно для работы с файлом `nodetab` (рис. 6).
3. Выделить (щелкнуть левой кнопкой мыши) в предложенном списке имя ЛИНТЕР-сервера, параметры соединения которого должны быть изменены.
4. Нажать кнопку **Изменить**. Появится диалоговое окно, аналогичное окну при добавлении ЛИНТЕР-сервера (см. рис. 7). Далее процедура изменения параметров совпадает с процедурой добавления нового ЛИНТЕР-сервера (см. шаги 4-7 предыдущего пункта).

Удаление ЛИНТЕР-сервера


Для удаления информации о ЛИНТЕР-сервере из файла `nodetab` следует:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выполнить команду **Сервера ⇒ Список...** или нажать клавиши `<Ctrl>+<L>` либо кнопку **Список**  на панели инструментов. Появится диалоговое окно для работы с файлом `nodetab` (см. рис. 6).
3. Выделить (щелкнуть левой кнопкой мыши) в предложенном списке имя ЛИНТЕР-сервера, который должен быть удален из файла `nodetab`.
4. Нажать кнопку **Удалить**.
5. Для подтверждения удаления нажать кнопку **Сохранить**, для отказа от удаления – кнопку **Отменить**.

Проверка сетевого доступа к ЛИНТЕР-серверу

После того как новый ЛИНТЕР-сервер был добавлен в файл `nodetab`, или были изменены параметры соединения существующего ЛИНТЕР-сервера, рекомендуется проверить корректность этих параметров.

Для этого необходимо:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выполнить команду **Сервера ⇒ Список...** или нажать клавиши `<Ctrl>+<L>` либо кнопку **Список**  на панели инструментов. Появится диалоговое окно для работы с файлом `nodetab` (см. рис. 6).
3. Выделить (щелкнуть левой кнопкой мыши) в предложенном списке имя ЛИНТЕР-сервера, сетевое соединение с которым должно быть проверено.
4. Нажать кнопку **Проверить**.

5. Результаты проверки (положительные или отрицательные) будут выданы в информационном окне, как, например, на Рис. 8, при этом во фразе Статус выдается код завершения СУБД ЛИНТЕР (первое числовое значение), и код завершения операционной системы (второе значение).

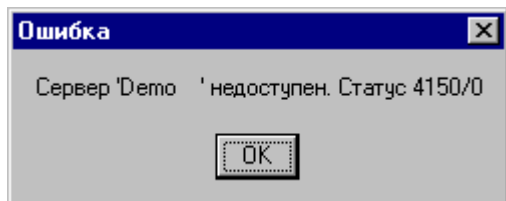



Рис. 8. Результат проверки

Назначение ЛИНТЕР-сервера по умолчанию

ЛИНТЕР-сервер по умолчанию – это такой сервер, с которым можно устанавливать соединение без указания его имени в клиентском приложении. На каждом клиентском компьютере может быть только один ЛИНТЕР-сервер по умолчанию (информация о нем хранится в реестре ОС windows). Имя ЛИНТЕР-сервера по умолчанию задается при помощи ключа /S или /DEFAULT в командной строке запуска драйвера или в графическом режиме – после запуска драйвера.

Для определения текущего ЛИНТЕР-сервера по умолчанию или установки нового необходимо:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выполнить команду **Сервера** ⇒ **По умолчанию...** или нажать клавиши <Ctrl>+<D> либо кнопку **Сервер БД**  на панели инструментов. Появится диалоговое окно для выбора ЛИНТЕР-сервера по умолчанию (рис. 9).

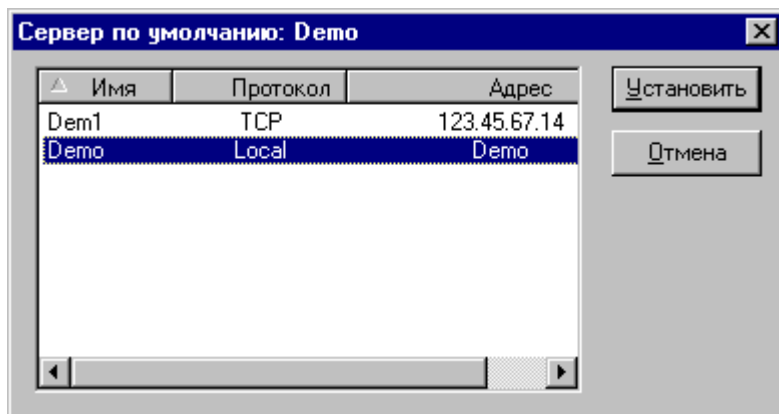


Рис. 9. Сервер по умолчанию

3. Выделить (щёлкнуть левой кнопкой мыши) в предложенном списке имя ЛИНТЕР-сервера, который должен быть объявлен ЛИНТЕР-сервером по умолчанию.
4. Нажать кнопку Установить.


Настройка драйвера

Разрешённые протоколы

Администратор СУБД ЛИНТЕР может устанавливать запрет/разрешение на постоянное или временное использование драйвером клиента некоторых протоколов обмена, поддерживаемых операционной системой. Если какой-либо протокол запрещен для

использования, его нельзя будет выбрать в диалоговом окне конфигурирования файла nodetab (см. рис. 7).

Для установки списка протоколов, доступных для сетевого конфигурирования драйвера клиента следует:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выполнить команду **Информация** ⇒ **Протоколы** или нажать клавиши <Ctrl>+<P> либо кнопку  на панели инструментов. Появится диалоговое окно настройки сетевых протоколов (рис. 10).

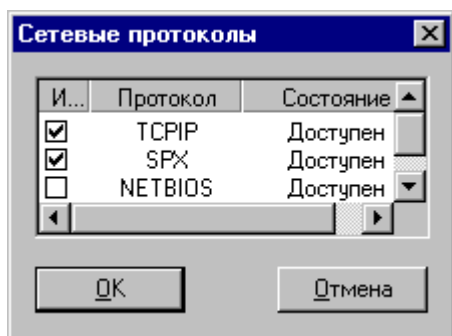


Рис. 10. Доступные протоколы обмена

В приведенном на Рис. 10 окне столбец Протокол показывает имена поддерживаемых СУБД ЛИНТЕР протоколов обмена, столбец Состояние – их доступность для данной версии драйвера клиента, а столбец Использовать – разрешение/запрет на использование этого протокола на данном клиентском компьютере.

3. Для разрешения использовать протокол установить флажок в столбце Использовать, для запрета – флажок сбросить.
4. Нажать кнопку ОК.

Настройка интерфейса

Для настройки пользовательского интерфейса необходимо:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выбрать в меню команду **Опции**. Появится всплывающее окно для индивидуальной настройки графического интерфейса драйвера (рис. 11).

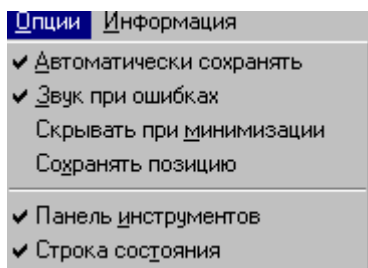


Рис. 11. Опции настройки интерфейса

3. Установить или отменить настройки интерфейса (щелкнуть левой кнопкой на выбранной опции). В зависимости от текущего состояния опция будет инвертирована – установлена или отменена.

Опции настройки

Автоматически сохранять

Сохранять/не сохранять размеры, местоположение на экране, размеры столбцов главного окна драйвера и опции настройки интерфейса.

Звук при ошибках

Подавать/не подавать звуковой сигнал при вводе ошибочных данных.

Скрывать при минимизации

Показывать/не показывать в статусной строке главного окна операционной системы MS Windows кнопку драйвера при минимизации главного окна драйвера.

Сохранять позицию

Сохранять/не сохранять размеры и местоположение на экране главного окна драйвера.

Панель инструментов

Показывать/скрывать панель инструментов главного окна драйвера.


Строка состояния

Показывать/скрывать строку состояния главного окна драйвера.

Статистическая информация

Просмотр статистической информации

Для просмотра статистической информации о работе драйвера (количество текущих активных соединений с ЛИНТЕР-серверами, объем переданных/принятых данных, ошибки соединения и др.) необходимо:

1. Активизировать главное окно драйвера (см. Рис. 5).
2. Выполнить команду **Информация** ⇒ **Статистика** или нажать клавиши <Ctrl>+<S> либо кнопку **Статистика**  на панели инструментов. Появится диалоговое окно статистической информации драйвера (рис. 12). Заголовок окна указывает на дату начала сбора статистики (дату последнего запуска драйвера).

Окно содержит обобщенную и подробную статистику.

Обобщённая статистическая информация

Запросов (прд/прн)

Общее количество обработанных запросов к ЛИНТЕР-серверам.

Байт (прд/прн)

Суммарное количество переданных/принятых от ЛИНТЕР-серверов байтов данных.

Ошибок (прд/прн)

Суммарное количество ошибок передачи/приема данных.

Подробная статистическая информация

Имя

Имя ЛИНТЕР-сервера.

Адрес

Сетевой адрес ЛИНТЕР-сервера.

Протокол

Имя протокола обмена.

Байт (прд/прн)

Количество переданных/принятых байтов данных.

Ошибок

Количество ошибок приема/передачи данных.

CPS (прд/прн)

Текущая скорость обмена данными.

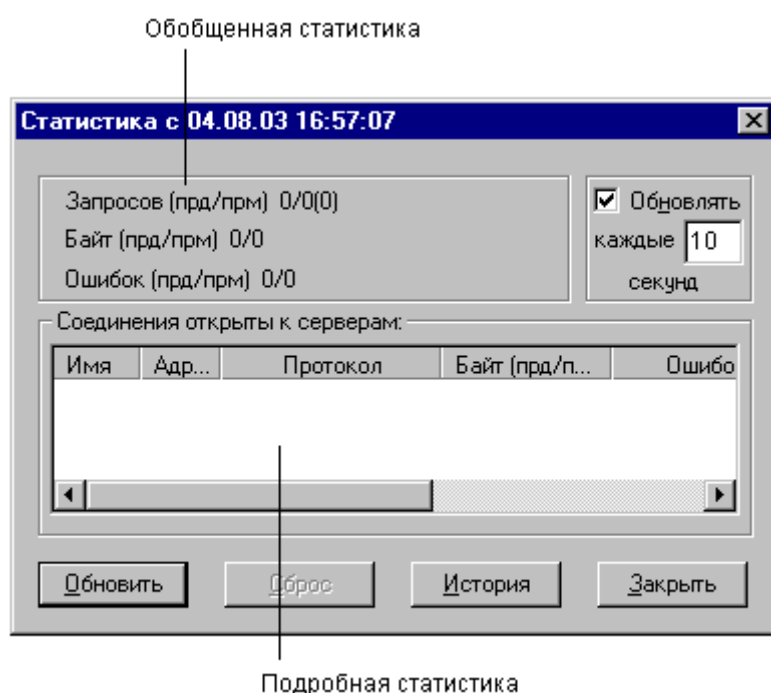


Рис. 12. Статистика работы драйвера

Обновление статистической информации

Для принудительного обновления статистической информации следует нажать кнопку Обновить (рис. 12).

Для автоматического обновления:

1. Задать в поле Обновлять периодичность (в сек.) обновления статистической информации.
2. Установить флажок автоматического обновления.

История соединений

Для просмотра истории соединений драйвера за текущий сеанс работы необходимо:

1. Активизировать окно отображения статистической информации (см. рис. 12).
2. Нажать кнопку История. Появится диалоговое окно (Рис. 13), отображающее в хронологическом порядке все результаты соединения драйвера со всеми ЛИНТЕР-серверами.

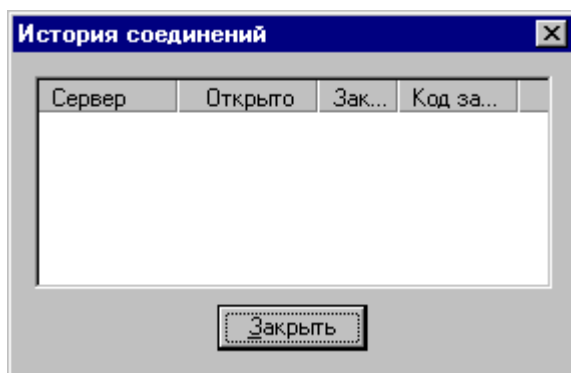


Рис. 13. История соединений

Статистика истории соединений

Сервер

Имя ЛИНТЕР-сервера.

Открыто

Дата и время начала соединения.

Закрето

Дата и время окончания соединения.

Код завершения

Код завершения СУБД результата соединения.

Тестирование конфигурации сети

После создания или модификации файла сетевой конфигурации рекомендуется проверить корректность введенных сетевых параметров.

Тестирование на Unix-платформе

Для тестирования файла `nodetab` на платформе Unix следует:

1. Активизировать любой ЛИНТЕР-сервер, указанный в файле `nodetab` (т.е. запустить на нем драйвер сервера и ядро СУБД ЛИНТЕР).
2. Запустить на клиентском компьютере драйвер клиента:

```
dbc_tcp /S <имя сервера> /ping
```

где `<имя сервера>` – имя тестируемого ЛИНТЕР-сервера.

3. Повторить операции 1, 2 для всех остальных ЛИНТЕР-серверов файла `nodetab`.

Тестирование на Win32-платформе

Для тестирования файла `nodetab` на платформе Win32 следует:

1. Активизировать любой ЛИНТЕР-сервер, указанный в файле `nodetab` (т.е. запустить на нем драйвер сервера и ядро СУБД ЛИНТЕР).
2. Запустить на клиентском компьютере драйвер клиента и выполнить проверку сетевого доступа (см. пункт «Проверка сетевого доступа к ЛИНТЕР-серверу» в данном документе).
3. Повторить операцию на всех клиентских компьютерах.

Коды завершения сетевых средств

Все коды завершения, генерируемые сетевыми драйверами сервера и клиента, передаются клиентскому приложению не напрямую, а косвенно, от имени СУБД ЛИНТЕР, как результат обработки клиентского SQL-запроса (см. документ «СУБД ЛИНТЕР. Справочник кодов завершения»).

Управление сетевым доступом

В этом разделе описана процедура управления сетевым доступом на примере типовой сетевой конфигурации СУБД ЛИНТЕР (рис. 14). Описание дается применительно к Unix-платформам.

Для Win32-платформ управление сетевым доступом выполняется аналогично с соответствующими поправками (другие имена ключей, идентификаторы межпроцессного обмена и т.п.).

Для обеспечения сетевого доступа к БД администратор СУБД должен выполнить следующие действия:

1. Определить каталоги БД на данном ЛИНТЕР-сервере, для которых должен быть разрешен удаленный (сетевой) доступ.
2. Для каждой такой БД, кроме первой, выполнить процедуру настройки переменных окружения, используемых СУБД ЛИНТЕР в процессе функционирования, и запустить экземпляр ядра СУБД. Для первой БД достаточно указать только переменную окружения SY00. Ядро СУБД использует переменную SY00 для определения местонахождения файлов БД и переменную LINTER_MBX для взаимодействия с сетевым драйвером сервера:
 - указываем местоположение БД: например SY00=/linter/db/bank;
export SY00;
 - задаем номер очереди сообщений: например, LINTER_MBX=30000;
 - запускаем на выполнение экземпляр СУБД ЛИНТЕР.
3. Для обеспечения взаимодействия запущенного экземпляра ядра СУБД ЛИНТЕР с удаленными клиентами запустить сетевой драйвер сервера с ключами /М, /Р.

Ключ /М определяет номер очереди, через которую драйвер будет взаимодействовать с ядром, поэтому его значение должно совпадать со значением переменной LINTER_MBX, под которым было запущено ядро.

Ключ /Р задает номер порта, через который должно осуществляться взаимодействие сетевого драйвера сервера и клиента (DBS_TCP, DBC_TCP) между собой.

По файлу nodetab определяем, что выбранной БД SY00=/linter/db/bank соответствует номер порта 1060, следовательно, /Р=1060. Или, наоборот, задаем номер порта и затем редактируем nodetab.

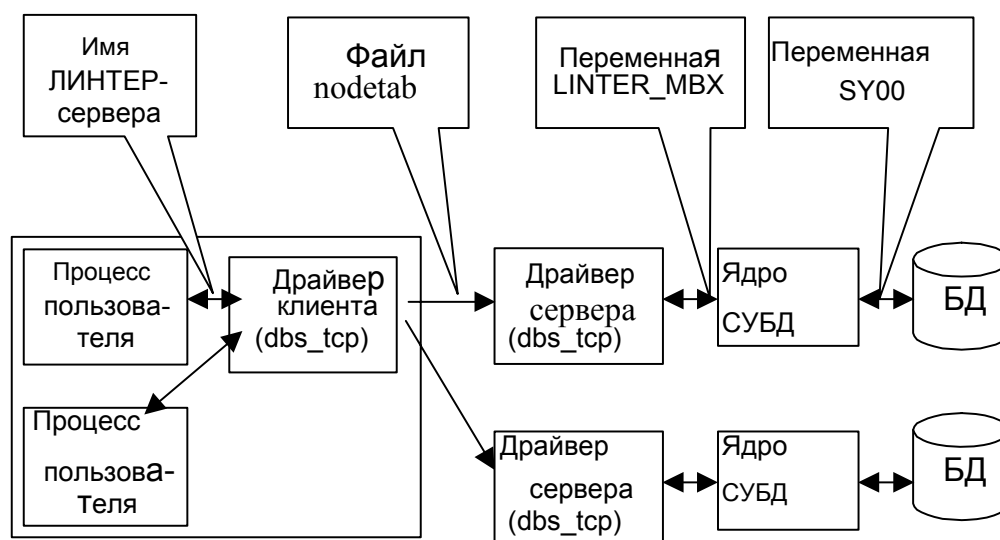


Рис. 14. Схема доступа к удаленной БД

4. Запустить сетевой драйвер сервера с установленными параметрами:
`DVC_TSP /M=30000 /P=1060` Полученная схема взаимодействия сетевого и клиентского драйверов показана на рис. 15.
5. Повторить первые четыре пункта с необходимыми модификациями для запуска всех БД на выбранном узле.
6. Повторить первые пять пунктов для всех серверов ЛИНТЕР.



Рис. 15. Схема взаимодействия сетевого и клиентского драйверов

Если на сервере установлена только одна БД, то при запуске сетевого драйвера сервера ключи `/P`, `/M` можно не задавать (драйвер и ядро СУБД будут использовать значение по умолчанию). При этом номер порта в файле `nodetab` для соответствующего сервера также должен иметь значение, равное значению порта по умолчанию для этого типа протокола.

Активизация клиента


Удаленный клиент

Для удаленного доступа к конкретной БД необходимо:

1. Запустить на клиентском компьютере сетевой драйвер клиента. Он может обслуживать несколько одновременно функционирующих на компьютере приложений, поэтому должен запускаться только один раз.

В процессе соединения драйвер использует управляющую информацию из файла `nodetab`. По умолчанию драйвер предполагает, что `nodetab` находится в том же каталоге, что и сам драйвер.

Если это не так, или на компьютере имеется несколько версий файла `nodetab`, то следует использовать ключ `/N`, задающий конкретный полный путь для поиска файла `nodetab`.

 Файл `nodetab` на пользовательском компьютере не обязан содержать список всех ЛИНТЕР-серверов локальной сети. Так, если с данного компьютера всегда осуществляется соединение лишь с одной конкретной БД, файл `nodetab` может состоять только из одной строки, таким образом, файлы `nodetab` на разных компьютерах сети могут различаться. Это предоставляет администратору СУБД дополнительные возможности для ограничения доступа пользователей к БД путем расширения или сужения списка доступных ЛИНТЕР-серверов в каждом локальном файле `nodetab`.

2. Запустить клиентское приложение и в процессе соединения указать имя соответствующего ЛИНТЕР-сервера. Дальнейший процесс протекает в следующей последовательности:
 - сетевой драйвер клиента получает имя ЛИНТЕР-сервера, с которым приложение предполагает установить соединение;

- в файле `nodetab`, указанном при запуске драйвера, ведется поиск строки с заданным именем ЛИНТЕР-сервера;
- если имя ЛИНТЕР-сервера не найдено, происходит отказ в соединении, в противном случае определяются сетевой адрес компьютера и номер порта, выделенный на сервере для взаимодействия с клиентским приложением. Если на указанном сервере запущен сетевой драйвер сервера с указанным номером порта, то все запросы данного клиентского приложения направляются этому экземпляру сетевого драйвера сервера, который, в свою очередь, будет передавать их связанному с ним ядру СУБД.

Локальный клиент

Для соединения с БД, расположенной на том же компьютере, что и клиентское приложение, необходимо:

1. Указать местоположение нужной БД, например, `SY00=/linter/db/kassa; export SY00`.
2. Задать номер очереди сообщений, например, `LINTER_MBX=30001`.
3. Запустить на выполнение экземпляр СУБД ЛИНТЕР: `linter`.

Активизация сервера

Локальный сервер

Вариант 1: на сервере одна БД

Выполнить следующие действия:

1. Если переменная окружения `SY00` установлена, выполнить команду `linter`.
2. Если переменная `SY00` не установлена, выполнить команду
`linter /base=<каталог БД> /local`

Вариант 2: на сервере несколько локальных БД

В этом случае работа с локальными ЛИНТЕР-серверами выполняется аналогично работе с удаленными ЛИНТЕР-серверами (см. ниже).

Удаленный сервер

Порядок запуска СУБД ЛИНТЕР зависит от того, к скольким ЛИНТЕР-серверам необходимо обеспечить одновременный доступ.

Вариант 1: на сервере одна БД

Выполнить следующие команды:

1. `linter` (запуск ядра СУБД).
2. `db_s_tcp` (запуск сетевого драйвера сервера).

Вариант 2: на сервере несколько БД

Для иллюстрации запуска БД в качестве примера используется следующий фрагмент файла `nodetab`:


#ЛИНТЕР- #сервер	Протокол	Сетевое имя	Номер порта	Местоположение БД
BANK	TCPIP	100.101.67.90		# ../db/bank
Sale	TCPIP	100.101.67.90	1061	# ../db/com/sale
Person	TCPIP	100.101.67.90	1062	# ../db/com/person
Plan	TCPIP	100.101.67.90	1063	# /usr/linter/db

БД могут запускаться в любой очередности.


Запуск БД Sale

1. Указать местоположение БД:
`SY00=<префикс>/db/com/sale`
`export SY00`
где
`<префикс>` – начальный путь к каталогу БД, например:
`/usr/linter` или `/home/user`
2. Выбрать из файла `nodetab` номер порта (1061), закрепленный за этой БД. Данное значение соответствует ключу `/P` сетевого драйвера сервера (`/P=1061`).
3. Задать номер очереди сообщений: например, 30100:
`LINTER_MBX=30100`
`export LINTER_MBX`
Данное значение соответствует ключу `/M` сетевого драйвера сервера (`/M=30100`).
4. Запустить на выполнение экземпляр СУБД ЛИНТЕР:
`linter`
5. Запустить сетевой драйвер сервера для этого экземпляра СУБД с ключами `/M`, `/P`:
`dbs_tcp /M=30100 /P=1061`

Запуск БД BANK

 В файле `nodetab` БД BANK описана как БД по умолчанию, то есть номер порта для нее не указан.


1. Указать местоположение БД:
`SY00=<префикс>/db/bank`
`export SY00`
2. Выбрать из файла `nodetab` закрепленный за этой БД номер порта, а т.к. он явно не задан, используется значение по умолчанию (1060). Это значение задается в ключе `/P` сетевого драйвера сервера (`/P=1060`).
3. Номер очереди не задаем, а используем значение по умолчанию (30000). Оно соответствует ключу `/M` сетевого драйвера сервера (`/M=30000`).
4. Запустить на выполнение экземпляр СУБД ЛИНТЕР:
`linter`
5. Запустить сетевой драйвер сервера для этого экземпляра СУБД с ключами `/M`, `/P` по умолчанию:
`dbs_tcp`

 Только одна БД может быть запущена таким образом (то есть по умолчанию).

Запуск БД Person

1. Указать местоположение БД:
`SY00= <префикс>/db/com/person`
`export SY00`
2. Выбрать из файла `nodetab` закрепленный за этой БД номер порта (1062). Это значение задается в ключе `/P` сетевого драйвера сервера (`/P=1062`).
3. Задать номер очереди сообщений: например, 30200:
`LINTER_MBX=30200`
`export LINTER_MBX`
Это значение соответствует ключу `/M` сетевого драйвера сервера (`/M=30200`).
4. Запустить на выполнение экземпляр СУБД ЛИНТЕР:
`linter`
5. Запустить сетевой драйвер сервера для этого экземпляра СУБД с ключами `/M`, `/P`:
`dbс_tcp /M=30200 /P=1062`

Аналогично выполняется запуск и всех остальных БД сервера.

 Убедиться, что все необходимые для функционирования СУБД ЛИНТЕР компоненты загружены, можно с помощью команды `ps -ax`.

В списке процессов ОС обязательно должны присутствовать:

- `linter`;
- `sql`;
- `tsp`;
- `intsrt`;
- `dbс_tcp`.

Количество процессов должно соответствовать количеству запущенных экземпляров БД.

Способы соединения с ЛИНТЕР-сервером

Возможны различные способы соединения клиентского приложения с ЛИНТЕР-сервером. Выбор конкретного способа зависит от специфики клиентского приложения и инструментальных средств, использованных при его разработке.

Типичными способами соединения являются:

- через командную строку операционной системы;
- через экран подключения инструментального средства;
- через программный интерфейс, используемый клиентским приложением;
- через специальные команды в некоторых инструментах.

При любом способе соединения необходимо указывать только один параметр – имя ЛИНТЕР-сервера. Использование для соединения логического имени вместо конкретных сетевых параметров делает клиентское приложение (особенно те из них, которые содержат встроенные в свой код параметр соединения с БД) независимым от физического местоположения нужной БД в любой момент времени. Пусть текущее местонахождение БД `Zarplata` в локальной сети БД описывается строкой файла `nodetab`:


```
Zarplata   TCPIP Accountant   1061
```

Если администратор БД по каким-либо соображениям перенес БД `Zarplata` на другой компьютер, например, на `Marketing` с номером порта 1062, то тогда соответствующая строка в файле будет иметь вид:

```
Zarplata   TCPIP Marketing   1062
```

Как видно, сетевые параметры соединения изменились, а логическое имя ЛИНТЕР-сервера, используемое при подсоединении к БД, осталось неизменным.

Примеры сетевых конфигураций СУБД ЛИНТЕР приведены в приложении 2.

 Так как на каждом клиентском компьютере находится свой экземпляр файла `nodetab`, то при изменении положения БД в сети администратор СУБД должен внести соответствующие изменения во все экземпляры файла `nodetab`, в противном случае возможность соединения с некоторыми БД не гарантируется.

Соединение из командной строки

Формат командной строки для соединения с ЛИНТЕР-сервером определяется разработчиком приложения. Утилиты СУБД ЛИНТЕР придерживаются следующего формата:

```
<утилита> -u имя/пароль - n ЛИНТЕР-сервер
```

где:

<утилита> – имя программы, выполняющей соединение с ЛИНТЕР-сервером;

-u – ключ параметра «Регистрационные данные»:

- **имя** – имя пользователя БД;
- **пароль** – пароль пользователя БД;

-n – имя ЛИНТЕР-сервера, с которым устанавливается соединение.

Пример

```
> inl -u SYSTEM/MANAGER -n DEMO
```

Чтобы предотвратить раскрытие пароля во время подключения, можно опустить его в командной строке; в этом случае приложение должно выдать интерактивную подсказку для ввода пароля без отображения на экране, например:

```
> inl -u SYSTEM -n DEMO
Пароль пользователя:
*****
```

Соединение с помощью экранной формы

Для соединения с ЛИНТЕР-сервером в клиентских приложениях с графическим интерфейсом используются специальные экранные формы (окна) (Рис. 16). Пользователь должен ввести имя пользователя БД, его пароль и выбрать из предлагаемого выпадающего списка доступный для соединения ЛИНТЕР-сервер. Пароль остается невидимым, что должно быть стандартным свойством для всех экранных форм подключения.

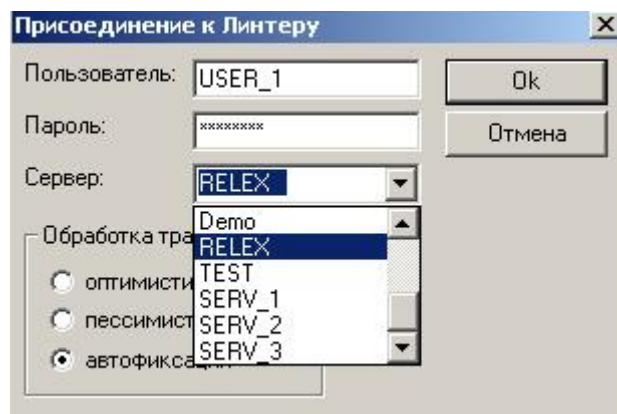


Рис. 16. Соединение через экран подключения

Соединение через программный интерфейс

Каждый программный интерфейс СУБД ЛИНТЕР, предназначенный для разработки клиентских приложений (интерфейс нижнего уровня, прикладной интерфейс, встроенный SQL, PHP, PYTHON и др.) обязательно содержит функцию (оператор) соединения с ЛИНТЕР-сервером. (см. документацию по соответствующему интерфейсу).

Приложение 1

Пример создания файла сетевой конфигурации

Для иллюстрации процесса создания файла сетевой конфигурации СУБД ЛИНТЕР будет использоваться фрагмент локальной вычислительной сети, представленный на рис. 17.

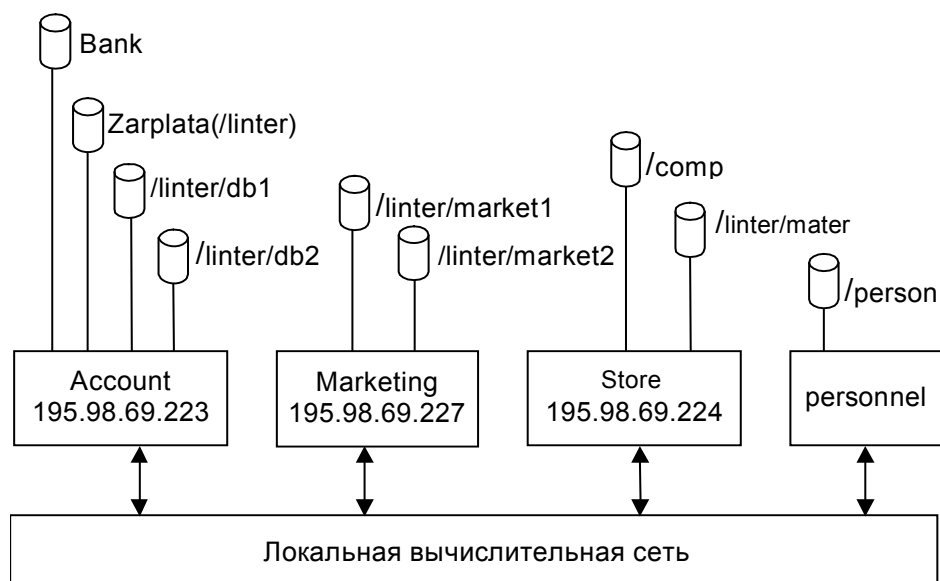


Рис. 17. Фрагмент локальной вычислительной сети

Для создания файла сетевой конфигурации необходимо:

1. Определить, к скольким БД одновременно на каждом компьютере будет разрешен сетевой доступ (то есть доступ с другого компьютера). На этой стадии не решается вопрос, к каким конкретно БД, а определяется только их суммарное количество. Например, если для компьютера **accountant** (см. рис. 17), где установлено четыре БД, мы решим, что в любой момент времени только к двум из них будет разрешен сетевой доступ, это значит, что сетевой доступ будет обеспечен к любым двум БД из перечисленных четырех, к любой из оставшихся БД доступ будет возможен только в локальном режиме. Количество одновременных сетевых доступов и определяет количество задаваемых в таблице **nodetab** портов для этого компьютера.
2. Получить у администратора локальной сети сетевые адреса установленных (или планируемых к установке) компьютеров.
Для протокола TCP/IP (TCP/IPS) это могут быть сетевой IP-адрес или имя, определяемое через службу DNS (Domain Name Service). Например, **100.101.102.103** – числовой IP-адрес, **mycomp.myorg.mydomen** – каноническое DNS-имя, **mycomp** – сопредельное DNS-имя.
3. Внести сетевые адреса в поле **Сетевой Адрес** строк файла **nodetab**.
4. Выяснить у администратора сети тип установленного протокола связи для доступа к компьютеру.
5. Внести обозначение используемых протоколов в поле **Имя протокола** файла **nodetab**.

После выполнения пунктов 1-4 файл **nodetab** может иметь следующий вид:


#	ЛИНТЕР-сервер	Протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера	Тайм-аут клиента	Тайм-аут соединения
		ТСРIP	Accountant				
		ТСРIP	195.98.69.227				

```
TCPIP 195.98.69.226
LOCAL Prsn1
LOCAL 25012
```


6. Для каждой строки **nodetab** создать столько копий этой строки, сколько портов определено для данного компьютера.


Пусть, например, для компьютера **accountant** (см. рис. 17) разрешено три одновременных сетевых доступа к БД, для компьютеров **Marketing** и **Store** – по два сетевых доступа к БД и для компьютера **Personnel** – один сетевой доступ к БД. Тогда файл **nodetab** на данном этапе может иметь следующий вид:

# ЛИНТЕР- # сервер	Протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера	Тайм-аут клиента	Тайм-аут соединения
	TCPIP	Accountant				
	TCPIP	195.98.69.227				
	TCPIP	Marketing				
	LOCAL	Personnel				
	TCPIP	Accountant				
	TCPIP	Accountant				

 При дублировании строки для компьютера 2 в столбце **Сетевое имя** вместо первоначального физического адреса **195.98.69.227** подставлено сетевое имя **Marketing**, тем не менее эти две строки совершенно идентичны с точки зрения сетевых служб. Дублирующие строки для компьютера 1 поставлены в конец файла – это говорит о том, что записи в файле могут располагаться в произвольном порядке.

7. Для каждой записи файла **nodetab** указать номер порта, через который будет происходить взаимодействие клиентского приложения с СУБД ЛИНТЕР на этом компьютере. В пределах одного компьютера все номера портов должны быть уникальными.

 Если на данном компьютере будет использоваться несколько портов для доступа к ЛИНТЕР-серверам, то для исключения конфликтных ситуаций при работе сети допускается использовать только один номер по умолчанию.

 В ОС Unix для получения более наглядной информации о портах TCP/IP по команде **netstat** лучше внести в файл **/etc/services** строку:

```
1060 tpcip Linter
```

После выполнения пункта 7 файл **nodetab** должен иметь следующий вид:

# ЛИНТЕР- # сервер	Протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера	Тайм-аут клиента	Тайм-аут соединения
	TCPIP	Accountant	3			
	TCPIP	195.98.69.227	1			
	TCPIP	Marketing	0x425			
	TCPIPS	195.98.69.226	0x426			

¹ для платформы Win32

² для платформы Unix

³ Используется номер порта по умолчанию.

LOCAL	Personnel	
ТСPIР	Accountant	1061
ТСPIР	Accountant	1062

8. Ввести, при необходимости, в столбцы Тайм-аут сервера, Тайм-аут клиента, Тайм-аут соединения отличные от значения по умолчанию значения тайм-аутов.

Таким образом, видно, что комбинация значений Имя протокола, Сетевой адрес и Сетевой порт в файле `nodetab` задает неповторяющееся (уникальное) значение в пределах строк файла, то есть данному набору значений можно поставить в соответствие обобщенное имя и в дальнейшем, при необходимости, ссылаться на него. Это имя называется «ЛИНТЕР-сервер» – логическое имя для обозначения сетевого имени компьютера и соответствующего сетевого порта на этом компьютере. В качестве имени ЛИНТЕР-сервера следует использовать названия, несущие информацию о содержании БД, например, Bank, Sale. После заполнения поля ЛИНТЕР-сервер файл `nodetab` приобретет свой окончательный вид:

# ЛИНТЕР- # сервер	Протокол	Сетевой адрес	Сетевой порт	Тайм-аут сервера	Тайм-аут клиента	Тайм-аут соединения
Bank	ТСPIР	Accountant				
Tender	ТСPIР	195.98.69.227	0x424			
Sale	ТСPIР	Marketing				
Series	ТСPIPS	195.98.69.226	1060			
Material	ТСPIPS	195.98.69.225	1060			
Cadres	LOCAL	Personnel				
Zaplata	ТСPIР	Accountant	1061			
Postavki	ТСPIР	Accountant	1062			

Приложение 2

Варианты сетевых конфигураций СУБД ЛИНТЕР

Ниже приведены схемы сетевых конфигураций клиентского компьютера и ЛИНТЕР-сервера.

1. Несколько клиентских компьютеров – один ЛИНТЕР-сервер (рис. 18).

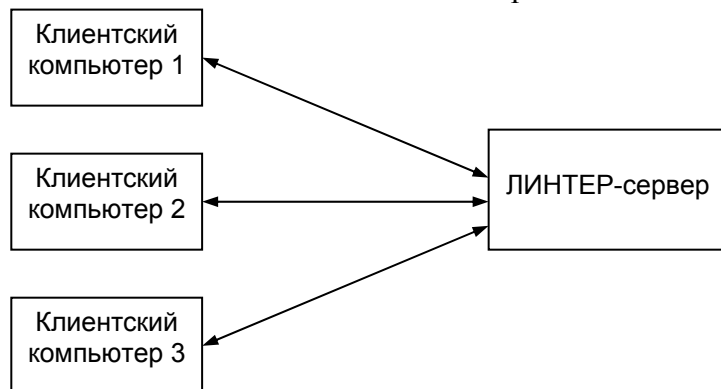


Рис. 18. Сетевая конфигурация с одним ЛИНТЕР-сервером и несколькими клиентами

В данной конфигурации файлы `nodetab` на каждом клиентском компьютере должны ссылаться на один и тот же сетевой адрес ЛИНТЕР-сервера (который задается по-разному для различных типов протоколов), все остальные параметры могут быть заданы индивидуально.

Примеры

Одинаковая сетевая конфигурация:

```
SERVER TCP/IP 195.98.69.227 0x424
```

Индивидуальная для каждого клиентского компьютера сетевая конфигурация:

Для компьютера 1

```
SERVER TCP/IP 195.98.69.227 106 2 20
1 0
```

Для компьютера 2

```
SERVER TCP/IP 195.98.69.227 106 1 30
2 2
```

Для компьютера 3

```
SERVER IPXSP 000001023e60cdddbb01d2d
3 X 1
```

2. Один клиентский компьютер – несколько ЛИНТЕР-серверов (рис. 19).

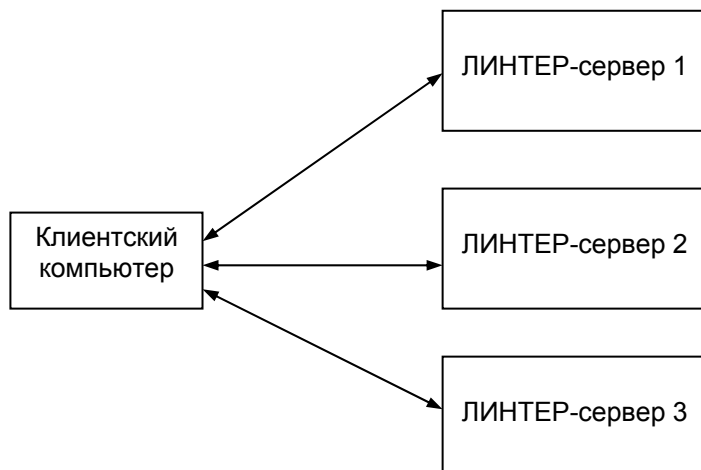


Рис. 19. Сетевая конфигурация с одним клиентом и несколькими ЛИНТЕР-серверами

Пример

Файл nodetab имеет вид:

```
SERVER  TCPIP  195.98.69.200  1060  2  10  2
1
SERVER  TCPIP  www.comp.ru  1062  1  10  5
2
SERVER  TCPIP  195.98.69.201  1063
3
```

3. Несколько клиентских компьютеров – несколько ЛИНТЕР-серверов (рис. 20).

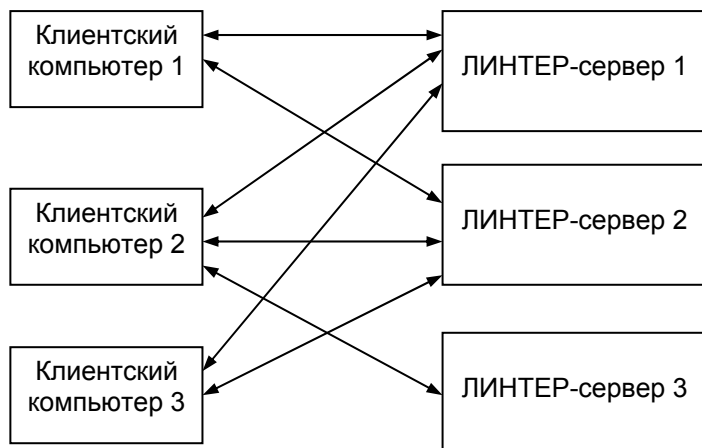


Рис. 20. Сетевая конфигурация с несколькими клиентами и ЛИНТЕР-серверами

Пример

Файлы nodetab имеют вид:

Для компьютера 1

```
SERVER  TCPIP  195.98.69.227  106  2  0
1  0
SERVER  TCPIP  195.98.69.200  106  2
2  S  0
```

Для компьютера 2

SERVER	TCPIP	195.98.69.200	106	1	20
2			2		
SERVER	TCPIP	195.98.69.227	106	2	20
1	S		0		
SERVER	TCPIP	195.98.69.100	106	2	0
3			0		

Для компьютера 3

SERVER	IPXSP	000001023e60cdddbb01d2d			0
3	X	1			
SERVER	TCPIP	195.98.69.227	106	2	
1			0		

4. Несколько клиентских компьютеров – несколько ЛИНТЕР-серверов (рис. 21).

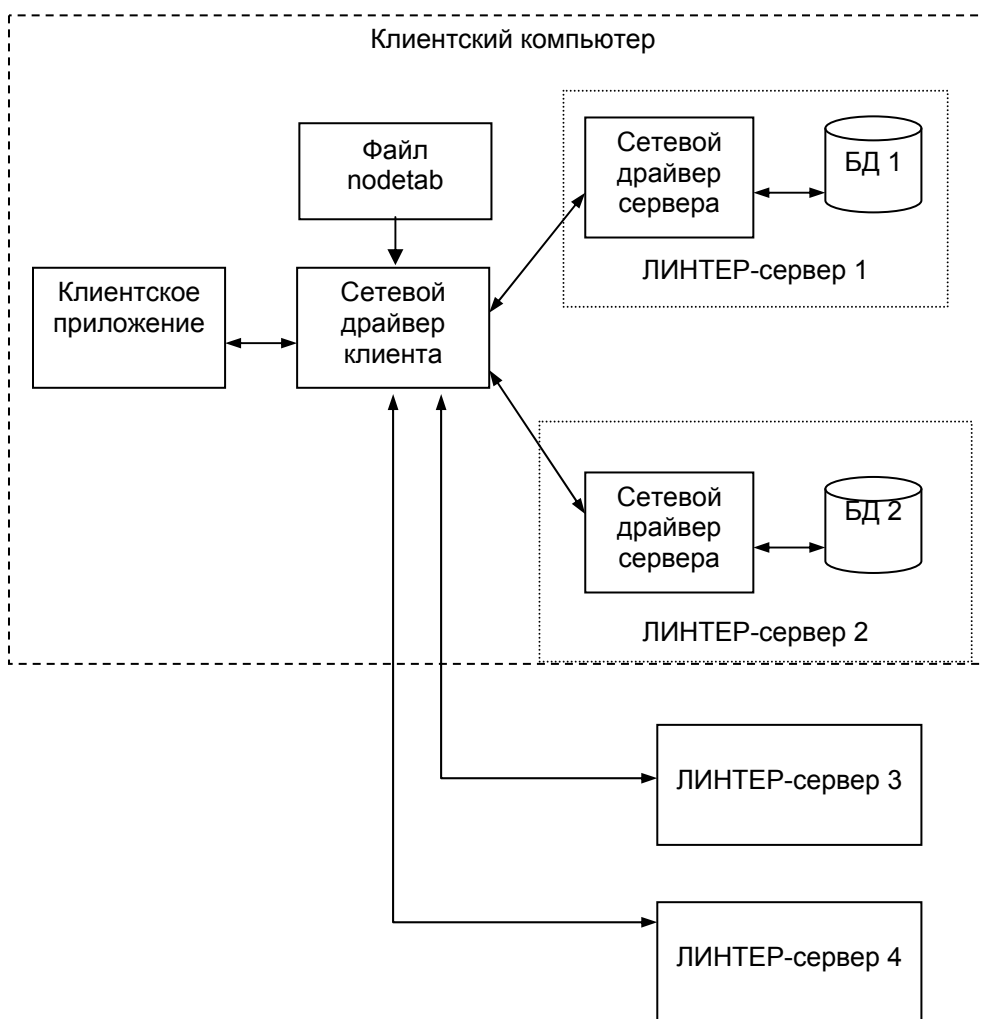


Рис. 21. Сетевая конфигурация с несколькими клиентскими компьютерами и несколькими ЛИНТЕР-серверами

Пример

Файлы nodetab имеют вид:

SERVER	LOCAL				
1					
SERVER	TCPIP	195.98.69.200	106	2	
2	S		0		

Приложение 2

SERVER	TCPIP	195.98.69.227	106	1	20
3			2		
SERVER	IPXSP	000001023e60cdddbb01d2d		0	
4	X	1			

5. Локальный ЛИНТЕР-сервер (без использования сетевых средств) (рис. 22).

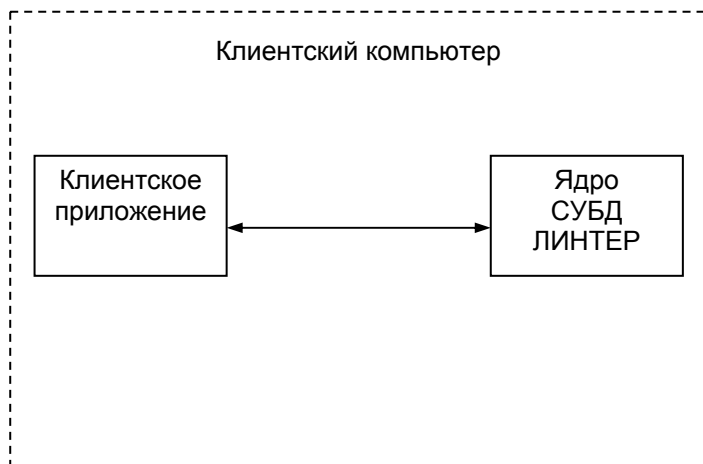


Рис. 22. Сетевая конфигурация локального ЛИНТЕР-сервера

Приложение 3

Проверка работы сетевого клиента в среде ОС Unix

Условия проверки

1. Для проверки используется два компьютера, один из которых является сервером, другой – клиентом.
2. На обоих компьютерах установлена Unix подобные ОС (не обязательно одинаковые).
3. На серверном компьютере установлены:
 - ядро СУБД ЛИНТЕР;
 - демонстрационная БД;
 - утилита inl;
 - сетевой драйвер сервера dbc_tcp;
 - сетевой драйвер клиента dbc_tcp.
4. Переменная окружения SY00 указывает на каталог демонстрационной БД.
5. К списку каталогов переменной окружения PATH добавлен каталог исполняемых файлов СУБД ЛИНТЕР.
6. На клиентской машине установлены:
 - утилита inl;
 - сетевой драйвер клиент dbc_tcp.
7. Если не оговорено особо, то в том же каталоге, где расположен драйвер dbc_tcp, должен находиться конфигурационный файл nodetab с описанием узлов сети.

```
RTCP TCPIP <remote_host>  
RTCPS TCPIPS <remote_host>  
L LOCAL 1254
```

- где <remote_host> должно быть заменено на реальный TCP/IP адрес серверного компьютера.
8. Путь к каталогу, содержащему драйвер dbc_tcp и утилиту inl должен быть добавлен к переменной окружения PATH.
 9. После выполнения очередного пункта проверки клиентский и серверный компьютеры должны быть приведены в исходное состояние, а именно:
 - файл nodetab к описанному выше;
 - процессы ядра СУБД ЛИНТЕР сетевого драйвера сервера и сетевого драйвера клиента, утилиты inl должны быть остановлены;
 - должны быть удалены или приведены к первоначальному все файлы, созданные или измененные в процессе предыдущей проверки.
 10. Под «попыткой установления соединения с узлом А» следует понимать запуск утилиты inl с ключами:

```
>inl -u SYSTEM/MANAGER -n A
```

Проверка запуска

1. На клиентском компьютере выполнить команду:
>dbc_tcp
2. Проверить появление процесса dbc_tcp в списке процессов:
>ps -e | grep dbc_tcp

Проверка завершения работы

1. Запустить процесс сетевого клиента на клиентском компьютере.
2. Определить PID процесса сетевого клиента, просмотрев список всех процессов:

```
>ps -e | grep dbc_tcp
```
3. Послать процессу сетевого клиента сигнал TERM:

```
>kill PID_OF_DBC_TCP
```

где PID_OF_DBC_TCP – идентификатор процесса сетевого сервера `dbc_tcp`.
4. Просмотрев список процессов, убедиться в завершении процесса `dbc_tcp`.
5. Снова запустить процесс сетевого клиента на клиентском компьютере и убедиться в активности процесса `dbc_tcp`.
6. Выполнить команду завершения работы сетевого клиента:

```
>dbc_tcp -U
```

Команда должна завершиться сообщением о посылке сигнала SIGTERM сетевому клиенту. В списке не должно остаться процесса с именем `dbc_tcp`.

 Команда завершения работы отсутствует в ОС QNX.

Проверка протоколов, декларированных в файле `nodetab`

1. Создать файл `nodetab` со следующими узлами:

```
N1 TCPIP <remote_node>
N2 TCPIPS <remote_node>
N3 LOCAL 1234
N4 LOCALI 4321
N5 LOCALS 2501
N6 ATCPIP
N7 ATCPIPS
N8 REZ
N9 IPX/SPX
N10 NPIPE
N11 DECNET
N12 TSR
N13 NETBIOS
N14 ABCDE
```

`<remote_node>` должно быть заменено на реальное доменное имя серверного компьютера.

2. Запустить драйвер `dbc_tcp` с указанием полного пути к созданному файлу `nodetab` из командной строки. При запуске будет выдана информация о неподдерживаемом протоколе ABCDE.
3. Последовательно осуществить попытку соединения с узлами N1-N14:
 - при попытке соединения с узлами N1–N5 будет возвращен код завершения 4006 «Ошибка создания сетевого соединения»;

- при попытке соединения с узлами N6–N13 будет возвращен код завершения 4066 «Сетевой протокол запрещен или не поддерживается»;
- при попытке соединения с узлом N14 будет возвращен код завершения 4050 «Неизвестное имя сервера».

Проверка IP-адреса в файле nodetab

1. На клиентском компьютере в отдельном каталоге создать файл nodetab:

```
N1 TCPIP 127.0.0.1
N2 TCPIP localhost
N3 TCPIP <remote_ip>
N4 TCPIP abcd.abcd.abcd
N5 TCPIPS 127.0.0.1
N6 TCPIPS localhost
N7 TCPIPS <remote_ip>
N8 TCPIPS abcd.abcd.abcd
```

где <remote_ip> должен быть заменен на числовой IP-адрес серверного компьютера.

2. Запустить драйвер dbc_tcp с указанием в командной строке пути к файлу nodetab.
3. При запуске должна быть напечатана информация о некорректности адреса в описании узла N4 и N8.
4. При попытке соединения с узлами N4, N8 будет возвращен код завершения 4050 «Неизвестное имя сервера».
5. При попытке соединения с узлами N1–N5 будет возвращен код завершения 4006 «Ошибка создания сетевого соединения»;
6. При попытке соединения с узлами N1–N3, N5–N8 будет возвращен код завершения 4006 «Ошибка создания сетевого соединения».

Проверка IP-соединения

1. На серверном компьютере запустить ядро СУБД ЛИНТЕР и сетевой драйвер сервера.
2. На клиентском компьютере запустить сетевой драйвер клиента.
3. Осуществить попытку соединения с узлами RTCP (описание файла nodetab для этой проверки см. в пункте «Условия проверки»):

```
>in1 -u SYSTEM/MANAGER -n RTCP
```

4. Должно быть получено приглашение:

```
>SQL
```

5. Выполнить запрос:

```
select * from auto;
```

На терминал должен быть выведен результат запроса.

Проверка тайм-аута клиента

1. На серверном компьютере запустить ядро СУБД ЛИНТЕР и сетевой драйвер сервера.
2. На клиентском компьютере создать файл nodetab:

```
RTCP TCPIP <remote_host> 1060 1 20
```

```
RTCP S TCPIPS <remote_host> 1060 1 20
```

где <remote_host> должно быть заменено на реальный TCP/IP адрес серверного компьютера.

3. Запустить драйвер dbc_tcp с данным файлом nodetab.
4. Открыть соединение из утилиты inl к узлу RTCP.
5. Разорвать сеть на 30 секунд.
6. Восстановить работу сети.
7. Подать любой запрос из утилиты inl. Должен вернуться код завершения 1069 «Неверный номер канала».
8. Повторить действия для узла RTCP S.

Проверка тайм-аута соединения

1. На серверном компьютере запустить ядро СУБД ЛИНТЕР и сетевой драйвер сервера.
2. На клиентском компьютере создать файл nodetab:

```
RTCP TCPIP <alien_host> 1060 1 20 10
```

```
RTCP S TCPIPS <alien_host> 1060 1 20 10
```

где <alien_host> – сетевой адрес реального компьютера, расположенного за межсетевым экраном. Межсетевой экран должен быть настроен так, чтобы не пропускает пакеты от компьютера с адресом <alien_host> к клиентскому компьютеру.

3. Запустить драйвер dbc_tcp с данным nodetab.
4. Осуществить попытку соединения к узлу RTCP с помощью утилиты inl.
5. Не более чем через 10(+4) сек утилита inl должна выдать сообщение о невозможности определения версии БД. По истечении не более 10(+4) сек – вернуть код завершения 4058 «Превышен интервал ожидания установления соединения» или 4006 «Ошибка создания сетевого соединения».
6. Повторить проверку для узла RTCP S.

Проверка соединения через локальный протокол

1. На серверном компьютере установить переменную окружения LINTER_MBX=1254.
2. Запустить ядро СУБД ЛИНТЕР.
3. Сбросить переменную окружения LINTER_MBX.
4. Проверить подключение к локальному ядру. Попытка открытия соединения к локальному ядру должна закончиться неудачно.
5. Создать файл nodetab:

```
N1 LOCAL 1254
```

```
N2 LOCALI 1254
```

```
N3 LOCALS 1254
```

6. На серверном компьютере запустить сетевой драйвер клиента.
7. Осуществить с помощью утилиты inl соединение к узлам N1, N2, N3.

8. Соединение должно быть успешно установлено с узлом N1.
9. В зависимости от версии СУБД ЛИНТЕР должно быть успешно открыто соединение и к одному из узлов N2 или N3.
10. Выполнить запрос (при успешном соединении):

```
select * from auto;
```

На терминал должен быть выведен результат выполнения запроса.

Проверка установки сервера по умолчанию

1. На серверном компьютере запустить ядро СУБД ЛИНТЕР и сетевой драйвер сервер.
2. На клиентском компьютере создать файл nodetab:

```
N1 TCP/IP 12.12.12.12
```

```
N2 remote_host
```

где <remote_host> должно быть заменено на реальный TCP/IP адрес серверного компьютера.

3. Запустить драйвер dbc_tcp с ключом -S и данным nodetab:

```
>dbc_tcp -N nodetab -S N2
```

При запуске на терминал должно быть выведено сообщение о сервере по умолчанию N2.

4. Осуществить соединение к серверу по умолчанию:

```
>inl -u SYSTEM/MANAGER
```

Соединение должно быть успешно установлено.

5. Выполнить запрос:

```
select * from auto;
```

На терминал должен быть выведен результат выполнения запроса.

6. Повторить шаги 1-5 заново, запустив драйвер dbc_tcp с ключом -DEFAULT вместо -S.

Проверка запуска с идентификатором межпроцессного обмена, отличным от идентификатора по умолчанию

1. На серверном компьютере запустить ядро СУБД ЛИНТЕР и сетевой драйвер сервера.
2. На клиентском компьютере запустить драйвер dbc_tcp с ключом -M:

```
>dbc_tcp -M 1255
```

3. Установить переменную окружения NET_MBX=1255.
4. Произвести попытку соединения с узлом RTCP. Соединение должно быть успешно установлено.
5. Выполнить зпрос:

```
select * from auto;
```

На терминал должен быть выведен результат запроса.

