

МОБИЛЬНАЯ
РЕЛЯЦИОННАЯ
СУБД

ЛИНТЕР®

Linter Standard
Linter Bastion
Linter RealTime
Linter Multiversion

Репликация данных

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ

 **РЕЛАКС**®

Товарные знаки

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

Интеллектуальная собственность

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

О документе

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

Адрес

394006, г. Воронеж, ул. 20-летия Октября, 119.
Тел./факс: (473) 2-711-711, 2-778—333.
e-mail: market@relex.ru.

Адрес для корреспонденции

394000, г. Воронеж, а/я 137.

Техническая поддержка

Отдел поддержки и сопровождения программных продуктов:

телефон: (473) 2-711-711 с 9:00 до 18:00 мск.
e-mail: support@relex.ru, market@relex.ru.

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

Оглавление

Предисловие	1
Назначение документа	1
Принятые обозначения и соглашения	1
Назначение репликации данных	3
Механизм репликации данных.....	3
Модель асинхронной репликации	4
Функции компонент асинхронной репликации	5
Описание процесса тиражирования данных	5
Порядок запуска компонент	6
Запуск СУБД ЛИНТЕР	6
Запуск хранилища данных репликации.....	6
Запуск сервера репликации	6
Запуск драйвера клиента	8
Управление средствами поддержки репликации	9
Общая информация.....	9
Создание системных таблиц поддержки репликации	9
В среде MS Windows NT.....	9
В среде UNIX	10
SQL-конструкции для поддержки репликации	11
Объявление сервера репликации.....	11
Удаление сервера репликации	11
Создание правила репликации	11
Изменение правила репликации	12
Удаление правила репликации	13
Синхронизация таблиц согласно правилу репликации.....	13
Возможные проблемы и пути их устранения	14
Администрирование репликационного сервера в MS Windows NT	16
Средства администрирования репликационного сервера.....	17
Добавление сервера репликации в таблицу серверов	17
Удаление сервера репликации из таблицы серверов	18
Создание правила репликации.....	19
Модификация правила репликации	21
Удаление правила репликации.....	22
Просмотр правил репликации.....	23
Запуск/останов репликационного сервера	23
Конфигурирование репликационного сервера	24
Администрирование репликационного сервера в UNIX	25

Оглавление

Настройка репликации на вновь создаваемых базах данных.....	25
Использование поставляемой в дистрибутиве БД.....	26
Настройка существующей БД на использование репликации.....	26
Создание новой базы данных.....	27
Запуск репликации вручную в UNIX.....	28
Настройка репликации вручную.....	30
Файл трассировки.....	32
Формирование имени файла.....	32
Содержание файла трассировки.....	32
Структура базы данных репликации.....	36
Очередь рассылки.....	36
Состояние потока рассылки.....	37
История рассылки.....	38
Приемная очередь.....	38
История приема.....	40
Список реплицируемых таблиц.....	40
Столбцы-источники реплицируемых таблиц.....	40
Столбцы-приемники реплицируемых таблиц.....	40
Список активных правил репликации.....	41
Использование таблиц базы данных репликации и файла трассировки.....	42
Типичные схемы репликации.....	43
Простая репликация с сервера s1 на сервер s2.....	43
Репликация с сервера s1 на сервера s2, s3, s4.....	44
Встречная репликация между серверами s1 и s2.....	46
Репликация с использованием ODBC.....	48
Особенности репликации с использованием ODBC.....	48
Пример репликации с ODBC.....	48
Сообщения сервера репликации.....	50
Сообщения модуля разрешения конфликтов.....	50
Конфликт между существующими и репликационными данными.....	50
Арифметическое разрешение конфликта.....	51
Арифметическое разрешение конфликта при нечисловом поле.....	52
Ошибка при разрешении конфликта.....	52
Ошибка при расширенном разрешении конфликта.....	52

Сообщения модуля инициализации хранилища	53
Ошибка при создании таблицы	53
Сообщение о создании системной таблицы	53
Сообщения модуля выполнения транзакций.....	53
Данные оказались большего размера, чем ожидалось	53
Ошибка чтения дополнительного буфера данных	54
Сообщения процесса приема данных	54
Ошибка добавления новых данных в хранилище	54
Запуск процесса приема данных от конкретного сервера	54
Сообщение процесса приема	55
Сообщение о привязке к порту	55
Запуск управляющего приемного процесса	55
Ошибка распаковки записи	55
Останов в результате отказа СУБД ЛИНТЕР	56
Ошибка выделения памяти.....	56
Внутренняя ошибка сервера репликации	56
Корректный останов сервера репликации по команде пользователя.....	57
Неизвестная внутренняя ошибка сервера репликации.....	57
Сообщения сервера репликации.....	57
Переполнен список одновременно исполняемых транзакций	57
Подается SQL-запрос к БД	57
Некритическая ошибка	58
Запрашиваемая таблица заблокирована другим пользователем.....	58
При обращении к СУБД ЛИНТЕР получен не нулевой код возврата.....	58
Ошибка при открытии канала к СУБД ЛИНТЕР	59
Добавление данных в хранилище	59
Ошибка при добавлении новых данных в хранилище	59
Внутренняя ошибка сервера репликации	60
Правила репликации были изменены и сейчас будут перезагружены	60
Неизвестное сообщение, внутренняя ошибка.....	60
Открывается специальный канал к ядру СУБД ЛИНТЕР	61
Ошибка открытия спецканала	61
Ошибка при обращении к ядру СУБД ЛИНТЕР.....	61
Получен сигнал Ctrl+Break	61
Ошибка при получении данных по спецканалу	62
Сообщения о начале и конце синхронизации таблиц	62
Сообщения о начале и конце перекачки данных	62
Ошибка открытия канала к основной БД	62
Ошибка открытия канала к хранилищу	63
Ошибка запроса имени сервера назначения	63
Ошибка запроса имени таблицы для репликации	63
Ошибка запроса имени владельца таблицы	63
Удаление устаревших данных из системных таблиц	64
Идет очистка хранилища.....	64
Считывание правил репликации	64
Отсутствует таблица правил репликации	64
Ни одного правила репликации не было создано.....	65
Ошибка при считывании правил репликации	65
Считывание структуры таблиц для репликации	65
Ошибка запуска сервера репликации как сервиса	65
Окончание работы сервера репликации	65
Контрольная точка в системном журнале очищена	66

Контрольная точка в системном журнале не найдена	66
Старт сервера репликации	66
Уровень трассировки	66
Задан неверный вес сервера для разрешения конфликтов	66
Устанавливается вес сервера для разрешения конфликтов	67
Указан режим работы	67
Указан используемый файл nodetab	67
Указан pid сервера репликации	67
Началось ожидание данных	67
Освобождение ресурсов перед остановом	68
Происходит останов процесса	68
Сообщения модуля рассылки данных репликации	68
Пришел запрос на правила разрешения конфликтов	68
Неправильное имя сервера репликации	68
Произошло соединение с указанным сервером через указанный порт	69
Останов процесса рассылки на указанный сервер	69
Стартует процесс рассылки на указанный сервер	69
Сообщения процесса очистки	69
Стартует процесс регулярной очистки хранилища	69
Очистка очередей хранилища	69
Сообщения о событиях на реплицируемой БД	70
Закончена транзакция	70
Выполнен откат транзакции	70
Частичный откат транзакции по ошибке ссылочной целостности	70
Выполнена указанная операция	70
Внутренняя ошибка сервера репликации	71

Предисловие

Назначение документа


В документе содержится описание механизма асинхронного тиражирования (репликации) данных для поддержки целостности распределенной базы данных на основе СУБД ЛИНТЕР.

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

Принятые обозначения и соглашения

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

Предисловие

Обозначение	Пример	Значение
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, 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

Назначение репликации данных

Многие современные информационные системы предъявляют достаточно высокие требования к скорости обработки поисковых запросов при условии одновременной работы большого количества клиентов. Кроме того, развиваясь, такие системы должны легко масштабироваться без ущерба для скоростных характеристик системы.

Один из способов удовлетворения этой потребности – создание распределенной базы данных (БД), поддерживающей механизм асинхронной репликации данных. В этом случае вместо одной БД, с которой должны работать все клиенты информационной системы, создается несколько одинаковых (по крайней мере, частично) серверов БД на разных машинах и/или узлах сети. Клиенты имеют доступ к некоторому распределяющему устройству (реализованному аппаратно или программным методом), которое при появлении нового клиента оценивает загрузку каждого сервера БД и направляет клиента к наименее загруженному, с которым он (клиент) и будет работать до отсоединения.

Сервера БД связаны между собой и все сделанные изменения пересылают друг другу (тиражируют) с тем, чтобы привести реплицируемые объекты (таблицы) в полное соответствие. Поскольку репликация асинхронная, то этот процесс происходит не сразу, а в течение некоторого времени, в ходе которого данные на разных серверах будут отличаться.

Такое построение позволяет значительно (в самом лучшем случае, прямо пропорционально количеству серверов БД) увеличить производительность системы и наращивать ее по мере роста нагрузки (увеличения количества клиентов или размеров БД) простым прибавлением серверов БД в информационную систему.

Для управления системой на логическом уровне используются правила репликации, которые создаются с помощью языка БД SQL и представляют собой описание того, какие объекты, куда и каким образом реплицировать.

Механизм репликации данных

Для поддержки целостности распределенной БД в СУБД ЛИНТЕР используется механизм асинхронного тиражирования (далее по тексту – репликации) транзакций.

Суть механизма асинхронного тиражирования состоит в том, что обработка данных выполняется локально, а распределенные данные копируются на тот сервер, где они должны использоваться. При таком методе поддержки логической целостности распределенной БД имеет место некоторая рассинхронизация состояния локальных БД во времени, т.е. изменение состояния одной локальной базы данных отстает от изменения другой локальной базы данных во времени.

Если один из серверов системы, требующих обновления тиражируемых данных, выходит из строя, то система продолжает работать с остальными, при этом обновление данных на сервере после его ремонта произойдет автоматически, т.е. ошибка на одном узле глобальной сети не повлияет на работу остальных узлов.

Механизм асинхронного тиражирования транзакций гарантирует доставку измененных данных на вторичные серверы непосредственно после завершения транзакции, если сервер доступен, или сразу после подключения сервера к сети. Такой подход предполагает хранение дублирующей информации в различных узлах сети и может обеспечить, по сравнению с другими подходами к репликации, снижение трафика,

улучшение времени ответа системы, а также позволяет оптимизировать нагрузку на серверы.

Асинхронная репликация, в отличие от двухфазной синхронизации, не обеспечивает полной синхронности информации на всех серверах в любой момент времени. Синхронизация происходит через некоторый, обычно небольшой, интервал времени, величина которого определяется быстродействием соответствующего канала связи. Для большинства задач кратковременное наличие устаревших данных в удаленных узлах вполне допустимо.

Вместе с тем, асинхронная репликация транзакций принципиально обеспечивает целостность данных, так как объектом обмена данными здесь является логическая единица работы – транзакция, а не просто данные из измененных таблиц.

Механизм асинхронной репликации поддерживает режим защищенной передачи данных по протоколу SSL (в Unix-системах). Длина ключа всегда равна 512 битам. Ключ не читается из файла, а всегда создается при запуске. При использовании протокола SSL аутентификация не выполняется, осуществляется только кодирование сетевого трафика.

Модель асинхронной репликации

В системе асинхронной репликации участвуют два или более серверов БД, на каждом из которых работает СУБД ЛИНТЕР (**БД-источник репликации**) и **процессы репликации**, которые обеспечивают выполнение правил репликации (рис. 1). Объектами репликации являются пользовательские таблицы базы данных, список которых вместе с правилами и адресами рассылки хранится в БД-источнике репликации (на рис. 1 – ЛИНТЕР).

Сервер репликации **lreplserver** представляет собой специальный процесс, который получает информацию об измененных данных от СУБД ЛИНТЕР и сохраняет ее в очередях репликации хранилища данных репликации. Хранилище данных репликации представляет собой соответствующим образом урезанное ядро другой СУБД ЛИНТЕР, которое используется сервером репликации для получения данных, подлежащих репликации и рассылке. Процессы приема(rcv) и рассылки(snd) порождаются сервером репликации (в NT это нити процесса lreplserver) и используются для передачи, приема и выполнения транзакций.

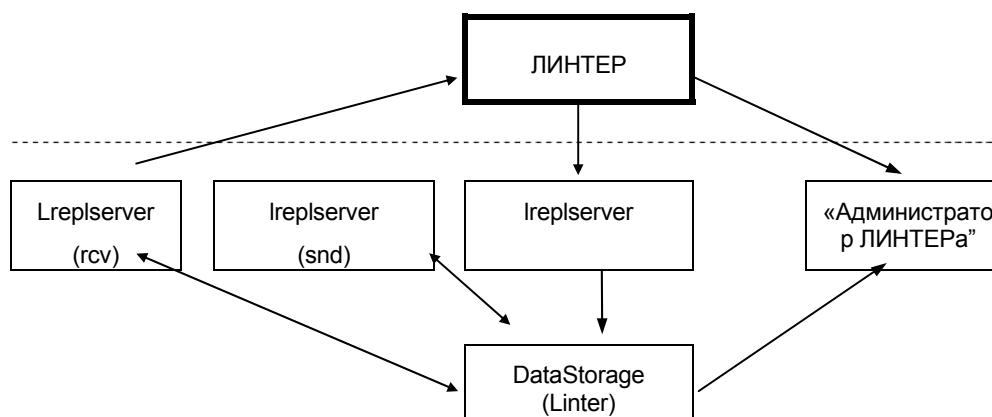


Рис. 1. Модель репликации в СУБД ЛИНТЕР

Управление асинхронной репликацией в среде Win32 выполняется с помощью утилиты администрирования «Сетевой администратор», а в среде Unix – с помощью специального командного файла.

Функции компонент асинхронной репликации

Функции компонент системы асинхронной репликации СУБД ЛИНТЕР приведены ниже:

<u>Компонента</u>	<u>Функциональное назначение</u>
ЛИНТЕР	Основное ядро СУБД ЛИНТЕР, работает независимо от остальных компонент. Должно обеспечивать только одну дополнительную функцию: выдавать для сервера репликации измененные записи
Сервер репликации lreplserver	Обеспечивает отслеживание транзакций в основной БД и запись всех необходимых для репликации изменений в хранилище данных репликации. Организует выполнение правил репликации. Формирует данные для рассылки в хранилище данных репликации. При старте запускает дополнительные процессы (в NT – нити), отвечающие за отсылку, прием и выполнение репликационных данных. Позволяет выполнять минимальное администрирование. Запускается отдельно и независимо от СУБД ЛИНТЕР
Хранилище данных репликации	Экземпляр СУБД ЛИНТЕР, который хранит данные для рассылки. К этим данным имеют доступ сервер репликации.
Утилита «Сетевой администратор»	Управляет запуском/остановом репликационного сервера

Описание процесса тиражирования данных

Элемент очереди рассылки включает в себя полную информацию о старом и новом состоянии записи, адрес назначения, номер канала, производящего операцию, номер транзакции и время операции.

Эта информация заносится в таблицу очереди рассылки на сервере репликации. В качестве первичного ключа этой таблицы используется время операции.

Процесс приема сервера репликации получает данные и помещает их в приемную очередь, структура которой похожа на структуру таблицы очереди рассылки, после помещения в эту очередь он формирует ответ, уведомляющий отправителя о нормальном приеме. Одновременно происходит собственно репликация, коды завершения сохраняются в таблице приемной очереди.

Повторное прохождение одного и того же блока отслеживается с помощью времени операции.

В качестве протокола проделанной работы используется файл трассировки lreplserver.log. Он может формироваться как в виде одного файла, так и в виде группы файлов вида: lreplserver.log.NNN, где NNN – порядковый номер файла.

Порядок запуска компонент

Запуск СУБД ЛИНТЕР

Для использования СУБД ЛИНТЕР в режиме репликации:

- база данных должна содержать системные таблицы `SERVERS` и `$$$REPL`. Эти таблицы создаются с помощью файла `distr.sql`;
- таблица `SERVERS` должна содержать имена удаленных серверов, на которые будут тиражироваться данные. Имена серверов заносятся в таблицу с помощью SQL-запроса
CREATE SERVER <имя сервера>;
или
CREATE NODE <имя сервера>;
- файл `nodetab`, используемый при рассылке данных репликации, должен также содержать имена удаленных серверов, на которые будут тиражироваться данные. Порт в записи `nodetab` должен совпадать с портом, указанным при запуске сервера репликации на принимающей стороне. Протокол для такого сервера в файле `nodetab` должен быть `ATCPIP` (или `ATCPIPS` в случае использования протокола `SSL` на `UNIX`).

Запуск хранилища данных репликации

В качестве хранилища данных репликации запускается:

- в среде `UNIX` – в другой сессии и под другим значением переменной `LINTER_MBX` экземпляр СУБД ЛИНТЕР;
- в среде `MS Windows NT` – экземпляр СУБД ЛИНТЕР с ключом `-name=XXXX`.

БД хранилища должна иметь имя по умолчанию `REPLSERV`. Если это не так, то имя БД указывается явно при запуске сервера репликации (см. стр. 6). Однако, в этом случае запуск может быть осуществлен только вручную, без использования утилиты «Сетевой администратор».

В БД хранилища данных репликации должны быть рабочие таблицы, которые создаются с помощью файла `arepl.sql` при установке репликационного сервера. При старте сервер репликации проверяет наличие рабочих таблиц, и если есть необходимость, пересоздает их.

Запуск сервера репликации

Общие ключи программы (для `Win32` и `Unix`):

<u>Ключ</u>	<u>Описание</u>
<code>-u <имя>/<пароль></code>	Имя и пароль пользователя БД-источника репликации
<code>-us <имя>/<пароль></code>	Имя и пароль пользователя хранилища данных репликации

Ключ	Описание
<code>-ux <имя>/<пароль></code>	Имя и пароль пользователя, который будет заносить приходящие данные в БД, если эти данные имеют ненулевой RAL (Read Access Level) или WAL (Write Access Level). Этот пользователь должен иметь соответствующие RAL и WAL.
<code>-t <число></code>	Уровень трассировки сообщений сервера репликации (значение в диапазоне от 0 до 4, по умолчанию 1)
<code>-log <число></code>	Создание log-файла. <ЧИСЛО> задает уровень трассировки сообщений, помещаемых в log-файл. Если число не задано, то устанавливается уровень, равный уровню трассировки на консоль
<code>-lvol <число></code>	Задаёт размер тома log-файла в килобайтах. Используется для разбиения log-файла на куски, которые можно было бы удалять или архивировать
<code>-lcount <число></code>	Задаёт максимальное количество томов log-файла
<code>-s <имя БД хранилища></code>	Имя БД хранилища данных репликации (если оно отличается от стандартного REPLSERV)
<code>-l <имя БД></code>	Имя БД (если она не по умолчанию)
<code>-p <номер порта></code>	Номер порта, через который осуществляется доступ к хранилищу данных репликации. По умолчанию используется 1080
<code>-clear <режим очистки></code>	Очистка хранилища данных репликации. Допустимые режимы: <ul style="list-style-type: none"> • ALL – очистка всей БД; • NUM – регулярная очистка, здесь NUM задает интервал в секундах, через который будет активизироваться процесс очистки. Старые данные не удаляются, а перемещаются в таблицы хранения
<code>-nohist</code>	Не хранить историю операций (используется совместно с ключом <code>-clear num</code>). Задаёт режим реального удаления устаревших данных
<code>-stop</code>	Отменить контрольную точку, установленную репликацией сервером репликации
<code>-down</code>	Останов сервера репликации
<code>-cfg <спецификация файла></code>	Задаёт имя текстового файла, содержащего ключи запуска <code>lreplserver</code>
<code>-? /? -h</code>	Получение справочной информации
<code>-initstorage</code>	Происходит полное пересоздание таблиц хранилища с удалением всех данных
<code>-n <путь></code>	Путь к файлу <code>nodetab</code> , который будет использоваться для определения адреса сервера, на который идет репликация. По умолчанию ищется в каталоге запуска сервера репликации

Назначение репликации данных

<u>Ключ</u>	<u>Описание</u>
-lpath <путь>	Путь к файлу трассировки.
-w <число>	«Вес» сервера, используемый при разрешении конфликтных ситуаций (по умолчанию 3)
-odbcsource <DSN>	Задаёт источник данных для ODBC-драйвера (в случае если репликация должна выполняться с помощью ODBC-lhfqthf)

Ключи только для Win32:

<u>Ключ</u>	<u>Описание</u>
-d	Режим отладки (отладочная информация будет выдаваться на консоль). В этом случае сервер репликации запускается не как сервис, а как обычное приложение

Ключи только для Unix:

<u>Ключ</u>	<u>Описание</u>
-ssl	Использовать протокол SSL
-daemon	Запускать lreplserver в фоновом режиме. Этот ключ должен обязательно находиться в командной строке, из CFG файла (ключ -cfg) он восприниматься не будет



Параметры ключа могут задаваться как через пробел, так и через знак «=».

Примеры

```
lreplserver -u=SYSTEM/MANAGER -us=SYSTEM/MANAGER -s=RDSTG  
-daemon -p=1080 -t=3 -log=4
```

```
lreplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -p 1080 -t 3 -log  
4
```

```
lreplserver -daemon -cfg file.cfg
```

Сервер репликации запускает автоматически процессы отсылки, приема и выполнения транзакций.



Для поиска каталога запуска lreplserver использует переменную среды окружения PATH.

Запуск драйвера клиента

Для доступа к хранилищу данных репликации необходимо запустить драйвер клиента DBC_TCP. Если сервер репликации запускается с использованием утилиты linsdm, то это происходит автоматически. Если нет, то запуск драйвера клиента DBC_TCP выполняется на БД-источнике стандартным образом.

Управление средствами поддержки репликации

Общая информация

Чтобы репликационный ЛИНТЕР-сервер мог тиражировать данных из БД-источника в другие БД, он должен иметь информацию о том, какие именно таблицы БД-источника должны тиражироваться и на какие именно удаленные БД. Эта информация задается с помощью так называемых правил репликации. Правила репликации создаются и хранятся в БД-источнике репликации в специальной системной таблице \$\$\$REPL. Кроме того, БД-источник должна содержать системную таблицу SERVERS со списком всех удаленных ЛИНТЕР SQL-серверов, на которые ссылаются правила репликации. Имена ЛИНТЕР SQL-серверов, вносимые в таблицу SERVERS, должны совпадать с именами ЛИНТЕР SQL-серверов в файле `nodetab`, используемым ЛИНТЕР SQL-сервером на данном компьютере для доступа к удаленным узлам, причем протокол работы с таким сервером в файле `nodetab` должен быть указан ATCPIP (или ATCPIPS в случае использования протокола SSL на UNIX).

Создание системных таблиц поддержки репликации

Для создания системных таблиц репликации следует выполнить SQL файл `distr.sql`.

В среде MS Windows NT

Для создания системных таблиц репликации:

1. запустить на выполнение утилиту «Сетевой администратор»
Start ⇒ Programs ⇒ RDBMS Linter SQL ⇒ Linter Administrator;
2. раскрыть дерево серверов и выделить на нем БД-источник репликационного ЛИНТЕР-сервера, для которой должны быть созданы таблицы поддержки репликации (рис. 2);

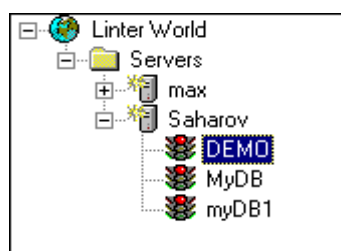


Рис. 2. Выбор БД-источника для создания таблиц поддержки репликации

3. выполнить команду **Database ⇒ Startup ...**, или нажать клавишу <F9>, или щелкнуть на пиктограмме , или щелкнуть правой кнопкой мыши и в выпадающем меню выбрать пункт **Startup**. Появится диалоговое окно запуска БД (рис. 3);

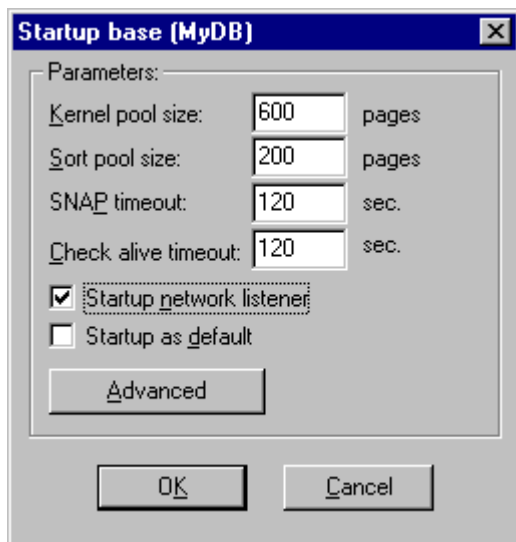


Рис. 3. Диалоговое окно параметров запуска ЛИНТЕР SQL-сервера

- установить флажок Start network listener, если это необходимо;
- нажать кнопку ОК;
- вернуться в дерево ЛИНТЕР-серверов и проверить, что БД запущена;
- на локальном компьютере запустить сетевой драйвер клиента;
- проверить, что с локального компьютера видна БД-источник репликационного сервера и что физический доступ к ней возможен;
- выполнить команду MS Windows для запуска утилиты inl:
Start ⇒ Programs ⇒ RDBMS Linter SQL ⇒ Interactive SQL;

Пункт меню может отличаться от **RDBMS Linter SQL** если он был изменен при установке ЛИНТЕР.

- в ответ на подсказку утилиты inl ввести, например:
`SQL>_d:\Program Files\linter\dict\distr.sql`

Путь к файлу может отличаться от указанного. Файл `distr.sql` находится в подкаталоге `dict` каталога установки ЛИНТЕР.

В среде UNIX

Для создания системных таблиц репликации:

- перейти в каталог, где находится БД-источник репликации, например:
`[sh]$cd ~/linter/db`
- запустить на выполнение ядро СУБД ЛИНТЕР:
`[sh]$../bin/linter`
- запустить программу командного интерфейса СУБД ЛИНТЕР:
`[sh]$~/linter/bin/inl -u SYSTEM/MANAGER`
- В ответ на подсказку утилиты inl ввести путь к командному файлу `distr.swl` создания таблиц поддержки репликации, например:
`SQL>_~/linter/dict/distr.sql`

 Путь к файлу может отличаться от указанного. Файл `distr.sql` находится в подкаталоге `dict` каталога установки ЛИНТЕР.

SQL-конструкции для поддержки репликации

Объявление сервера репликации

Функция

Добавление сервера репликации в системную таблицу серверов репликации.

Спецификация

<объявить сервер репликации>::=

```
CREATE { NODE | SERVER } <имя сервера>
```

Синтаксические правила

- 1) <имя сервера> – идентификатор длиной не более 8 символов.

Общие правила

- 1) в БД-источнике репликации должна быть системная таблица `SERVERS`;
- 2) <имя сервера> должно совпадать с одним из имен серверов репликации из файла `nodetab` с протоколами `ATCPIP` (или `ATCPIPS` в случае использования протокола `SSL` на `UNIX`);
- 3) для объявления сервера репликации необходимо иметь права администратора БД-источника.

Удаление сервера репликации

Функция

Удаление сервера репликации из системной таблицы серверов репликации.

Спецификация

<удаление сервера репликации>::=

```
DROP { NODE | SERVER } <имя сервера>
```

Синтаксические правила

- 1) <имя сервера> – идентификатор длиной не более 8 символов.

Общие правила

- 1) <имя сервера> должно присутствовать в системной таблице `SERVERS`;
- 2) для удаления сервера репликации необходимо иметь права администратора БД-источника;
- 3) при удалении сервера также удаляются все репликационные правила, которые ссылаются на него.

Создание правила репликации

Функция

Создание правила репликации и запись его в системную таблицу `$$$REPL`.

Спецификация

```
<создать правило репликации>::=  
CREATE REPLICATION RULE <имя правила>  
FOR <имя таблицы источника>  
TO <имя таблицы приемника>  
ON {NODE | SERVER} <имя узла>  
[ USER <имя пользователя>]  
[ PASSWORD '<пароль>']  
[ ENABLE | DISABLE]  
[ SYNC | ASYNC]  
[ IGNORE OLD VALUE  
| CHECK OLD VALUE  
| CORRECT NUMBERS];
```

 **CHECK OLD VALUE** и **CORRECT NUMBERS** зарезервированы для будущего применения.

Синтаксические правила

- 1) <имя правила> – идентификатор длиной не более 66 символов уникальный в БД;
- 2) <имя таблицы источника> и <имя таблицы приемника> должны задавать имена базовых пользовательских таблиц. Использовать имена представлений и системных таблиц не допускается;
- 3) <имя узла> задает имя ЛИНТЕР-сервера, в котором находится таблица-приемник;
- 4) <имя пользователя> – имя пользователя БД-приемника;
- 5) <пароль> – пароль пользователя БД-приемника;
- 6) по умолчанию используются значения **ENABLE**, **ASYNC** и **IGNORE OLD VALUE**.

Общие правила

- 1) в БД-источнике репликации должны быть системные таблицы **SERVERS** и **\$\$\$REPL**;
- 2) структуры таблицы-источника и таблицы-приемника должны быть идентичны;
- 3) репликация таблиц, содержащих данные типа **Extfile**, не допускается;
- 4) таблица-источник и таблица-приемник должны обязательно иметь столбец с атрибутом **PRIMARY KEY**;
- 5) создавать правило репликации имеет право только владелец тиражируемой таблицы;
- 6) **ENABLE** – правило активно, **DISABLE** – правило неактивно;
- 7) **SYNC** – режим синхронной, **ASYNC** – асинхронной репликации. При асинхронном режиме тиражирование данных выполняется по мере возможности.

 В данной версии СУБД ЛИНТЕР опция **SYNC** не работает.

- 8) режимы репликации при возникновении рассогласования между значением записи в базах данных;
- 9) **IGNORE OLD VALUE** – игнорировать несовпадение старого значения;
- 10) **CHECK OLD VALUE** – обязательно проверить старое состояние и вернуть ошибку, если нет полного совпадения;
- 11) **CORRECT NUMBERS** – если не совпадают числовые значения, то сохранить разницу между старым и новым значением.

Изменение правила репликации

Функция

Изменение свойств существующего правила репликации.

Спецификация

<изменить правило репликации>::=

```
ALTER REPLICATION RULE <имя правила>  
[ PASSWORD '<пароль>']  
[ ENABLE | DISABLE]  
[ SYNC | ASYNC]  
[ IGNORE OLD VALUE  
| CHECK OLD VALUE  
| CORRECT NUMBERS];
```

Синтаксические правила

- 1) <имя правила> – идентификатор длиной не более 66 символов уникальный в БД;
- 2) <пароль> – пароль пользователя БД-приемника;
- 3) по умолчанию используются значения **ENABLE**, **ASYNC** и **IGNORE OLD VALUE**.

Общие правила

- 1) <имя правила> – это имя правила репликации, которое было создано с помощью команды **CREATE REPLICATION RULE**.
- 2) изменять правило репликации имеет право только владелец тиражируемой таблицы.
- 3) значение параметров **ENABLE**, **DISABLE**, **SYNC**, **ASYNC**, **IGNORE OLD VALUE**, **CHECK OLD VALUE**, **CORRECT NUMBERS** см. стр. 11.

Удаление правила репликации

Функция

Удаление существующего правила репликации.

Спецификация

<удалить правило репликации>::=

```
DROP REPLICATION RULE <имя правила>;
```

Синтаксические правила

- 1) <имя правила> – идентификатор длиной не более 66 символов уникальный в БД.

Общие правила

- 1) <имя правила> – это имя правила репликации, которое было создано с помощью команды **CREATE REPLICATION RULE**;
- 2) удалить правило репликации имеет право только владелец тиражируемой таблицы.

Синхронизация таблиц согласно правилу репликации

Функция

Синхронизация таблицы-источника и таблицы-приемника. Эта конструкция используется, как правило, после того, как создано новое правило репликации. Перед тем, как начать его использовать, необходимо обеспечить идентичность участвующих в репликации таблиц-источника и таблицы-приемника.

Спецификация

<синхронизация таблиц> ::=

SYNCHRONIZE REPLICATION RULE <имя правила>;

Синтаксические правила

- 1) <имя правила> – идентификатор длиной не более 66 символов уникальный в БД.

Общие правила

- 1) <имя правила> – это имя правила репликации, которое было создано с помощью команды **CREATE REPLICATION RULE**;
- 2) синхронизировать таблицы имеет право только владелец тиражируемой таблицы.

Возможные проблемы и пути их устранения

Ниже описаны некоторые проблемы, которые могут возникнуть при работе асинхронной репликации.

<u>Проблема</u>	<u>Способ устранения</u>
База данных хранилища постоянно растет и достигла больших размеров	<p>Следует очистить базу данных хранилища, запустив сервер репликации с ключом -clear ALL, например:</p> <pre>Trplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -clear ALL</pre> <p>Следует заметить, что этот вызов удалит не только устаревшие данные, но и актуальные, поэтому таким вызовом желательно пользоваться тогда, когда все данные тиражированы и наступила временная пауза в изменении данных на основной БД.</p> <p>Для регулярной автоматической очистки следует запускать сервер репликации с ключами -clear 3600 и -nohist:</p> <pre>Trplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -clear 3600 -nohist</pre> <p>Здесь 3600 – время в секундах, процесс очистки будет запускаться регулярно раз в час. Можно указать другое время. Если не задавать ключ -nohist, то данные, предназначенные для очистки реально не удаляются, а перемещаются в таблицы DISPATCH_QUEUE_HIST и RECEIVING_QUEUE_HIST, их можно удалить вручную, как описано на с. 38, 40</p>
Системный журнал реплицируемой базы данных сильно вырос (файлы *.61)	<p>Возможно, сервер репликации когда-то был запущен, что-то реплицировалось, а потом он был остановлен и больше не запускался. При этом данные, предназначенные для репликации, временно хранятся в системном журнале БД. В этом случае можно либо запустить сервер репликации, что приведет к перекачиванию этих данных из системного журнала в хранилище данных репликации с последующей</p>

<u>Проблема</u>	<u>Способ устранения</u>
	рассылкой; либо остановить накопление данных, при этом системный журнал будет освобожден, но данные, накопленные за время простоя сервера репликации, уже не будут переданы на сервера назначения. Это делается с помощью ключа <code>-stop</code> : <pre>lreplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -stop</pre>
Сервер репликации не запускается	Скорее всего, не запущена основная БД либо БД хранилища, или же неверно указаны имена или пароли к ним. Для точной диагностики нужно запустить сервер репликации с трассировкой, добавив ключ <code>-log 4</code> : <pre>lreplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -log 4</pre> Формат файла трассировки описан на с. 32
Данные не реплицируются	Самое первое, что следует проверить, это: <ul style="list-style-type: none">• есть ли правило репликации;• на какой сервер оно указывает;• какой IP адрес для этого сервера указан в файле <code>nodetab</code>;• какой протокол для этого сервера указан в файле <code>nodetab</code> (должен быть ATCPIP). Однако, причин может быть много, для точной диагностики нужно запустить сервер репликации с трассировкой, добавив ключ <code>-log 4</code> : <pre>lreplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -log 4</pre> Формат файла трассировки описан на с. 32 Наиболее вероятная причина отсутствие описание сервера, указанного в правиле репликации в файле <code>nodetab</code> . Следует проверить, выполнены ли все шаги настройки репликации как описано на стр.

Репликация меток доступа

Средства

Сервер репликации предоставляет возможность реплицировать данные с сохранением меток доступа, которые были присвоены им в исходной базе данных. Для того чтобы репликация таких данных стала возможна, сервер репликации на приемной стороне следует запустить с ключом `-ux`, например:

```
1replserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -ux  
SYSTEM/MANAGER -s RDSTG
```

Ключ `-ux` задает имя и пароль пользователя, RAL (Read Access Level) которого будет больше или равен максимально возможному RAL реплицируемых данных, а WAL (Write Access Level) будет меньше или равен минимально возможному WAL реплицируемых данных. Все записи, имеющие метки RAL и WAL больше единицы будут занесены в базу данных на приемной стороне вышеописанным пользователем.


Для того чтобы данные, имеющие метки доступа, были прореплицированы корректно, в приемной базе данных должны быть созданы такие же уровни доступа, какие есть и в передающей базе (такие же в числовом выражении, текстовые идентификаторы уровней могут различаться).

Если приемная сторона получит запись, снабженную метками доступа и при этом ключ `-ux` не будет указан, то будет сделана попытка занести данные через пользователя, указанного ключом `-u`. Если он не имеет соответствующих прав, то запись не сможет быть внесена в базу данных.

Администрирование репликационного сервера в MS Windows NT

Средства администрирования репликационного сервера

Администрирование репликационного сервера возможно следующими способами:

- с помощью командного интерфейса, запуск которого описан на стр. 9;
- с помощью утилиты «Рабочий стол ЛИНТЕРА», запуск которой выполняется следующим образом:
 1. запустить на выполнение утилиту «Сетевой администратор» (см. стр. 9);
 2. запустить сетевой драйвер клиента;
 3. запустить на выполнение из утилиты «Сетевой администратор» утилиту «Рабочий стол ЛИНТЕРА». Для этого перейти в окно утилиты «Сетевой администратор», выбрать пункт меню **Tools** ⇒ **Desktop**, или нажать клавиши <Ctrl>+<D> или пиктограмму ;
 4. в утилите «Рабочий стол ЛИНТЕРА» выполнить соединение с БД-источником репликации;
 5. в утилите «Рабочий стол ЛИНТЕРА» выбрать пункт меню **Database** ⇒ **Replications**. Появится диалоговое окно для ввода информации в таблицы **SERVERS** и **\$\$\$REPL** (рис. 4).

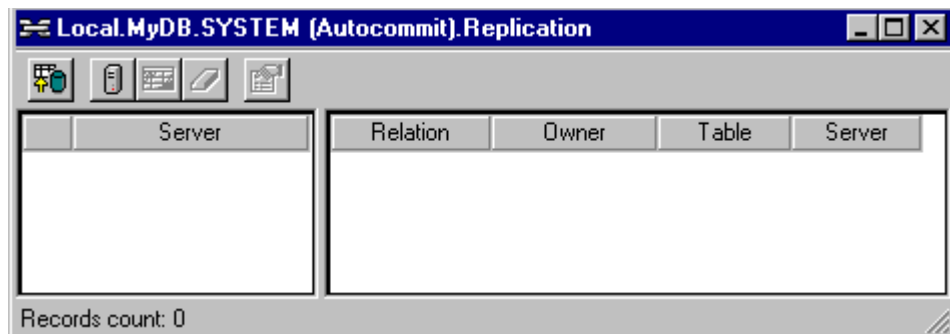


Рис. 4. Диалоговое окно управления объектами репликации

Добавление сервера репликации в таблицу серверов

Добавление нового сервера возможно двумя способами:

- с помощью командного интерфейса;
- с помощью утилиты «Рабочий стол ЛИНТЕРА».


Для добавления записей первым способом:

1. запустить командный интерфейс (см. стр. 9);
2. в ответ на подсказку утилиты `inl` ввести команду
`SQL>create node <имя ЛИНТЕР SQL-сервера>`



где <имя ЛИНТЕР SQL-сервера> – имя добавляемого ЛИНТЕР SQL-сервера, например:

```
SQL>create node sale;  
SQL>create node "Market";  
SQL>create node "Кадры";
```

3. повторить операцию 2 для всех добавляемых в таблицу серверов;
4. завершить выполнение утилиты inl:
SQL>exit

 Добавление нового сервера с помощью SQL-запроса можно выполнить также и средствами утилиты «Рабочий стол ЛИНТЕРА» (см. документ «Рабочий стол ЛИНТЕР» пункт Выполнение SQL-запросов)

Для добавления записей вторым способом:

1. запустить на выполнение утилиту «Рабочий стол ЛИНТЕРА» (см. стр. 17);
2. в утилите «Рабочий стол ЛИНТЕРА» выбрать пункт меню **Database ⇒ Replications**. Появится диалоговое окно для ввода информации в таблицы **SERVERS** и **\$\$\$REPL** (рис. 4);
3. активизировать фрейм таблицы **SERVER** (щелкнуть левой кнопкой мыши в любом месте левого фрейма). В результате должны активизироваться пункт меню **Replication ⇒ New server...** и пиктограмма ;
4. в окне утилиты «Рабочий стол ЛИНТЕРА» выполнить команду **Replication ⇒ New server** или нажать клавиши <Ctrl>+<Ins> или щелкнуть мышью на пиктограмме ;
5. в появившемся окне (рис. 5) ввести имя добавляемого сервера;
6. нажать кнопку ОК для подтверждения операции или Cancel – для отмены;
7. повторить операции для добавления других серверов.

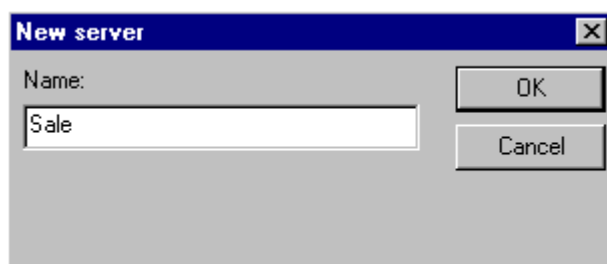



Рис. 5. Диалоговое окно добавление сервера

 Для корректной работы сервера репликации после добавления серверов они должны быть описаны в файле **nodetab**. Порт, указываемый для каждого сервера должен соответствовать порту, указанному при запуске сервера репликации на этом сервере. Протокол такого сервера в файле **nodetab** должен быть ATCPIP (или ATCPIPS в случае использования протокола SSL на UNIX).

Удаление сервера репликации из таблицы серверов


Удаление записей в таблицу **SERVERS** возможно двумя способами:

- с помощью командного интерфейса;
- с помощью утилиты «Рабочий стол ЛИНТЕРА».

Для удаления записей первым способом:

1. запустить на выполнение командный интерфейс (см. стр. 9);
2. в ответ на подсказку утилиты `inl` ввести команду:
`SQL>drop node <имя ЛИНТЕР SQL-сервера>;`
где `<имя ЛИНТЕР SQL-сервера>` – имя удаляемого ЛИНТЕР SQL-сервера, например:
`SQL>drop node sale;`
`SQL>drop node "Market";`
`SQL>drop node "Кадры";`
3. повторить операцию 2 для всех удаляемых из таблицы серверов;
4. завершить выполнение утилиты `inl`:
`SQL>exit`

Для удаления сервера вторым способом:

1. запустить на выполнение утилиту «Рабочий стол ЛИНТЕРА» (см. стр. 17);
2. в окне утилиты «Рабочий стол ЛИНТЕРА» выполнить команду **Replication** ⇒ **Delete server** или нажать клавиши `<Ctrl>+` или щелкнуть мышью на пиктограмме ;
3. в стандартном диалоговом окне о подтверждении удаления сервера нажать кнопку ОК, для отказа – кнопку Cancel;
4. повторить операции 1-3 для удаления других серверов.

Создание правила репликации


Создание правила репликации возможно двумя способами:

- с помощью командного интерфейса;
- с помощью утилиты «Рабочий стол ЛИНТЕРА».



Для создания добавления записей первым способом:

1. запустить на выполнение командный интерфейс (см. стр. 9).
2. в ответ на подсказку утилиты `inl` ввести команду создания правила репликации, например:
`SQL> CREATE REPLICATION RULE repl_table1
FOR table1 TO table1
ON NODE sale
USER Manager;
CREATE REPLICATION RULE repl_table2
FOR table2 TO table2
ON NODE sale
USER Manager
IGNORE OLD VALUE;`

3. завершить выполнение утилиты `inl`:
`SQL>exit`

 Создание правила репликации с помощью SQL-запроса можно выполнить также и средствами утилиты «Рабочий стол ЛИНТЕРА» (см документ «Рабочий стол» пункт Выполнение SQL-запросов).

Для создания правила репликации вторым способом:

1. запустить на выполнение утилиту «Рабочий стол ЛИНТЕРА» (см. стр. 17).
2. в утилите «Рабочий стол ЛИНТЕРА» выбрать пункт меню **Database** ⇒ **Replications**. Появится диалоговое окно для ввода информации в таблицы **SERVERS** и **\$\$\$REPL** (рис. 4);
3. активизировать фрейм таблицы репликаций **\$\$\$REPL** (щелкнуть левой кнопкой мыши в любом месте правого фрейма) В результате должны активизироваться пункт меню **Replication** ⇒ **New relation...** и пиктограмма ;
4. в окне утилиты «Рабочий стол ЛИНТЕРА» выполнить команду **Replication** ⇒ **New replication...** или нажать клавиши `<Ctrl>+<Alt>+<Ins>` или щелкнуть мышью на пиктограмме . Появится диалоговое окно создания правила репликации (рис. 6).

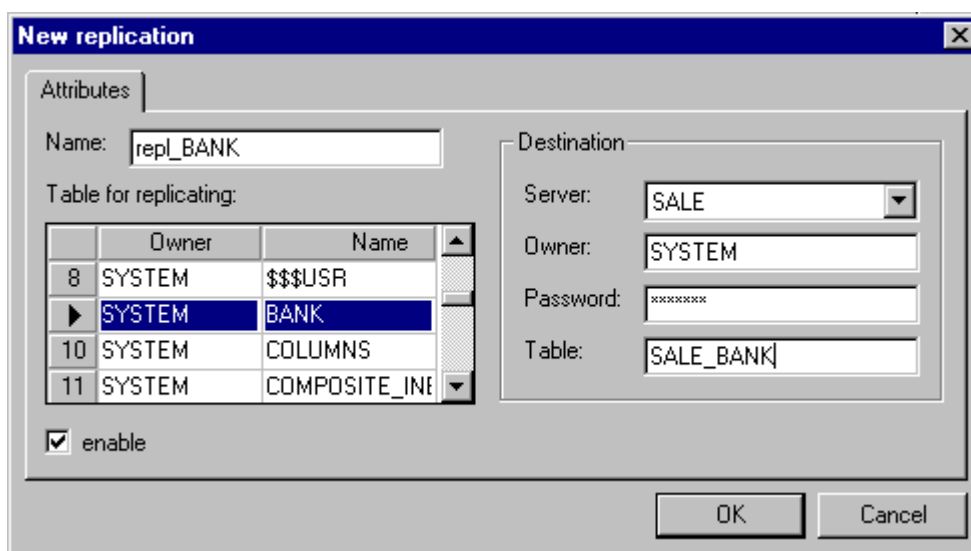



Рис. 6. Диалоговое окно создания правила репликации

5. заполнить поля диалогового окна создания правила репликации.

<u>Поле</u>	<u>Описание</u>
Name	Уникальное в пределах БД-источника имя репликационного правила. Строка длиной до 66 символов
Table for replicating	Реплицируемая таблица. Выбирается из полного списка всех таблиц БД-источника
Destination	Параметры БД-приемника
Server	Имя ЛИНТЕР SQL-сервера БД-приемника
Owner	Имя владельца таблицы-приемника
Password	Пароль владельца таблицы-приемника
Table	Имя таблицы-приемника

<u>Поле</u>	<u>Описание</u>
enable	Флажок состояния правила репликации: установлен – правило должно выполняться, сброшен – правило отменено
6.	нажать кнопку ОК для подтверждения создания правила репликации или Cancel – для отмены.

 При необходимости задания правила репликации с дополнительными опциями (отсутствующими в диалоговом окне) следует использовать командный интерфейс для создания правила.

Модификация правила репликации

Модификация правила репликации возможна двумя способами:


- с помощью командного интерфейса;
- с помощью утилиты «Рабочий стол ЛИНТЕРА».

Для модификации правила репликации первым способом:



1. запустить на выполнение командный интерфейс (см. с. 9);
2. в ответ на подсказку утилиты `inl` ввести команду модификации правила репликации, например:

```
SQL> ALTER REPLICATION RULE repl_table1 DISABLE;
```
3. завершить выполнение утилиты `inl`:

```
SQL>exit
```

 Модификацию правила репликации с помощью SQL-запроса можно выполнить также и средствами утилиты «Рабочий стол ЛИНТЕРА» (см. документ «Рабочий стол» пункт «Выполнение SQL-запросов»).

Для модификации правила репликации вторым способом:

1. запустить на выполнение утилиту «Рабочий стол ЛИНТЕРА» (см. стр. 17);
2. в утилите «Рабочий стол ЛИНТЕРА» выбрать пункт меню **Database ⇒ Replications**. Появится диалоговое окно для ввода информации в таблицы **SERVERS** и **\$\$\$REPL** (рис. 4);
3. активизировать фрейм таблицы репликаций **\$\$\$REPL** (щелкнуть левой кнопкой мыши в любом месте правого фрейма) В результате должны активизироваться пункт меню **Replication ⇒ New relation...** и пиктограмма .
4. в окне утилиты «Рабочий стол ЛИНТЕРА» выполнить команду **Replication ⇒ Properties...** или нажать клавиши `<Alt>+<Enter>` или щелкнуть мышью на пиктограмме  Появится диалоговое окно модификации правила репликации (рис. 7);

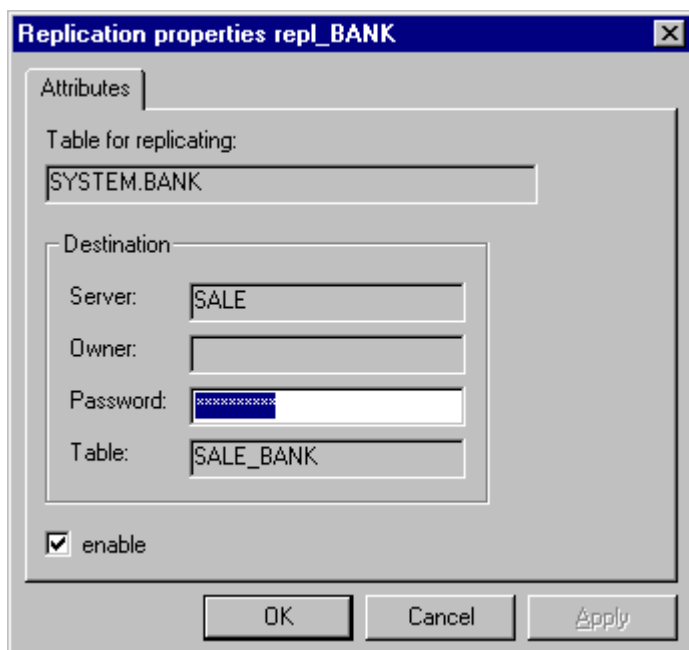


Рис. 7. Диалоговое окно модификации правила репликации

5. заполнить поля диалогового окна модификации правила репликации БД.

<u>Поле</u>	<u>Описание</u>
Password	Пароль владельца таблицы-приемника. Строка длиной не более 18 символов. Проверка корректности введенного значения при модификации правила не производится (неправильно заданное значение выявляется только в момент применения данного правила сервером репликации)
enable	Флажок состояния правила репликации: установлен – правило должно выполняться, сброшен – правило отменено

6. Нажать кнопку Apply , затем OK – для подтверждения выполненной модификации правил репликации и закрытии окна, Cancel – для закрытия окна.

Удаление правила репликации

Удаление правила репликации возможно двумя способами:



- с помощью командного интерфейса;
- с помощью утилиты «Рабочий стол ЛИНТЕРа».

Для удаления правила репликации первым способом:

1. запустить на выполнение командный интерфейс (см. стр. 9).
2. в ответ на подсказку утилиты inl ввести команду удаления правила, например:
SQL> drop replication rule "repl_Rule_bank";

Для удаления правила репликации вторым способом:

1. запустить на выполнение утилиту «Рабочий стол ЛИНТЕРа» (см. стр. 17);
2. в утилите «Рабочий стол ЛИНТЕРа» выбрать пункт меню **Database** ⇒ **Replications**. Появится диалоговое окно для ввода информации в таблицы SERVERS и \$\$\$REPL (рис. 4);

3. активизировать фрейм таблицы репликаций \$\$\$REPL (щелкнуть левой кнопкой мыши в любом месте правого фрейма) В результате должны активизироваться пункт меню **Replication** ⇒ **Delete** и пиктограмма .
4. выделить в фрейме таблицы репликаций правило, которое должно быть удалено;
5. в окне утилиты «Рабочий стол ЛИНТЕРА» выполнить команду **Replication** ⇒ **Delete** или нажать клавиши <Ctrl>+ или щелкнуть мышью на пиктограмме .
6. нажать кнопку ОК в стандартном окне подтверждения операции удаления или Cancel – для отмены удаления.

Просмотр правил репликации



Просмотр правила репликации возможен двумя способами:

- с помощью командного интерфейса;
- с помощью утилиты «Рабочий стол ЛИНТЕРА».

Для просмотра правила репликации первым способом:

1. запустить на выполнение командный интерфейс (см. стр. 9).
2. в ответ на подсказку утилиты in1 ввести команду просмотра правила , например:
`SQL> select * from $$$REPL where rule_name= "Rep1_Rule_Bank";`

Для просмотра правила репликации вторым способом:

1. запустить на выполнение утилиту «Рабочий стол ЛИНТЕРА» (см. стр. 17);
2. активизировать фрейм таблицы репликаций \$\$\$REPL (щелкнуть левой кнопкой мыши в любом месте правого фрейма) В результате должны активизироваться пункт меню **Replication** ⇒ **Properties...** и пиктограмма .
3. выделить в фрейме таблицы репликаций правило, которое должно быть просмотрено;
4. в окне утилиты «Рабочий стол ЛИНТЕРА» выполнить команду **Replication** ⇒ **Properties** или нажать клавиши <Alt>+<Enter> или щелкнуть мышью на пиктограмме .
5. появится диалоговое окно с дополнительными опциями правила репликации;
6. нажать кнопку ОК для завершения просмотра.

Запуск/останов репликационного сервера

Процедура запуска/останова репликационного сервера описана в подразделе «Управление репликационным ЛИНТЕР-сервером» документа «Сетевой администратор».

Конфигурирование репликационного сервера

Процедура конфигурирования репликационного сервера описана в подразделе «Управление репликационным ЛИНТЕР-сервером» документа «Сетевой администратор».

Администрирование репликационного сервера в UNIX

Для настройки асинхронной репликации данных из одной БД в другую необходимо иметь два Unix-сервера, соединяемых между собой посредством ТСП/IP-протокола.

Настройка репликации на вновь создаваемых базах данных

Для установки сервера репликации необходимо получить дистрибутив СУБД ЛИНТЕР с поддержкой асинхронной репликации. Дистрибутив состоит из архива, содержащего исполняемые файлы, примеры, библиотеки и т.д., файлов содержащих текст лицензии `license.txt` и некоторые сведения по установке и запуску СУБД `install.txt`, а также инсталляционного shell-скрипта `install.sh`.

Для установки необходимо запустить программу `install.sh`. При этом текущий каталог должен содержать все файлы дистрибутива (`install.sh`, `license.txt`, `linter.tar.Z`). Процесс распаковки и настройки СУБД для разработчика выполняются обычным образом (стадия настройки для разработчика может быть даже опущена). Далее программа установки выведет запрос на создание скриптов запуска ядра СУБД ЛИНТЕР:

```
Create Linter start procedure script?
```

На этот запрос необходимо ответить Y. Далее Вы должны выбрать будете ли вы использовать уже готовую, поставляемую в дистрибутиве базу данных или создать новую. В случае использования поставляемой базы, будет использована БД из каталога `db` распакованного дерева файлов и каталогов ЛИНТЕР. Если же вы будете создавать новую базу данных, то на запрос

```
Enter the database directory:
```

необходимо ввести абсолютный путь (от /) к каталогу БД. При отсутствии такого каталога в файловой системе Вам будет предложено его создать.



Каталог создается командой `mkdir` и, следовательно, не может быть создано несколько вложенных каталогов.

База данных аналогичная поставляемой в дистрибутиве будет создана в течение нескольких минут.

Если в вашем дистрибутиве ЛИНТЕР присутствует поддержка асинхронной репликации, то вы увидите следующий запрос:

```
Create auxiliary database needed for asynchronous replication  
feature activation?
```

Если вы не желаете использовать асинхронную репликацию, ответьте N, и дальнейшая установка пойдет как обычно.

Для хранения информации об изменившихся данных сервер репликации использует вспомогательную базу данных. Именно эту вспомогательную базу данных и предлагается создать. Ответьте на запрос Y и подождите несколько минут. В подкаталоге `storage` в каталоге основной базы данных будет создана вспомогательная БД.

Администрирование репликационного сервера в UNIX

Далее Вам будет предложено создать shell-программы запуска/останова ядра СУБД ЛИНТЕР:

Create Linter start/stop procedure for the database?

Подтвердите создание программ, введите Y.

Далее следуют запросы о запуске совместно с ядром утилит ЛИНТЕР. Среди них будет и запрос о создании программ запуска с возможностью активации асинхронной репликации:

Do you want to activate asynchronous replication feature?


на который необходимо также ответить утвердительно.

На запрос программы установки необходимо ввести имя удаленного сервера и его IP адрес. Вы можете настроить репликацию данных с несколькими удаленными серверами. Ввод пустого значения в имени сервера свидетельствует, что все удаленные сервера введены, и программа установки может начать настройку БД для репликации. После каждой введенной пары имя-адрес отображается полная таблица удаленных серверов. Рекомендуется для начала настроить репликацию с одним удаленным сервером.

Настройка базы данных и создание программ запуска занимает некоторое время. Далее следуют те же запросы, что и для дистрибутива без репликации.

Результатом работы программы установки должны стать:

- вновь созданная база данных в указанном каталоге;
- вновь созданная вспомогательная база данных в подкаталоге `storage` базы данных;
- скрипты запуска и останова ядра со вспомогательной БД, репликационного сервера и сетевого клиента(агента) в каталоге `bin` дерева дистрибутива `async_start.sh`, `async_stop.sh`;

 Эти программы управляют только запуском ядра для вспомогательной БД, сетевым агентом и сервером репликации.

- для запуска/останова ядра для основной БД совместно со вспомогательной и сервером репликации необходимо использовать программы `start_linter.sh` и `stop_linter.sh` в каталоге `bin`. Они автоматически выполняют программы предыдущего пункта.

Использование поставляемой в дистрибутиве БД

Процедура установки ничем не отличается от описанной выше, кроме выбора существующей БД вместо создания новой. При этом стадия создания БД будет пропущена.

Настройка существующей БД на использование репликации

Эта ситуация возможна, если Вы уже используете СУБД ЛИНТЕР, но без асинхронной репликации. Вами должен быть получен дистрибутив ЛИНТЕР той же версии (в первых

двух числах), но с поддержкой возможности асинхронной репликации. При установке Вы должны отказаться от создания БД и стартовых скриптов. По окончании установки вы должны запустить программу `make_scripts` из корня дерева дистрибутива ЛИНТЕР. На запрос о местоположении базы данных Вы должны ввести путь до каталога Вашей существующей БД. После обычной настройки запуска утилит Вам будет предложено создать вспомогательную БД системы асинхронной репликации, активизировать в скриптах запуска систему репликации. После этого необходимо ввести список удаленных серверов как при инсталляции. Этой же программой (`make_scripts`) вы можете воспользоваться и в случае существования БД с поддержкой асинхронной репликации. Результатом работы программы будут перечисленные выше 4 управляющих программы, создание новых серверов (таблица `SERVERS` основной БД), создание новых записей в файле `nodetab` и, возможно, новая вспомогательная БД. С использованием этой программы создаются скрипты запуска и при установке.

Создание новой базы данных

Для создания новой базы данных можно использовать программу `create_database` в корне дерева дистрибутивов. После запроса каталога будет создана новая БД, аналогичная поставляемой в дистрибутиве. Эта же программа используется для создания БД и при установке. В конце ее выполнения Вам будет предложено создание скриптов запуска (те выполнить `make_scripts`).



Стартовые скрипты рассчитаны на работу только с одной базой данных на каждом сервере. Если необходимо использование нескольких баз данных, то необходим второй комплект стартовых скриптов с изменениями параметров запуска, внесенными вручную. Можно также все операции по старту ядра и утилит проводить вручную (см. ниже).

На этом предварительная настройка на данном сервере может быть завершена. Точно такая же процедура настройки должна быть произведена и на удаленном сервере. При этом в качестве удаленного необходимо использовать адрес сервера, на котором была произведена первая установка.

Необходимо следить, чтобы структура реплицируемых таблиц была одинакова на обеих базах данных. Возможно, удобнее будет сделать одну БД, потом скопировать ее целиком на удаленный сервер и только после этого настроить репликацию. Необходимо также учитывать, что передача данных начнется с момента первого запуска сервера репликации и создании правил репликации специальными запросами. Поэтому на этот момент времени базы данных должны быть идентичны (по крайней мере, реплицируемые таблицы).

После настройки запуска необходимо запустить ядро для основной и вспомогательной БД и утилиты ЛИНТЕР. Запуск всех необходимых программ производится командным файлом `start_linter.sh`, находящемся в каталоге `bin` развернутого дерева дистрибутива.

После этого необходимо создать правила репликации. Правило создается индивидуально для каждой таблицы для каждого удаленного сервера. Та база данных, на которой создается правило репликации, становится источником данных. База данных, на которой не задано правила репликации – становится приемником. Возможен также двусторонний обмен данными.

В простейшем случае, если Вы хотите, чтобы данные таблицы с именем `T1` передавались на удаленную базу `REMOVEDB` в таблицу `T2` запрос на создание правила должен быть следующим:

Администрирование репликационного сервера в UNIX

```
create replication rule RULE_ONE for T1 to T2 on server REMOTEDB;
```

Этот запрос необходимо выполнить, используя утилиту `inl`.

Далее необходимо проверить функционирование системы репликации. Для этого нужно создать одинаковые таблицы в базе-источнике и базе-приемнике, содержащие первичные ключи. Для выполнения запросов можно воспользоваться утилитой `inl`.

```
SQL>create table T1 ( ID int primary key, C char(20));
SQL>create table T2 ( ID int primary key, C char(20));
```

Потом на базе-источнике необходимо создать правило репликации:

```
SQL>create replication rule RULE_ONE for T1 to T2 on server
REMOVEDB;
```

Вставим запись в таблицу в базе-источнике:

```
SQL>insert into T1 values ( 1, "A");
```

Через некоторое время запись должна появиться в базе-приемнике:

```
SQL>select * from T2;
|1 |A |
```

Если у Вас возникли какие либо проблемы с системой репликации, то, прежде всего, необходимо исследовать лог-файл сервера репликации `lreplserver.log`. Он создается в том же каталоге, что и `lreplserver`. Возможно, для выяснения причин проблемы необходимо будет увеличить уровень выводимой отладочной информации, по умолчанию используется уровень 1.

Запуск репликации вручную в UNIX

 Раздел для опытных пользователей.

Ниже перечислены действия, необходимые для запуска системы репликации вручную. Т.к. система репликации симметрична относительно направления передачи данных, поэтому все действия должны производиться как на БД-источнике, так и на БД приемнике, если обратное не оговорено особо.

1. создать основную базу данных. Структура базы должна быть идентична для источника и приемника. На этой базе должен быть выполнен набор запросов из файла `dict/distr.sql`. После этого ядро для этой базы данных должно быть остановлено и запущено заново;
2. необходимо создать вспомогательную базу хранения репликационных данных. Для этого необходимо выполнить скрипт `db/make_storage`. Этот скрипт создает базу данных в текущем каталоге, поэтому перед запуском необходимо сменить текущий каталог на каталог вспомогательной базы данных. Можно также создать базу и вручную. Для этого надо:
 - создать базу утилитой `gendb`;
 - задать идентификатор механизма обмена, отличный от значения по умолчанию. Идентификатор механизма обмена задается путем установки переменной окружения `LINTER_MBX` в какое либо натуральное (в математ. смысле) значение. Значение по умолчанию равно 20561. Если установлено значение

LINTER_MBX, то ядро и приложение для обмена данными будут использовать именно этот идентификатор. Таким образом, возможен запуск нескольких ядер и приложений, работающих с несколькими ядрами;




- запустить ядро для вспомогательной базы данных, выполнить запросы из файла `dict/arepl.sql`, перестартовать ядро. LINTER_MBX должен быть задан во время выполнения этих действий и быть одинаковым;
 - удалить переменную окружения LINTER_MBX.
3. настроить файл конфигурации сетевого агента и сервера репликации `nodetab`. Он должен располагаться в каталоге `bin` дерева дистрибутива. Для настройки надо выполнить следующие действия:
- добавить строку `RDSTG local 12345`, где `RDSTG` – условное имя вспомогательной БД на данной машине, `local` – ключевое слово, `12345` идентификатор механизма обмена (см. выше);
 - добавить строку `REMOTEDB ATCPIP my_name.my_domain 1080`, где `REMOTEDB` – условное имя удаленной основной базы данных. `ATCPIP` – ключевое слово для задания TCP/IP протокола обмена, `my_name.my_domain` – адрес удаленного сервера, `1080` – порт удаленного сервера. Если база данных будет только приемником данных, данный подпункт необязателен.
4. объявить новый сервер, для этого запустить утилиту `INL` и набрать в командной строке:
- ```
sql> create server REMOTEDB;
```
5. запустить сетевого агента `dbc_tcp`, возможно указав ему в командной строке точное расположение `nodetab`;
6. запустить сервер репликации `lreplserver`. В аргументах командной строки необходимо передать имя администратора-пароль основной и вспомогательной базы данных, условное имя вспомогательной БД (`RDSTG`) из файла `nodetab`, путь к файлу `nodetab`, номер порта на котором сервер будет принимать удаленные соединения (номера портов для установления соединения задаются в `nodetab`), `-daemon` – для работы в фоновом режиме, уровень трассировки;
7. для основной базы-источника создать правила репликации необходимых данных (`create replication rule...`);
8. проверить работу системы репликации, изменив данные одной из реплицируемых таблиц базы-источника, и проверив их изменение в базе приемнике;
9. останов производится в обратном порядке:
- остановить `lreplserver` (опция `-down`);
  - остановить `dbc_tcp` (командой `kill`, `pid` процесса можно определить командой `ps`, или при старте указать на необходимость создания файла со значением `pid`);
  - остановить работу вспомогательной БД утилитой `shut`. Для останова вспомогательной БД необходимо указать тот же идентификатор обмена (LINTER\_MBX), что и при старте. После останова БД переменную окружения LINTER\_MBX необходимо удалить;
  - остановить работу ядра с основной базой данных.

# Настройка репликации вручную

 Раздел для опытных пользователей.

После установки сервера репликации может понадобиться настроить и запустить его вручную. Сервер репликации должен быть установлен на двух серверах: источнике (И) и приемнике (П). Предположим, что нам нужно реплицировать таблицу **ТАВ** с источника на приемник, используя порт на приемнике 1081. TCP/IP адрес приемника будет test.server.ru.

Последовательность шагов:

| Шаг | Сервер | Действие                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-----|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | И      | Установить сервер репликации из дистрибутива                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| 2   | П      | Установить сервер репликации из дистрибутива                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| 3   | И      | Запустить БД и прогнать файл <code>distr.sql</code> – с. 9. (Должны быть созданы системные таблицы <code>\$\$\$REPL</code> и <code>SERVERS</code> ).                                                                                                                                                                                                                                                                                                                                                       |
| 4   | И      | Перезапустить БД                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 5   | П      | Запустить БД и прогнать файл <code>distr.sql</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 6   | П      | Перезапустить БД                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 7   | И      | Создать новый сервер (например <code>SRV</code> ) командой SQL (с. 17):<br><code>SQL&gt; Create server SRV;</code>                                                                                                                                                                                                                                                                                                                                                                                         |
| 8   | И      | Дописать в <code>nodetab</code> строку с адресом нового сервера:<br><code>SRV ATCPIP test.server.ru 1081</code><br>Здесь <code>1081</code> – номер порта принимающей стороны.<br> Для серверов, на которые будет осуществляться репликация, в файле <code>nodetab</code> должны быть прописаны строки с протоколом <code>ATCPIP</code> (или <code>ATCPIPS</code> в UNIX при использовании протокола <code>SSL</code> ). |
| 9   | И      | Запустить БД хранилища. Местоположение БД хранилища указывается при установке сервера репликации. Это обычная БД <code>ЛИНТЕР</code> , ее запуск осуществляется стандартным образом. При запуске необходимо указать имя базы данных <code>RDSTG</code> , например, для NT:<br><code>Linter /local /name=RDSTG</code>                                                                                                                                                                                       |
| 10  | И      | Запустить клиент <code>dbc_tcp</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 11  | П      | Запустить БД хранилища аналогично источнику                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 12  | И      | Запустить сервер репликации, как описано на стр. 6<br>В NT рекомендуется запускать с ключом <code>-d</code> , при этом создается консоль, и на нее выводятся вся трассировка.<br> При первом запуске уровень трассировки желательно задавать не меньше 3, чтобы легче разрешить возможные проблемы.                                                                                                                     |
| 12  | П      | Запустить сервер репликации, как описано на с. 6<br>Убедиться, что порт приема (задается через ключ <code>-p</code> ) совпадает с прописанным в <code>nodetab</code> (в нашем случае 1081).<br>В NT рекомендуется запускать с ключом <code>-d</code> , при этом создается консоль, и на нее выводятся вся трассировка.<br> При первом запуске уровень трассировки желательно задавать                                   |

| <u>Шаг</u> | <u>Сервер</u> | <u>Действие</u>                                                                                          |
|------------|---------------|----------------------------------------------------------------------------------------------------------|
|            |               | не меньше 3, чтобы легче разрешить возможные проблемы.                                                   |
| 13         | И             | Создать таблицу для репликации с первичным ключом (или убедиться, что он есть), например таблица TAB     |
| 14         | П             | Создать таблицу для приема, эквивалентную той, которая на источнике                                      |
| 15         | И             | Создать правило репликации с помощью <code>inl</code> , или «Рабочий стол ЛИНТЕРа», как описано на с. 11 |
| 16         | И             | Добавить запись в таблицу TAB                                                                            |
| 17         | П             | Убедиться, что такая же запись добавилась и на сервере <code>test.server.ru</code>                       |

# Файл трассировки

## Формирование имени файла

Имя файла трассировки обычно – `lreplserver.log`. Однако, это имя используется только в том случае, если сервер репликации запускается для базы данных по умолчанию и не разбивается на тома. Если при старте сервера репликации указано имя базы данных, для которой он запускается (ключ `-l`), то оно будет использовано для образования имени файла трассировки. Для базы данных DEMO имя файла будет `lreplserver(DEMO).log`.

В случае разбиения файла трассировки на тома (ключ `-lvol`) к имени добавится расширение с номером тома, начиная с 0, например: `lreplserver(DEMO).log.0` или `lreplserver(DEMO).log.1`. Номера томов будут последовательно увеличиваться.

## Содержание файла трассировки

При работе сервера репликации может создаваться файл трассировки, который, в зависимости от выбранного уровня трассировки, содержит информацию о работе сервера репликации. Уровень трассировки задается с помощью ключа `-log <число>`, где `<число>` – уровень трассировки:

| <u>Число</u> | <u>Описание</u>                                                             |
|--------------|-----------------------------------------------------------------------------|
| 0            | Нет трассировки, файл не создается                                          |
| 1            | Минимальная трассировка, выдаются основные сообщения и сообщения об ошибках |
| 2            | Выдаются сообщения о предупреждениях                                        |
| 3            | Выдаются информационные сообщения                                           |
| 4            | Подробная трассировка                                                       |

При перезапуске сервера репликации файл не удаляется, запись происходит в конец файла.

При запуске сервера репликации в файл пишется заголовок (если трассировка разрешена):

```
----- Mon Oct 28 13:21:57 2002 -----
```

В заголовке содержится дата и время старта сервера репликации.

Далее идут информационные сообщения, каждое из которых начинается с префикса, включающего в себя дату и время записи в формате:

```
Mon.dd hh24:mi:ss.tt
```

затем номер процесса (pid) и номер нити (thread), которая произвела запись. В UNIX номер нити равен номеру процесса, вот часть типичного файла трассировки:

```
----- Mon Oct 28 13:21:57 2002 -----
Oct.28 13:21:57.36,p1068,t104: start replication server
Oct.28 13:21:57.36,p1068,t104: Trace level 4
Oct.28 13:21:57.36,p1068,t104: service flag is: 0
Oct.28 13:21:57.40,p1068,t104: Use nodetab file
'Z:\Linter\bin\nodetab'
Oct.28 13:21:57.90,p1068,t104: Begin delete old data
```

```

Oct.28 13:21:57.96,p1068,t104: End delete old data
Oct.28 13:21:57.96,p1068,t104: Begin download replication rules
Oct.28 13:21:58.22,p1068,t104: End download replication rules - 3 OK
Oct.28 13:21:58.27,p1068,t104: Begin download replication tables
Oct.28 13:21:58.58,p1068,t104: End download replication tables - 3
OK
Oct.28 13:21:58.61,p1068,t1012: Start main RCV process
Oct.28 13:21:58.68,p1068,t1044: Start SND process for node 'ORACLE'
Oct.28 13:21:58.68,p1068,t992: Start SND process for node 'SERV100'
Oct.28 13:21:58.68,p1068,t104: Open special channel (Log addr
355:213034)
Oct.28 13:21:58.69,p1068,t1056: start find and recovery accumulated
records.
Oct.28 13:21:58.70,p1068,t1044: Unknown server name: "ORACLE"
Oct.28 13:21:58.70,p1068,t1044: error sending to network: 10057
Oct.28 13:21:58.75,p1068,t1056: End recovery accumulated records - 0
found and recovering now.
Oct.28 13:21:59.19,p1068,t104: Current Lreplserver pid=1068
Oct.28 13:21:59.19,p1068,t104: wait data mode started

```

Сообщения файла трассировки:

| Сообщение                                   | Описание                                                                                                                                                        |
|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start replication server                    | Старт                                                                                                                                                           |
| Trace level 4                               | Уровень трассировки – 4                                                                                                                                         |
| Service flag is: 0                          | Сервер репликации запускается не как сервис (для NT). Если запускается как сервис – будет указано значение 1                                                    |
| Use nodetab file<br>'Z:\Linter\bin\nodetab' | Указано, какой конкретно файл с описанием серверов будет использоваться                                                                                         |
| Begin delete old data                       | Начался процесс очистки служебных таблиц от устаревшей информации                                                                                               |
| End delete old data                         | Процесс закончен                                                                                                                                                |
| Begin download<br>replication rules         | Начался процесс скачивания информации о правилах репликации                                                                                                     |
| End download replication<br>rules - 3 OK    | Процесс закончен, найдено три правила                                                                                                                           |
| Begin download<br>replication tables        | Начался процесс скачивания информации о структуре таблиц, предназначенных к репликации                                                                          |
| End download replication<br>tables - 3 OK   | Процесс закончен, найдено три таблицы                                                                                                                           |
| Start main RCV process                      | Запускается процесс приема данных. В префиксе указан id процесса и номер нити (p1068, t1012)                                                                    |
| Start SND process for<br>node 'ORACLE'      | Запускается процесс передачи данных. В префиксе указан id процесса и номер нити (p1068, t1044). Этот процесс будет производить рассылку данных на сервер ORACLE |

### Сообщение

Start SND process for  
node 'SERV100'

Open special channel  
(Log addr 355:213034)

Start find and recovery  
accumulated records.

Unknown server name:  
"ORACLE"

error sending to  
network: 10057

End recovery accumulated  
records - 0 found and  
recovering now.

Current Lreplserver  
pid=1068

wait data mode started

### Описание

Запускается процесс передачи данных. В префиксе указан id процесса и номер нити (p1068, t992). Этот процесс будет производить рассылку данных на сервер SERV100. Всего запущено два процесса рассылки, значит три правила репликации созданы для рассылки на два сервера

Открывается интерфейс к БД-источнику данных. Адрес начала потока данных в системном журнале – файл 355 смещение в файле 213034

Запускается процесс получения данных, измененных во время того, как сервер репликации был остановлен

Сообщение процесса рассылки с номером нити 1044 (это видно из префикса). Сервер ORACLE не найден в файле nodetab, эту ошибку должен устранить администратор

Тот же процесс сообщает об ошибке при попытке отослать данные

Закончен процесс получения данных, измененных во время того, как сервер репликации был остановлен, таких данных не обнаружено

ID процесса репликации – 1068

Начало ожидания изменения данных.

При изменении данных в таблицах, предназначенных для репликации, может появиться подобный фрагмент:

```
Oct.28 13:22:26.26,p1068,t104: Insert record (Tr=302900)
Oct.28 13:22:26.26,p1068,t104: Insert complete
Oct.28 13:22:26.27,p1068,t104: Commit (Tr=302900)
Oct.28 13:22:26.27,p1068,t104: Commit complete
Oct.28 13:22:26.27,p1068,t104: Insert record (Tr=302901)
Oct.28 13:22:26.27,p1068,t104: Insert complete
Oct.28 13:22:26.27,p1068,t104: Commit (Tr=302901)
Oct.28 13:22:26.27,p1068,t104: Commit complete
Oct.28 13:22:26.27,p1068,t104: Insert record (Tr=302902)
Oct.28 13:22:26.27,p1068,t104: Insert complete
Oct.28 13:22:26.27,p1068,t104: Commit (Tr=302902)
Oct.28 13:22:26.27,p1068,t104: Commit complete
Oct.28 13:22:26.27,p1068,t104: Start appends 6 records
```

Здесь в базу данных были вставлены три записи, при этом было три транзакции (302900, 302901, 302902), в последней строке данные заносятся в служебные таблицы, которые использует процесс рассылки данных.

На приемной стороне в файле трассировки может появиться следующее:

```
Oct.28 13:22:29.09,p776,t1256: Processing transaction 302900
(pid=1256)
```

```
Oct.28 13:22:29.22,p776,t1256: Attention! (err#903): cb1[
,DEMO1,3], "INSERT INTO "T1" VALUES(10,'10.10.2000:20:20:20.20');"
Oct.28 13:22:29.23,p776,t1256: Replication error: -- Error 903
```

При выполнении транзакции 302900 произошла ошибка и выдана следующая информация:

- (err#903) – ЛИНТЕР вернул код завершения 903 (повторное значение для первичного или уникального ключа);
- cb1[ ,DEMO1,3] – ошибка произошла при выполнении команды CALL интерфейса ЛИНТЕРа типа ‘ ’ по каналу номер 3, открытому к базе данных DEMO1;
- "INSERT INTO "T1" VALUES(10,'10.10.2000:20:20:20.20');" – SQL-текст команды.

Здесь оказалось, что в таблице T1 уже есть запись с таким первичным ключом. Эта коллизия должна разрешаться администратором.

# Структура базы данных репликации

Сервер репликации хранит данные, предназначенные для реплицирования в таблицах базы данных репликации. Информация, находящаяся в этих таблицах, может быть использована администратором для более полного представления о динамике процесса репликации, особенно в случае возникновения различного рода ошибок, когда трудно понять, в чем именно заключается проблема.

SQL файл для создания нижеописанных таблиц находится в каталоге `dict` установленного ЛИНТЕР сервера и называется `arepl.sql`.

## Очередь рассылки

Очередь рассылки (таблица `DISPATCH_QUEUE`) используется для хранения измененных записей базы данных, которые предназначены для репликации. Эта таблица заполняется на источнике данных. Каждая запись в таблице содержит информацию об одной операции с данными, плюс эти данные.

Структура таблицы:

| <u>Поле</u>              | <u>Тип</u>            | <u>Описание</u>                                                                                                                                                                                                                                                                                                                            |
|--------------------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>OPER_DATE</code>   | <code>BYTE(6)</code>  | Содержит дату операции. Первые 4 байта поля представляют собой количество секунд, прошедших с 1990 г, остальные 2 байта – 65536-е доли секунды. Эти значения хранятся в прямом формате: старшие байты содержат старшую часть слова                                                                                                         |
| <code>EXTDATA</code>     | <code>SMALLINT</code> | Признак продолжения записи, если она не уместилась в 3400 байта поля <code>DATA</code> . Для первой части данных это поле заполнено нулем, для второй – единицей и т.д. Т.е., одна операция может описываться двумя и даже тремя записями в таблице <code>DISPATCH_QUEUE</code> , если реплицируемая таблица имеет записи большого размера |
| <code>DBNODE</code>      | <code>CHAR(8)</code>  | Имя базы данных, из которой эти данные получены                                                                                                                                                                                                                                                                                            |
| <code>TRANSACT_ID</code> | <code>BIGINT</code>   | Номер транзакции, которой принадлежит эта операция                                                                                                                                                                                                                                                                                         |
| <code>SYSLOG_ADDR</code> | <code>BIGINT</code>   | Адрес в системном журнале, по которому находится эта запись. Старшие 4 байта содержат номер файла, младшие – адрес в файле                                                                                                                                                                                                                 |
| <code>TYP</code>         | <code>SMALLINT</code> | Тип операции:<br>1 – <code>DELETE</code> ;<br>2 – <code>INSERT</code> ;<br>3 – <code>UPDATE</code> ;<br>7 – <code>CREATE BLOB</code> ;<br>8 – <code>INSERT BLOB</code> ;<br>9 – <code>CLEAR BLOB</code> ;<br>4 – <code>COMMIT</code> ;                                                                                                     |

| <u>Поле</u> | <u>Тип</u>  | <u>Описание</u>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             |             | 10 – ROLLBACK.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| BLOBATR     | SMALLINT    | Используется для BLOB операций. Младший байт содержит номер атрибута в таблице, которому принадлежит BLOB, старший – тип BLOB'а                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| CHANNEL     | INT         | Номер канала, который произвел операцию                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| TABNUM_SRC  | INT         | Номер реплицируемой таблицы                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| ROWID_SRC   | INT         | Идентификатор строки в исходной таблице                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| LEN1        | INT         | Длина первой части буфера данных (в поле DATA)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| LEN2        | INT         | Длина второй части буфера данных (в поле DATA)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| DST_SERVER  | CHAR(8)     | Имя сервера назначения                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| DST_OWNER   | CHAR(66)    | Имя владельца таблицы-приемника данных                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| DST_TABLE   | CHAR(66)    | Имя таблицы-приемника данных                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| DATA        | BYTE (3400) | Буфер с данными. Условно разделен на две части с длинами len1 и len2.<br>Тип операции:<br>1. DELETE. Первая часть содержит старую запись, которая удаляется. Вторая часть пуста;<br>2. INSERT. Первая часть содержит новую запись, которая вставляется. Вторая часть пуста;<br>3. UPDATE. Первая часть содержит старую запись, которая замещается. Вторая часть содержит новую запись, которая замещает старую;<br>4. DELETE BLOB. Первая часть содержит запись, которой принадлежит удаляемый BLOB. Вторая часть пуста;<br>5. COMMIT, ROLLBACK. Первая и вторая части пусты |

## Состояние потока рассылки

Состояние потока рассылки (таблица DISPATCH\_STREAM) содержит информацию о состоянии рассылки на каждый из серверов. Каждой паре (БД источник – БД приемник) соответствует одна запись в этой таблице.

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                                                                                                                                                                                                                                                                                        |
|-------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OPER_DATE   | BYTE(6)    | Содержит дату операции. Первые 4 байта поля представляют собой количество секунд, прошедших с 1990 г., остальные 2 байта – 65536-е доли секунды. Эти значения хранятся в прямом формате: старшие байты содержат старшую часть слова. Эта дата соответствует дате последней записи из очереди рассылки, |

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                                                                        |
|-------------|------------|----------------------------------------------------------------------------------------|
| DBNODE      | CHAR(8)    | отправленной на приемный сервер<br>Имя базы данных, из которой производится репликация |
| SERVER_DEST | CHAR(8)    | Имя сервера, на который идет репликация                                                |

## История рассылки

История рассылки (таблица DISPATCH\_QUEUE\_HIST) используется для архивного хранения репликационных данных, которые уже были использованы для репликации. Ее структура аналогична структуре таблицы DISPATCH\_QUEUE. Если при запуске сервера репликации задан ключ `-c1ear <число>`, то процесс очистки периодически будет перегружать данные из DISPATCH\_QUEUE в DISPATCH\_QUEUE\_HIST. Данные из таблицы DISPATCH\_QUEUE\_HIST можно удалять, если они не нужны.

## Приемная очередь

Приемная очередь (таблица RECEIVING\_QUEUE) используется для хранения принятых данных, которые пришли с передающего сервера. Эта таблица заполняется на приемнике данных. Каждая запись в таблице содержит информацию об одной операции с данными, плюс эти данные. Структура таблицы похожа на структуру DISPATCH\_QUEUE:

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                                                                                                                                                                                                                                                                                                |
|-------------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OPER_DATE   | BYTE(6)    | Содержит дату операции. Первые 4 байта поля представляют собой количество секунд, прошедших с 1990 г, остальные 2 байта – 65536-е доли секунды. Эти значения хранятся в прямом формате: старшие байты содержат старшую часть слова                                                                             |
| REPL_DATE   | BYTE(6)    | Содержит дату пересылки записи. Формат записи аналогичен OPER_DATE                                                                                                                                                                                                                                             |
| EXTDATA     | SMALLINT   | Признак продолжения записи, если она не уместилась в 3400 байта поля DATA. Для первой части данных это поле заполнено нулем, для второй – единицей и т.д. Т.е., одна операция может описываться двумя и даже тремя записями в таблице DISPATCH_QUEUE, если реплицируемая таблица имеет записи большого размера |
| RESERVE     | SMALLINT   | Не используется                                                                                                                                                                                                                                                                                                |
| DBNODE      | CHAR(8)    | Имя базы данных, из которой эти данные получены                                                                                                                                                                                                                                                                |
| TRANSACT_ID | BIGINT     | Номер транзакции, которой принадлежит эта операция.                                                                                                                                                                                                                                                            |
| TYP         | SMALLINT   | Тип операции:<br>1 – DELETE;                                                                                                                                                                                                                                                                                   |

| Поле          | Тип         | Описание                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               |             | 2 – INSERT;<br>3 – UPDATE;<br>7 – CREATE BLOB;<br>8 – INSERT BLOB;<br>9 – CLEAR BLOB;<br>4 – COMMIT;<br>10 – ROLLBACK                                                                                                                                                                                                                                                                                           |
| BLOBATR       | SMALLINT    | Используется для BLOB операций. Младший байт содержит номер атрибута в таблице, которому принадлежит BLOB, старший – тип BLOB’а                                                                                                                                                                                                                                                                                 |
| RETCODE_LIN   | INT         | Код завершения, который вернул ЛИНТЕР при выполнении этой операции                                                                                                                                                                                                                                                                                                                                              |
| RETCODE_SYS   | INT         | Код системной ошибки, который вернул ЛИНТЕР при выполнении этой операции, заполняется, если RETCODE_LIN не равен 0                                                                                                                                                                                                                                                                                              |
| TABNUM_SRC    | INT         | Номер реплицируемой таблицы                                                                                                                                                                                                                                                                                                                                                                                     |
| ROWID_SRC     | INT         | Идентификатор строки в исходной таблице                                                                                                                                                                                                                                                                                                                                                                         |
| LEN1          | INT         | Длина первой части буфера данных (в поле DATA)                                                                                                                                                                                                                                                                                                                                                                  |
| LEN2          | INT         | Длина второй части буфера данных (в поле DATA)                                                                                                                                                                                                                                                                                                                                                                  |
| DST_SERVER    | CHAR(8)     | Имя сервера назначения                                                                                                                                                                                                                                                                                                                                                                                          |
| SRC_SERVER    | CHAR(8)     | Имя сервера, с которого пришли данные                                                                                                                                                                                                                                                                                                                                                                           |
| TABNAME_SRC   | CHAR(66)    | Имя таблицы, из которой пришли данные                                                                                                                                                                                                                                                                                                                                                                           |
| USER_SRC      | CHAR(66)    | Имя владельца таблицы, из которой пришли данные                                                                                                                                                                                                                                                                                                                                                                 |
| TABNAME_DEST  | CHAR(66)    | Имя таблицы-приемника данных                                                                                                                                                                                                                                                                                                                                                                                    |
| USER_DEST     | CHAR(66)    | Имя владельца таблицы-приемника данных                                                                                                                                                                                                                                                                                                                                                                          |
| REPLICAT_NAME | CHAR(66)    | Имя правила репликации, по которому пришла эта запись                                                                                                                                                                                                                                                                                                                                                           |
| DATA          | BYTE (3400) | Буфер с данными. Условно разделен на две части с длинами len1 и len2.<br>Тип операции:<br>1. DELETE. Первая часть содержит старую запись, которая удаляется. Вторая часть пуста;<br>2. INSERT. Первая часть содержит новую запись, которая вставляется. Вторая часть пуста;<br>3. UPDATE. Первая часть содержит старую запись, которая замещается. Вторая часть содержит новую запись, которая замещает старую; |

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                                                                             |
|-------------|------------|---------------------------------------------------------------------------------------------|
|             |            | 4. INSERT, CREATE BLOB. Первая часть содержит добавляемую часть BLOB'а. Вторая часть пуста; |
|             |            | 5. COMMIT, ROLLBACK. Первая и вторая части пусты                                            |

## История приема

История приема (таблица RECEIVING\_QUEUE\_HIST) используется для архивного хранения репликационных данных, которые уже были использованы для репликации. Ее структура аналогична структуре таблицы RECEIVING\_QUEUE. Если при запуске сервера репликации задан ключ `-clear <число>`, то процесс очистки периодически будет перегружать данные из RECEIVING\_QUEUE в RECEIVING\_QUEUE\_HIST. Данные из таблицы RECEIVING\_QUEUE\_HIST можно удалять, если они не нужны.

## Список реплицируемых таблиц

Список реплицируемых объектов (таблица DISPATCH\_TABS) содержит список таблиц, предназначенных для репликации. Заполняется на источнике данных.

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>             |
|-------------|------------|-----------------------------|
| TABNAME     | CHAR(66)   | Имя таблицы                 |
| RESERVE     | SMALLINT   | Не используется             |
| TABNUM      | INT        | Номер таблицы в базе данных |

## Столбцы-источники реплицируемых таблиц

Список столбцов-источников реплицируемых объектов (таблица DISPATCH\_ATTRS) содержит список столбцов таблиц, предназначенных для репликации. Заполняется на источнике данных.

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                                   |
|-------------|------------|---------------------------------------------------|
| TABNUM_SRC  | INT        | Номер таблицы в базе данных                       |
| NAME        | CHAR(66)   | Имя столбца                                       |
| ATRNUM      | SMALLINT   | Номер столбца в таблице                           |
| TYP         | SMALLINT   | Тип столбца                                       |
| PKEY        | SMALLINT   | Признак того, что столбец входит в первичный ключ |
| LEN         | INT        | Размерность столбца (в байтах)                    |

## Столбцы-приемники реплицируемых таблиц

Список столбцов-приемников реплицируемых объектов (таблица RECEIVING\_ATTRS) содержит список столбцов таблиц, предназначенных для

репликации. Заполняется на приемнике данных в момент, когда процесс выполнения транзакций требует описание таблицы, записи из которой нужно занести в целевую БД. Данные запрашиваются из источника.

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                                   |
|-------------|------------|---------------------------------------------------|
| TABNUM_SRC  | INT        | Номер таблицы в базе данных                       |
| NAME        | CHAR(66)   | Имя столбца                                       |
| ATRNUM      | SMALLINT   | Номер столбца в таблице                           |
| TYP         | SMALLINT   | Тип столбца                                       |
| PKEY        | SMALLINT   | Признак того, что столбец входит в первичный ключ |
| LEN         | INT        | Размерность столбца (в байтах)                    |
| SRC_SERVER  | CHAR(8)    | Имя сервера, на котором находится таблица         |

### Список активных правил репликации

Список правил репликации (таблица REPL\_RULES) содержит список правил репликации, активных на сервере. Заполняется на источнике данных.

| <u>Поле</u> | <u>Тип</u> | <u>Описание</u>                        |
|-------------|------------|----------------------------------------|
| REPL_ID     | INT        | Номер правила репликации в базе данных |
| TYP         | SMALLINT   | Тип правила                            |
| REPL_NAME   | CHAR(66)   | Имя правила                            |
| DST_SERVER  | CHAR(8)    | Сервер назначения                      |
| SRC_TABNUM  | INT        | Номер таблицы в базе данных            |
| SRC_OWNER   | CHAR(66)   | Имя владельца таблицы                  |
| SRC_TABLE   | CHAR(66)   | Имя таблицы                            |
| DST_OWNER   | CHAR(66)   | Имя владельца целевой таблицы          |
| DST_TABLE   | CHAR(66)   | Имя целевой таблицы                    |

# Использование таблиц базы данных репликации и файла трассировки

Таблицы базы данных репликации могут быть использованы при поиске ошибок, которые могут возникнуть в результате репликации. Также, их можно использовать для сбора статистики. Получать эти данные можно с помощью программы `inl`, которую следует запускать для базы данных RDSTG, предварительно запустив сервер репликации (стр. 6):

```
inl -u SYSTEM/MANAGER -n RDSTG
```

После этого можно подавать SQL- запросы, например:

| <u>Цель</u>                                                  | <u>SQL- запрос</u>                                                                                                                                                                                                                                                                                                                                |
|--------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Получить список реплицируемых таблиц                         | <pre>select SRC_TABLE from REPL_RULES<br/>group by SRC_TABLE;</pre>                                                                                                                                                                                                                                                                               |
| Получить список серверов, на которые идет репликация         | <pre>select DST_SERVER from REPL_RULES<br/>group by DST_SERVER;</pre>                                                                                                                                                                                                                                                                             |
| Получить список пользователей, таблицы которых реплицируются | <pre>select SRC_OWNER from REPL_RULES<br/>group by SRC_OWNER;</pre>                                                                                                                                                                                                                                                                               |
| Очистить хранилище от старых данных                          | <pre>delete from dispatch_queue_hist;<br/>delete from receiving_queue_hist;</pre>                                                                                                                                                                                                                                                                 |
| Получить дату последней отсылки на сервер SERV               | <pre>select oper_date from<br/>dispatch_stream where<br/>server_dest='SERV';</pre> <p>Ответ записывается в буфер размером 6 байт. Первые 4 байта поля представляют собой количество секунд, прошедших с 1990 г., остальные 2 байта – 65536-е доли секунды. Эти значения хранятся в прямом формате: старшие байты содержат старшую часть слова</p> |
| Определение транзакций, закончившихся с ошибками             | <pre>select transact_id from<br/>receiving_queue where<br/>retcode_lin != 0 group by<br/>transact_id;</pre>                                                                                                                                                                                                                                       |
| Определение кодов ошибок для не прошедших транзакций         | <pre>select transact_id, retcode_lin,<br/>retcode_sys from receiving_queue<br/>where retcode_lin != 0;</pre>                                                                                                                                                                                                                                      |

# Типичные схемы репликации

## Простая репликация с сервера s1 на сервер s2

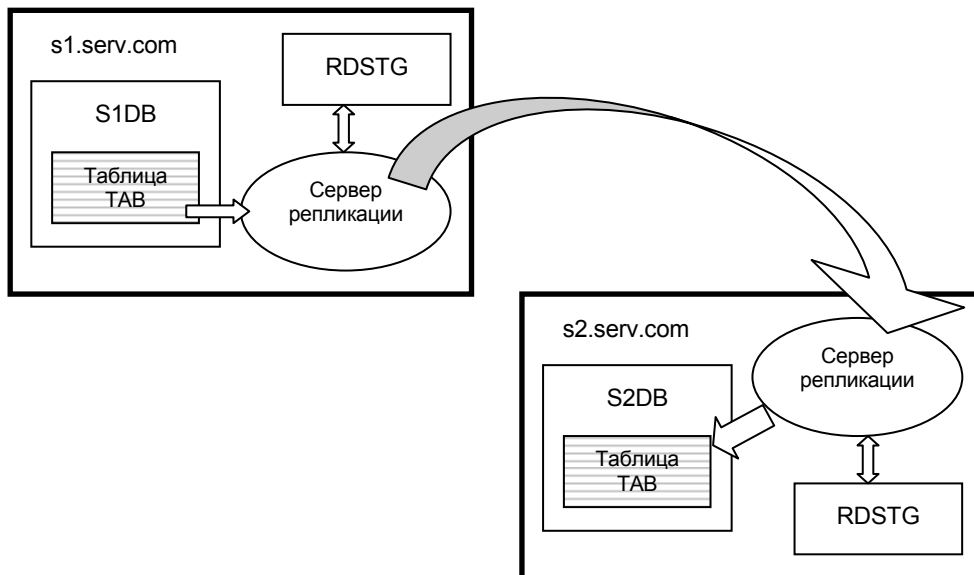


Рис. 8. Схема простой репликации с сервера s1 на сервер s2

### Начальные данные

Имеется два сервера с IP адресами s1.serv.com и s2.serv.com. На каждом из них установлен ЛИНТЕР с поддержкой репликации. На каждом имеется таблица TAB с одинаковой структурой данных. Имена баз данных соответственно S1DB и S2DB.

### Задача

Требуется настроить репликацию таблицы TAB с сервера s1 на сервер s2.

### Последовательность операций

#### На сервере s1

1. текстовым редактором открыть файл `nodetab`, находящийся в каталоге `bin` установленного ЛИНТЕРа и добавить строку:  
`S2 ATCPIP s2.serv.com 1080`

#### Только для NT, Windows 2000

На сервере s1:

1. запустить утилиту «Администратор ЛИНТЕРа» (`linadm`);
2. выбрать базу данных S1DB и выполнить пункт меню **Database** ⇒ **Startup**. Запустить БД;
3. выбрать пункт меню **Network** ⇒ **Replication server** ⇒ **startup**. В появившемся диалоговом окне набрать имя и пароль к базе данных S1DB и хранилищу. Если пароли не переопределялись, то имена и пароли будут стандартными: SYSTEM/MANAGER. После нажатия ОК автоматически стартует БД хранилища;
4. повторить операции на сервере s2 для БД S2DB.

#### Только для UNIX

1. запустить БД S1DB на сервере s1;
2. убедиться, что в файле `nodetab` есть строка типа:  
`RDSTG LOCAL 100`  
Вместо числа 100 может быть другое число.
3. установить переменную окружения `LINTER_MBX`, для этого

выполнить команду командного интерфейса. `export LINTER_MBX=100`

Или другое число, указанное в `nodetab`;

- установить переменную окружения `SY00`, для этого выполнить команду командного интерфейса `export SY00=~/linter/db/storage`  
Здесь должен быть указан путь к БД хранилища. Обычно это каталог `db/storage` в каталоге, куда установлен ЛИНТЕР;
  - запустить БД хранилища;
  - запустить сетевого клиента командой `dbc_tcp`.
  - запустить сервер репликации командой:  
`lreplserver -l S1DB -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -d -p 1080`
  - повторить операции на сервере `s2` для базы данных `S2DB`.
- На сервере s1**
- запустить утилиту `inl`;
  - выполнить команду:  
`Create server s2;`
  - создать правило репликации:  
`create replication rule tabrule for tab to tab on node S2 async;`
  - занести запись в таблицу `TAB`.

**На сервере s2**

- запустить утилиту `inl`;
- убедиться, что запись, занесенная в таблицу `TAB` на сервере `s1` появилась в таблице `TAB` на сервере `s2`.

## Репликация с сервера s1 на сервера s2, s3, s4

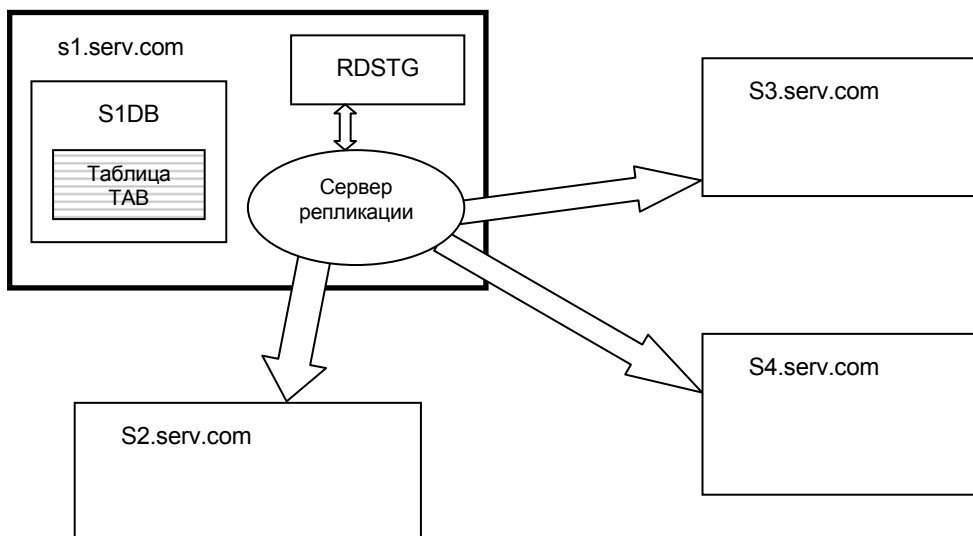


Рис. 9. Схема репликации с сервера `s1` на сервера `s2`, `s3`, `s4`

### Начальные данные

Имеется четыре сервера с IP адресами `s1.serv.com`, `s2.serv.com`, `s3.serv.com`, `s4.serv.com`. На каждом из них установлен ЛИНТЕР с поддержкой репликации. На каждом имеется таблица `TAB` с одинаковой структурой данных. Имена баз данных соответственно `S1DB`, `S2DB`, `S3DB`, `S4DB`.

**Задача**

Требуется настроить репликацию таблицы TAB с сервера s1 на сервера s2, s3, s4.

**Последовательность операций**

**На сервере s1**

1. Текстовым редактором открыть файл `nodetab`, находящийся в каталоге `bin` установленного ЛИНТЕРа и добавить строки:  
`S2 ATCPIP s2.serv.com 1080`  
`S3 ATCPIP s3.serv.com 1080`  
`S4 ATCPIP s4.serv.com 1080`

**Только для NT, Windows 2000**

- На сервере s1:
1. Запустить утилиту «Администратор ЛИНТЕРа» (`linadm`).
  2. Выбрать базу данных `S1DB` и выполнить пункт меню **Database** ⇒ **Startup**. Запустить БД.
  3. Выбрать пункт меню **Network** ⇒ **Replication server** ⇒ **startup**. В появившемся диалоговом окне набрать имя и пароль к базе данных `S1DB` и хранилищу. Если пароли не переопределялись, то имена и пароли будут стандартными: `SYSTEM/MANAGER`. После нажатия ОК автоматически стартует БД хранилища.
  4. Повторить операции на серверах `s2, s3, s4` для соответственно БД `S2DB, S3DB, S4DB`.

**Только для UNIX**

1. Запустить БД `S1DB` на сервере `s1`.
2. Убедиться, что в файле `nodetab` есть строка типа:  
`RDSTG LOCAL 100`  
 Вместо числа `100` может быть другое число.
3. Установить переменную окружения `LINTER_MBX`, для этого выполнить команду командного интерфейса:  
`export LINTER_MBX=100`  
 или другое число, указанное в `nodetab`.
4. Установить переменную окружения `SY00`, для этого выполнить команду командного интерфейса:  
`export SY00=~/linter/db/storage`  
 Здесь должен быть указан путь к БД хранилища. Обычно это каталог `db/storage` в каталоге, куда установлен ЛИНТЕР.
5. Запустить БД хранилища.
6. Запустить сетевого клиента командой `dbc_tcp`.
7. Запустить сервер репликации командой:  
`lreplserver -l S1DB -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s RDSTG -d -p 1080`
8. Повторить операции на серверах `s2, s3, s4` для соответственно БД `S2DB, S3DB, S4DB`.

**На сервере s1**

1. Запустить утилиту `inl`.
2. Выполнить команды:  
`Create server s2;`  
`Create server s3;`  
`Create server s4;`
3. Создать правила репликации:  
`create replication rule tabrule1 for tab to`  
`tab on node S2 async;`  
`create replication rule tabrule1 for tab to`  
`tab on node S3 async;`  
`create replication rule tabrule1 for tab to`  
`tab on node S4 async;`
4. Занести запись в таблицу `TAB`.

**На серверах s2,**

1. Запустить утилиту `inl`.
2. Убедиться, что запись, занесенная в таблицу `TAB` на сервере

s3 и s4

s1 появилась в таблице TAB на локальном сервере.

### Встречная репликация между серверами s1 и s2

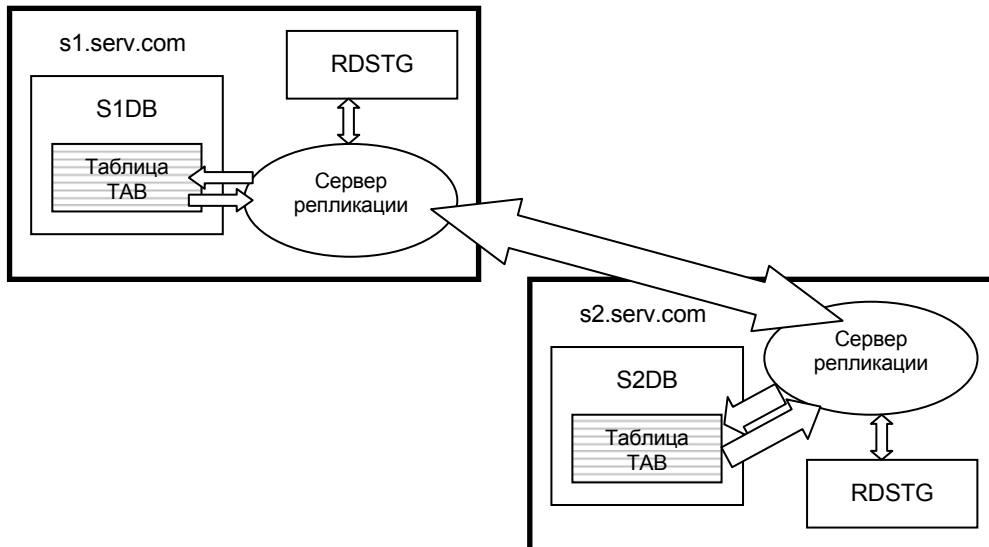


Рис. 10. Схема встречной репликации между серверами s1 и s2

#### Начальные данные

Имеется два сервера с IP адресами s1.serv.com и s2.serv.com. На каждом из них установлен ЛИНТЕР с поддержкой репликации. На каждом имеется таблица TAB с одинаковой структурой данных. Имена баз данных соответственно S1DB и S2DB.

#### Задача

Требуется настроить репликацию таблицы TAB с сервера s1 на сервер s2 и, одновременно, с сервера s2 на сервер s1.

#### Последовательность операций

##### На сервере s1

Текстовым редактором открыть файл nodetab, находящийся в каталоге bin установленного ЛИНТЕРа и добавить строку:  
S2 ATCPIP s2.serv.com 1080

##### На сервере s2

Текстовым редактором открыть файл nodetab, находящийся в каталоге bin установленного ЛИНТЕРа и добавить строку:  
S1 ATCPIP s1.serv.com 1080

##### Только для NT, Windows 2000

На сервере s1:

1. Запустить утилиту «Администратор ЛИНТЕРа» (linadm).
2. Выбрать базу данных S1DB и выполнить пункт меню **Database** ⇒ **Startup**. Запустить БД.
3. Выбрать пункт меню **Network** ⇒ **Replication server** ⇒ **startup**. В появившемся диалоговом окне набрать имя

Только для UNIX

и пароль к базе данных S1DB и хранилищу. Если пароли не переопределялись, то имена и пароли будут стандартными: SYSTEM/MANAGER. После нажатия ОК автоматически стартует БД хранилища.

4. Повторить операции на сервере s2 для БД S2DB.
1. Запустить БД S1DB на сервере s1.
2. Убедиться, что в файле nodetab есть строка типа:  
RDSTG LOCAL 100  
Вместо числа 100 может быть другое число.
3. Установить переменную окружения LINTER\_MBX, для этого выполнить команду командного интерфейса:  
export LINTER\_MBX=100  
Или другое число, указанное в nodetab.
4. Установить переменную окружения SY00, для этого выполнить команду командного интерфейса:  
export SY00=~/linter/db/storage
5. Здесь должен быть указан путь к БД хранилища. Обычно это каталог db/storage в каталоге, куда установлен ЛИНТЕР.
6. Запустить БД хранилища.
7. Запустить сетевого клиента командой dbc\_tcp.
8. Запустить сервер репликации командой:  
lreplserver -l S1DB -u SYSTEM/MANAGER  
-us SYSTEM/MANAGER -s RDSTG -d -p 1080
9. Повторить операции на сервере s2 для базы данных S2DB.

На сервере s1

1. Запустить утилиту inl.
2. Выполнить команду:  
create server s2;
3. Создать правило репликации:  
create replication rule tabrule for  
tab to tab on node s2 async;
4. Занести запись в таблицу TAB.

На сервере s2

1. Запустить утилиту inl.
2. Выполнить команду:  
create server s1;
3. Создать правило репликации:  
create replication rule tabrule for  
tab to tab on node s1 async;

На сервере s1

1. Запустить утилиту inl.
2. Занести запись в таблицу TAB.

На сервере s2

1. Запустить утилиту inl.
2. Убедиться, что запись, занесенная в таблицу TAB на сервере s1 появилась в таблице TAB на сервере s2.
3. Занести запись в таблицу TAB.

На сервере s1

1. Запустить утилиту inl.
2. Убедиться, что запись, занесенная в таблицу TAB на сервере s2 появилась в таблице TAB на сервере s1.

# Репликация с использованием ODBC

Сервер репликации имеет возможность работать с использованием ODBC. Эта возможность доступна только на стороне, принимающей данные. Для того, чтобы запустить сервер репликации в этом режиме, нужно задать ключ ‘-odbcsource DSN’:

## Пример

```
lreplserver -u SYSTEM/MANAGER -us SYSTEM/MANAGER -s REPLSERV -p 1080 -t 3 -log 4 -odbcsource Oracle
```

Здесь Oracle – это источник данных ODBC. В данном случае репликация будет производиться в базу данных Oracle, установленную на этом компьютере.

## Особенности репликации с использованием ODBC

Запуск, настройка и работа репликации происходит так, как описано выше (см. стр. 30), за исключением некоторых отличий.

1. нет никаких требований к системным таблицам БД приемника (не нужно прогонять файл `distr.sql`);
2. не будет работать механизм разрешения конфликтов.

## Пример репликации с ODBC



Сервер репликации должен быть установлен на двух серверах: источнике (И) и приемнике (П) (см.

рис. 8). Кроме того, на приемнике должен быть установлен и запущен источник данных ODBC, пусть это будет Oracle. Предположим, что нам нужно реплицировать таблицу TAB с источника на приемник, используя порт на приемнике 1081. TCP/IP адрес приемника будет `test.server.ru`.

Последовательность шагов:

| Шаг | Сервер | Действие                                                                                                                                            |
|-----|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | И      | Установить сервер репликации из дистрибутива                                                                                                        |
| 2   | П      | Установить сервер репликации из дистрибутива                                                                                                        |
| 3   | И      | Запустить БД и прогнать файл <code>distr.sql</code> – стр. 9. (Должны быть созданы системные таблицы \$\$\$REPL и SERVERS).                         |
| 4   | И      | Перезапустить БД                                                                                                                                    |
| 5   | И      | Создать новый сервер (например SRV) командой SQL (стр. 17):<br>SQL> Create server SRV;                                                              |
| 6   | И      | Дописать в <code>nodetab</code> строку с адресом нового сервера:<br>SRV ATCPIP test.server.ru 1081<br>Здесь 1081 – номер порта принимающей стороны. |

 Для серверов, на которые будет осуществляться репликация, в файле `nodetab` должны быть прописаны строки с протоколом ATCPIP (или ATCPIPS в UNIX

| Шаг | Сервер | Действие                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-----|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|     |        | при использовании протокола SSL).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 7   | И      | <p>Запустить БД хранилища. Местоположение БД хранилища указывается при установке сервера репликации. Это обычная БД ЛИНТЕР, ее запуск осуществляется стандартным образом. При запуске необходимо указать имя базы данных RDSTG, например, для NT:</p> <pre>Linter /local /name=RDSTG</pre>                                                                                                                                                                                                                                                                                                                                                                               |
| 8   | И      | Запустить клиент <code>dbc_tcp</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 9   | П      | Запустить БД хранилища аналогично источнику                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 10  | И      | <p>Запустить сервер репликации, как описано на стр. 6</p> <p>В NT рекомендуется запускать с ключом <code>-d</code>, при этом создается консоль, и на нее выводятся вся трассировка.</p> <div style="background-color: #e0e0e0; padding: 5px;"> <p> При первом запуске уровень трассировки желательно задавать не меньше 3, чтобы легче разрешить возможные проблемы.</p> </div>                                                                                                                                                                                                         |
| 11  | П      | <p>Запустить сервер репликации, как описано на стр. 6 , используя при этом ключ</p> <pre>/odbcsource=Oracle</pre> <p>Убедиться, что порт приема (задается через ключ <code>-p</code>) совпадает с прописанным в <code>nodetab</code> (в нашем случае 1081).</p> <p>В NT рекомендуется запускать с ключом <code>-d</code>, при этом создается консоль, и на нее выводятся вся трассировка.</p> <div style="background-color: #e0e0e0; padding: 5px;"> <p> При первом запуске уровень трассировки желательно задавать не меньше 3, чтобы легче разрешить возможные проблемы.</p> </div> |
| 12  | И      | Создать таблицу для репликации с первичным ключом (или убедиться, что он есть), например таблица TAB                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 12  | П      | Создать таблицу для приема, эквивалентную той, которая на источнике                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 13  | И      | Создать правило репликации с помощью <code>inl</code> , или «Рабочий стол ЛИНТЕРа», как описано на стр. 11                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| 14  | И      | Добавить запись в таблицу TAB                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 15  | П      | Убедиться, что такая же запись добавилась и на сервере <code>test.server.ru</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

# Сообщения сервера репликации

Сообщения сервера репликации включают следующие данные:

- 1) текст сообщения:
  - <STR> - в этом месте сообщения будет подставлена строка;
  - <NUM> - в этом месте сообщения будет подставлено число;
  - <FLT> - в этом месте сообщения будет подставлено число с плавающей точкой;
- 2) уровень трассировки;
- 3) значение сообщения репликации (содержит дополнительное описание сообщения репликации);
- 4) действия сервера репликации (пропуск рубрики означает, что работа продолжается в нормальном режиме);
- 5) рекомендации по устранению (пропуск рубрики означает, что ничего делать не нужно).

## Сообщения модуля разрешения конфликтов

### Конфликт между существующими и репликационными данными

#### Текст сообщения

```
Conflict DNEQ[FIRST]: <STR>(<STR>), no action.
Conflict DNEQ[NEW]: <STR>(<STR>), no action.
Conflict DNEQ[NEW]: <STR>(<STR>), no action.
Conflict DNEQ[OLD]: <STR>(<STR>), try to delete.
Conflict DNEQ[OLD]: <STR>(<STR>), try to delete.
Conflict DNEQ[SECOND]: <STR>(<STR>), try to delete.
Conflict DNEQ[WEIGHT]: <STR>(<STR>), no action.
Conflict DNEQ[WEIGHT]: <STR>(<STR>), try to delete.
Conflict DNEQ: <STR>(<STR>), no action
Conflict IEX: <STR>(<STR>) - but record not found!
Conflict IEX[FIRST]: <STR>(<STR>), no action.
Conflict IEX[NEW]: <STR>(<STR>), no action.
Conflict IEX[NEW]: <STR>(<STR>), no action.
Conflict IEX[OLD]: <STR>(<STR>), try to update.
Conflict IEX[OLD]: <STR>(<STR>), try to update.
Conflict IEX[SECOND]: <STR>(<STR>), try to update.
Conflict IEX[WEIGHT]: <STR>(<STR>), no action.
Conflict IEX[WEIGHT]: <STR>(<STR>), try to update.
Conflict UNEQ[FIRST]: <STR>(<STR>), no action.
Conflict UNEQ[NEW]: <STR>(<STR>), no action.
Conflict UNEQ[NEW]: <STR>(<STR>), no action.
Conflict UNEQ[OLD]: <STR>(<STR>), try to update.
Conflict UNEQ[OLD]: <STR>(<STR>), try to update.
Conflict UNEQ[SECOND]: <STR>(<STR>), try to update.
Conflict UNEQ[WEIGHT]: <STR>(<STR>), no action.
Conflict UNEQ[WEIGHT]: <STR>(<STR>), try to update.
Conflict UNEX[?]: <STR>(<STR>), no action
Conflict UNEX[FIRST]: <STR>(<STR>), no action.
Conflict UNEX[NEW]: <STR>(<STR>), no action.
Conflict UNEX[OLD]: <STR>(<STR>), no action.
Conflict UNEX[SECOND]: <STR>(<STR>), try to insert.
Conflict UNEX[WEIGHT]: <STR>(<STR>), no action.
Conflict UNEX[WEIGHT]: <STR>(<STR>), try to insert.
```

## Уровень трассировки

Уровень 3.

## Значение сообщения репликации

Произошел конфликт между существующими и репликационными данными. Указывается:

- 1) тип конфликта:
  - UNEQ - Попытка изменения записи, в которой не совпадает значение некоторых полей;
  - UNEX - Попытка изменить несуществующую запись;
  - DNEQ - Попытка удаления записи, в которой не совпадает значение некоторых полей;
  - DNEX - Попытка удалить несуществующую запись;
  - IEX - Попытка вставить запись с дубликатом первичного ключа
- 2) способ разрешения конфликта:
  - SECOND - Приоритет пришедшей записи (по умолчанию);
  - FIRST - Приоритет существующей записи;
  - NEW - Приоритет более новой записи;
  - OLD - Приоритет более старой записи;
  - WEIGHT - Приоритет сервера с большим весом
- 3) Имя таблицы и столбцы первичного ключа кортежа.

## Рекомендации по устранению

Выяснить причину возникновения конфликта и принять решение о дальнейших действиях.

## Арифметическое разрешение конфликта

### Текст сообщения

```

Calculate BIGINT <STR>.<STR>=<STR> / avg(<STR>,<STR>)
Calculate BIGINT <STR>.<STR>=<STR> / diff(<STR>,<STR>,<STR>)
Calculate BIGINT <STR>.<STR>=<STR> / max(<STR>,<STR>)
Calculate BIGINT <STR>.<STR>=<STR> / min(<STR>,<STR>)
Calculate DECIMAL <STR>.<STR>=<STR> / avg(<STR>,<STR>)
Calculate DECIMAL <STR>.<STR>=<STR> / diff(<STR>,<STR>,<STR>)
Calculate DECIMAL <STR>.<STR>=<STR> / max(<STR>,<STR>)
Calculate DECIMAL <STR>.<STR>=<STR> / min(<STR>,<STR>)
Calculate DOUBLE <STR>.<STR>=<STR> / avg(<FLT>,<FLT>)
Calculate DOUBLE <STR>.<STR>=<STR> / diff(<FLT>,<FLT>,<FLT>)
Calculate DOUBLE <STR>.<STR>=<STR> / max(<FLT>,<FLT>)
Calculate DOUBLE <STR>.<STR>=<STR> / min(<FLT>,<FLT>)
Calculate INT <STR>.<STR>=<STR> / avg(<NUM>,<NUM>)
Calculate INT <STR>.<STR>=<STR> / diff(<NUM>,<NUM>,<NUM>)
Calculate INT <STR>.<STR>=<STR> / max(<NUM>,<NUM>)
Calculate INT <STR>.<STR>=<STR> / min(<NUM>,<NUM>)
Calculate REAL <STR>.<STR>=<STR> / avg(<FLT>,<FLT>)
Calculate REAL <STR>.<STR>=<STR> / diff(<FLT>,<FLT>,<FLT>)
Calculate REAL <STR>.<STR>=<STR> / max(<FLT>,<FLT>)
Calculate REAL <STR>.<STR>=<STR> / min(<FLT>,<FLT>)

```

## Сообщения сервера репликации

---

```
Calculate SMALLINT <STR>.<STR>=<STR> / avg(<NUM>,<NUM>)
Calculate SMALLINT <STR>.<STR>=<STR> / diff(<NUM> , <NUM> ,<NUM>)
Calculate SMALLINT <STR>.<STR>=<STR> / max(<NUM> ,<NUM>)
Calculate SMALLINT <STR>.<STR>=<STR> / min(<NUM> ,<NUM>)
```

### Уровень трассировки

Уровень 4.

### Причина

Произошло арифметическое разрешение конфликта, новое значение числового поля получено в результате вычислений:

- АМАХ-Оставить большее;
- АМИН-Оставить меньшее;
- ААВЕР-Взять среднее;
- АДИФФ-Изменить текущее на разницу между старым и новым

## Арифметическое разрешение конфликта при нечисловом поле

### Текст сообщения

```
Priority FIRST <STR>.<STR>=<STR> , <STR>
Priority NEW <STR>.<STR>=<STR> , <STR>
Priority OLD <STR>.<STR>=<STR> , <STR>
Priority SECOND <STR>.<STR>=<STR> , <STR>
Priority WEIGHT <STR>.<STR>=<STR> , <STR>
```

### Уровень трассировки

Уровень 4.

## Ошибка при разрешении конфликта

### Текст сообщения

```
Error reading extra conflict data from <STR>
```

### Уровень трассировки

Уровень 2.

### Действия сервера репликации

Будет использован способ разрешения конфликта общий для всего кортежа.

### Рекомендации по устранению

Выяснить причину возникновения ошибки, используя дополнительные сообщения, например, об ошибке sql- запроса.

## Ошибка при расширенном разрешении конфликта

### Текст сообщения

```
Date is not equivalence from extra conflict data from <STR>
```

## Уровень трассировки

Уровень 2.

## Действия сервера репликации

Будет использован способ разрешения конфликта общий для всего кортежа.

## Сообщения модуля инициализации хранилища

### Ошибка при создании таблицы

#### Текст сообщения

```
Create table DISPATCH_ATTRS error <NUM>
Create table DISPATCH_QUEUE error <NUM>
Create table DISPATCH_QUEUE error <NUM>
Create table DISPATCH_STREAM error <NUM>
Create table DISPATCH_TABS error <NUM>
Create table RECEIVING_ATTRS error <NUM>
Create table RECEIVING_QUEUE error <NUM>
Create table RECEIVING_QUEUE_HIST error <NUM>
Create table RECEIVING_STREAM error <NUM>
Create table REPL_RULES error <NUM>
```

## Уровень трассировки

Уровень 1.

## Рекомендации по устранению

Используя код ошибки, выяснить ее причину.

## Сообщение о создании системной таблицы

#### Текст сообщения

```
Create table DISPATCH_ATTRS
Create table DISPATCH_QUEUE
Create table DISPATCH_QUEUE_HIST
Create table DISPATCH_STREAM
Create table DISPATCH_TABS
Create table RECEIVING_ATTRS
Create table RECEIVING_QUEUE
Create table RECEIVING_QUEUE_HIST
Create table RECEIVING_STREAM
Create table REPL_RULES
```

## Уровень трассировки

Уровень 3.

## Сообщения модуля выполнения транзакций

### Данные оказались большего размера, чем ожидалось

#### Текст сообщения

```
Rcv: Error reading data, small buffer!
```

## Сообщения сервера репликации

---

```
RQ:dt=<NUM> oper <STR> from <STR>(db:<STR>),trid=<STR>,
tab=<STR>,ri=<NUM>,L1=<NUM>, L2=<NUM>
```

### Уровень трассировки

Уровень 1.

### Причина

Данные оказались большего размера, чем ожидалось. Эта ошибка может возникнуть при порче буфера с данными. Второе сообщение содержит дополнительную информацию:

- dt – дата (номер) записи в таблице хранилища RECEIVING\_QUEUE;
- oper ... - тип операции;
- from ... - откуда пришли данные: имя сервера и БД;
- Tab – имя таблицы;
- Ri – rowid записи в исходной таблице;
- L1,L2 – длины двух буферов, которые оказались слишком большими

### Рекомендации по устранению

Пользуясь дополнительной информацией, постараться выяснить причину ошибки.

## Ошибка чтения дополнительного буфера данных

### Текст сообщения

```
Rcv: Error reading extra data (TrID=<STR>, date=<STR>,Err=<NUM>)!
```

### Уровень трассировки

Уровень 1.

## Сообщения процесса приема данных

## Ошибка добавления новых данных в хранилище

### Текст сообщения

```
RECEIVING_QUEUE: Start Append Error <NUM>
RECEIVING_QUEUE: PUTM Error <NUM>
RECEIVING_QUEUE: End Append Error <NUM>
```

### Уровень трассировки

Уровень 1.

### Причина

Произошла ошибка добавления новых данных в хранилище (в таблицу RECEIVING\_QUEUE). Указан код ошибки Линтера.

## Запуск процесса приема данных от конкретного сервера

### Текст сообщения

```
Start RCV process...
```

### Уровень трассировки

Уровень 0.

## Сообщение процесса приема

### Текст сообщения

RCV process for database '<STR>' use socket <NUM>

### Уровень трассировки

Уровень 5.

### Причина

Сообщение процесса приема, указано имя БД, от которой принимаются данные, и номер сокета.

## Сообщение о привязке к порту

### Текст сообщения

Bind to port <NUM>

### Уровень трассировки

Уровень 5.

## Запуск управляющего приемного процесса

### Текст сообщения

Start main RCV process

### Уровень трассировки

Уровень 0.

## Ошибка распаковки записи

### Текст сообщения

ERROR: INTEGER len=<NUM>, real len=<NUM>, tab <NUM> col <NUM> (<STR>)

### Уровень трассировки

Уровень 1.

### Причина

Произошла ошибка распаковки кортежа, реальная длина поля не совпала с указанной, выводится номер таблицы, номер и имя столбца.

### Действия сервера репликации

Длина поля приводится к допустимой длине, работа продолжается.

### Рекомендации по устранению

Проверить корректность записи (данные могут быть испорчены).

### Останов в результате отказа СУБД ЛИНТЕР

#### Текст сообщения

Stopped, Linter error: <STR> (<NUM>)

#### Уровень трассировки

Уровень 1.

#### Причина

Произошел останов в результате возникновения ошибки ядра СУБД ЛИНТЕР, при которой продолжение работы невозможно. Указывается номер ошибки и текстовая расшифровка (если возможно).

#### Действия сервера репликации

Конец работы.

#### Рекомендации по устранению

В зависимости от типа ошибки принять меры по ее устранению.

### Ошибка выделения памяти

#### Текст сообщения

Stopped, no memory, system error <NUM>.  
Fatal error: no memory.

#### Уровень трассировки

Уровень 1, уровень 0.

#### Действия сервера репликации

Конец работы.

### Внутренняя ошибка сервера репликации

#### Текст сообщения

Stopped, internal error: buffer empty.

#### Уровень трассировки

Уровень 1.

#### Действия сервера репликации

Конец работы.

#### Рекомендации по устранению

Связаться с разработчиком.

## Корректный останов сервера репликации по команде пользователя

### Текст сообщения

Lintel replication server down.

### Уровень трассировки

Уровень 1.

### Действия сервера репликации

Конец работы.

## Неизвестная внутренняя ошибка сервера репликации

### Текст сообщения

Stopped, unknown error type <NUM>, system error <NUM>.

### Уровень трассировки

Уровень 1.

### Действия сервера репликации

Конец работы.

### Рекомендации по устранению

Связаться с разработчиком.

## Сообщения сервера репликации

### Переполнен список одновременно исполняемых транзакций

#### Текст сообщения

Transaction list full, try to extend it...

#### Уровень трассировки

Уровень 4.

#### Причина

Переполнен список одновременно исполняемых транзакций. Список будет расширяться. Вначале выделяется буфер для 2048 одновременно выполняемых транзакций, если этого оказывается недостаточно, то он увеличивается на 1024 каждый раз, когда это необходимо.

### Подается SQL-запрос к БД

#### Текст сообщения

cb1[<CMD>,<DB>,<CHAN>] <SQL command>

### Уровень трассировки

Уровень 5.

### Причина

Подается sql-запрос к БД. Указана команда, имя БД и номер открытого канала. Может быть указан sql-запрос, если он был.

## Некритическая ошибка

### Текст сообщения

```
Attention! (err#<ERRNUM>): cbl[<CMD>,<DB>,<CHAN>] <SQL command>
```

### Уровень трассировки

Уровень 1.

### Причина

Произошла некритическая ошибка. Указан номер ошибки и параметры запроса, при котором она возникла.

### Действия сервера репликации

Работа продолжается.

### Рекомендации по устранению

Выяснить причину ошибки.

## Запрашиваемая таблица заблокирована другим пользователем

### Текст сообщения

```
Data locked(<NUM>), try again...
```

### Уровень трассировки

Уровень 3.

### Действия сервера репликации

Делается попытка повторного обращения после паузы.

### Рекомендации по устранению

Выяснить, кто и почему заблокировал таблицу.

## При обращении к СУБД ЛИНТЕР получен не нулевой код возврата

### Текст сообщения

```
Lintner return code <NUM>, RowCount=<NUM>
```

### Уровень трассировки

Уровень 5.

### Причина

При обращении к ядру СУБД ЛИНТЕР получен код возврата, отличный от 0. Это может быть штатная ситуация.

### Действия сервера репликации

Работа продолжается.

### Рекомендации по устранению

Проанализировать ситуацию и выяснить причину ошибки.

## Ошибка при открытии канала к СУБД ЛИНТЕР

### Текст сообщения

```
Error open channel for Linter (err=<NUM>,node='<STR>', nodetab='<STR>')
```

### Уровень трассировки

Уровень 1.

### Причина

Произошла ошибка при открытии канала к ядру СУБД ЛИНТЕР. Указан код ошибки, узел, к которому была попытка соединения и используемый nodetab.

### Действия сервера репликации

Конец работы.

### Рекомендации по устранению

Выяснить, правильно ли заданы параметры соединения, запущено ли ядро СУБД ЛИНТЕР.

## Добавление данных в хранилище

### Текст сообщения

```
Start appends <NUM> records
```

### Уровень трассировки

Уровень 2.

### Действия сервера репликации

Работа продолжается.

## Ошибка при добавлении новых данных в хранилище

### Текст сообщения

```
DISPATCH_QUEUE: Start Append Error <NUM>
DISPATCH_QUEUE: PUTM Error <NUM>
DISPATCH_QUEUE: End Append Error <NUM>
```

### Уровень трассировки

Уровень 1.

## Сообщения сервера репликации

---

### Причина

Произошла ошибка при добавлении новых данных в хранилище. Указан код ошибки.

### Действия сервера репликации

Работа продолжается.

### Рекомендации по устранению

В зависимости от кода ошибки определить, почему данные не были добавлены.

## Внутренняя ошибка сервера репликации

### Текст сообщения

```
Fatal error: Data buffer full!
```

### Уровень трассировки

Уровень 0.

### Действия сервера репликации

Конец работы.

### Рекомендации по устранению

Связаться с разработчиком СУБД ЛИНТЕР.

## Правила репликации были изменены и сейчас будут перезагружены

### Текст сообщения

```
Replication rules changed.
```

### Уровень трассировки

Уровень 2.

## Неизвестное сообщение, внутренняя ошибка

### Текст сообщения

```
Unknown record <NUM>!
```

### Уровень трассировки

Уровень 1.

### Действия сервера репликации

Работа продолжается.

### Рекомендации по устранению

Связаться с разработчиком СУБД ЛИНТЕР.

## Открывается специальный канал к ядру СУБД ЛИНТЕР

### Текст сообщения

Open special channel (Log addr <NUM>:<NUM>)

### Уровень трассировки

Уровень 3.

## Ошибка открытия спецканала

### Текст сообщения

Error open special channel for Linter - <NUM>.

### Уровень трассировки

Уровень 1.

### Причина

Ошибка открытия спецканала. Делается попытка повторить операцию.

### Действия сервера репликации

Если канал открыть не удастся, то работа заканчивается.

### Рекомендации по устранению

Проверить параметры запуска сервера репликации.

## Ошибка при обращении к ядру СУБД ЛИНТЕР

### Текст сообщения

Can't find Linter (<NUM>)

### Уровень трассировки

Уровень 0.

### Действия сервера репликации

Конец работы.

### Рекомендации по устранению

Проверить параметры запуска сервера репликации.

## Получен сигнал Ctrl+Break

### Текст сообщения

SIGINT has occurred, stop the program (pid <NUM>).

### Уровень трассировки

Уровень 0.

### Действия сервера репликации

Конец работы.

### Ошибка при получении данных по спецканалу

#### Текст сообщения

Receiving data: Linter error code <NUM>

#### Уровень трассировки

Уровень 0.

#### Действия сервера репликации

Конец работы.

#### Рекомендации по устранению

В зависимости от кода ошибки выяснить причину ошибки.

### Сообщения о начале и конце синхронизации таблиц

#### Текст сообщения

Start synchronize table.  
Stop synchronize table.

#### Уровень трассировки

Уровень 2.

### Сообщения о начале и конце перекачки данных

#### Текст сообщения

Start find and recovery accumulated records.  
End recovery accumulated records - <NUM> found and recovering now.

#### Уровень трассировки

Уровень 2.

#### Значение сообщения

Сообщения о начале и конце перекачки накопившихся за время останова сервера репликации данных.

### Ошибка открытия канала к основной БД

#### Текст сообщения

Error open channel for Linter (err=<NUM>,node='<STR>', nodetab='<STR>')

#### Уровень трассировки

Уровень 1.

#### Причина

Ошибка открытия канала к основной БД. Делается попытка повторить операцию.

#### Действия сервера репликации

Если канал открыть не удастся, то работа заканчивается.

### Рекомендации по устранению

Проверить параметры запуска сервера репликации.

## Ошибка открытия канала к хранилищу

### Текст сообщения

```
Error open channel for storage (err=<NUM>,node='<STR>', nodetab='<STR>').
```

### Уровень трассировки

Уровень 1.

### Причина

Ошибка открытия канала к хранилищу. Делается попытка повторить операцию.

### Действия сервера репликации

Если канал открыть не удастся, то работа заканчивается.

### Рекомендации по устранению

Проверить параметры запуска сервера репликации.

## Ошибка запроса имени сервера назначения

### Текст сообщения

```
Can't read server name (<NUM>)
```

### Уровень трассировки

Уровень 1.

### Рекомендации по устранению

В зависимости от кода ошибки определить, что произошло.

## Ошибка запроса имени таблицы для репликации

### Текст сообщения

```
Can't read table names (<NUM>)
```

### Уровень трассировки

Уровень 1.

### Рекомендации по устранению

В зависимости от кода ошибки определить, что произошло.

## Ошибка запроса имени владельца таблицы

### Текст сообщения

```
Can't read owner name (<NUM>)
```

## Сообщения сервера репликации

---

### Уровень трассировки

Уровень 1.

### Рекомендации по устранению

В зависимости от кода ошибки определить, что произошло.

## Удаление устаревших данных из системных таблиц

### Текст сообщения

```
Begin delete old data
End delete old data
```

### Уровень трассировки

Уровень 2.

## Идет очистка хранилища

### Текст сообщения

```
Begin clear storage
End clear storage
```

### Уровень трассировки

Уровень 2.

## Считывание правил репликации

### Текст сообщения

```
Begin download replication rules
End download replication rules - <NUM> OK
```

### Уровень трассировки

Уровень 2.

### Причина

Считывание правил репликации, указано количество считанных правил.

## Отсутствует таблица правил репликации

### Текст сообщения

```
Error: replication rules table ($$$REPL) not exist! Run distr.sql
```

### Уровень трассировки

Уровень 0.

### Действия сервера репликации

Выход.

### Рекомендации по устранению

Выполнить скрипт distr.sql.

## Ни одного правила репликации не было создано

### Текст сообщения

```
Replication rules not exist.
```

### Уровень трассировки

Уровень 0.

## Ошибка при считывании правил репликации

### Текст сообщения

```
Can't read replication rules. Linter exit code <NUM>!
```

### Уровень трассировки

Уровень 1.

### Рекомендации по устранению

В зависимости от кода ошибки определить, что произошло.

## Считывание структуры таблиц для репликации

### Текст сообщения

```
Begin download replication tables
End download replication tables - <NUM> OK
```

### Уровень трассировки

Уровень 2.

### Причина

Считывание структуры таблиц для репликации, указано количество считанных таблиц.

## Ошибка запуска сервера репликации как сервиса

### Текст сообщения

```
StartServiceCtrlDispatcher failed: err#<NUM>.
```

### Уровень трассировки

Уровень 1.

### Действия сервера репликации

Сервер репликации запустится как обычное приложение.

## Окончание работы сервера репликации

### Текст сообщения

```
Working done.
Terminate Lreplserver (<NUM>)...
Working done (pid=<NUM>).
```

## Сообщения сервера репликации

---

### Уровень трассировки

Уровень 2.

### Причина

Окончание работы сервера репликации. Первое сообщение – для Win32, остальные – для Unix.

## Контрольная точка в системном журнале очищена

### Текст сообщения

```
Clear control point: done.
```

### Уровень трассировки

Уровень 1.

## Контрольная точка в системном журнале не найдена

### Текст сообщения

```
Clear control point: not found.
```

### Уровень трассировки

Уровень 1.

## Старт сервера репликации

### Текст сообщения

```
Start replication server
```

### Уровень трассировки

Уровень 2.

## Уровень трассировки

### Текст сообщения

```
Trace level <NUM>
```

### Уровень трассировки

Уровень 1.

## Задан неверный вес сервера для разрешения конфликтов

### Текст сообщения

```
Invalid server weight <NUM>, set default (3)
```

### Уровень трассировки

Уровень 2.

### Действия сервера репликации

Устанавливается вес по умолчанию.

### Рекомендации по устранению

Проверить параметры запуска. Вес должен быть в пределах 0-7.

## Устанавливается вес сервера для разрешения конфликтов

### Текст сообщения

```
Set server weight <NUM>
```

### Уровень трассировки

Уровень 2.

## Указан режим работы

### Текст сообщения

```
Service flag is: <NUM>
```

### Уровень трассировки

Уровень 2.

### Причина

Указан режим работы, если 1 – программа стартует как сервис.

## Указан используемый файл nodetab

### Текст сообщения

```
Use nodetab file '<STR>'
```

### Уровень трассировки

Уровень 2.

## Указан pid сервера репликации

### Текст сообщения

```
Current Lreplserver pid=<NUM>
```

### Уровень трассировки

Уровень 2.

## Началось ожидание данных

### Текст сообщения

```
Wait data mode started
```

## Сообщения сервера репликации

---

### Уровень трассировки

Уровень 1.

## Освобождение ресурсов перед остановом

### Текст сообщения

Cleanup...

### Уровень трассировки

Уровень 1.

## Происходит останов процесса

### Текст сообщения

Terminate Process (pid=<NUM>)...

### Уровень трассировки

Уровень 2.

## Сообщения модуля рассылки данных репликации

## Пришел запрос на правила разрешения конфликтов

### Текст сообщения

Extra resolve a conflicts record request from <STR>.

### Уровень трассировки

Уровень 4.

### Причина

Пришел запрос на правила разрешения конфликтов. Указано имя сервера, пославшего запрос.

## Неправильное имя сервера репликации

### Текст сообщения

Error: server '<STR>' have to use ATCPIP or ATCPIPS protocol.

### Уровень трассировки

Уровень 1.

### Причина

Имя сервера, указанное для репликации, описано в nodetab неправильно. Протокол должен быть ATCPIP или ATCPIPS.

### Рекомендации по устранению

Исправить файл сетевой конфигурации nodetab.

## Произошло соединение с указанным сервером через указанный порт

### Текст сообщения

Connect to server <STR> port <NUM>

### Уровень трассировки

Уровень 5.

## Останов процесса рассылки на указанный сервер

### Текст сообщения

Stop out: '<STR>'

### Уровень трассировки

Уровень 0.

## Стартует процесс рассылки на указанный сервер

### Текст сообщения

Start SND process for node '<STR>'

### Уровень трассировки

Уровень 0.

## Сообщения процесса очистки

## Стартует процесс регулярной очистки хранилища

### Текст сообщения

Start CLEAR process

### Уровень трассировки

Уровень 0.

## Очистка очередей хранилища

### Текст сообщения

Clear storage queue (<STR>)

### Уровень трассировки

Уровень 4.

### Причина

Произошла очистка очередей хранилища начиная с даты <STR> (дата выводится во внутреннем представлении).

# Сообщения о событиях на реплицируемой БД

## Закончена транзакция

### Текст сообщения

```
Commit (Tr=<STR>)
Commit complete
```

### Уровень трассировки

Уровень 2.

### Причина

Закончена транзакция. Если есть данные в этой транзакции, предназначенные для репликации, то они будут пересылаться.

## Выполнен откат транзакции

### Текст сообщения

```
Rollback (Tr=<STR>)
Rollback complete
```

### Уровень трассировки

Уровень 2.

## Частичный откат транзакции по ошибке ссылочной целостности

### Текст сообщения

```
Small Rollback (Tr=<STR>)
Small Rollback complete
```

### Уровень трассировки

Уровень 2.

## Выполнена указанная операция

### Текст сообщения

```
Delete record (Tr=<STR>)
Delete complete
Insert record (Tr=<STR>)
Insert complete
Update record (Tr=<STR>)
Update complete
Create blob (Tr=<STR>)
Create blob complete
Clear blob (Tr=<STR>)
Clear blob complete
```

### Уровень трассировки

Уровень 2.

## Внутренняя ошибка сервера репликации

### Текст сообщения

Fatal error: Data buffer full!

### Уровень трассировки

Уровень 0.

### Действия сервера репликации

Конец работы.

### Рекомендации по устранению

Связаться с разработчиком.



