



OpenStore.Сервер данных (Руководство администратора)

Оглавление

Оглавление	1
Общее описание.....	1
Корневой сервер данных.....	2
Подчиненный сервер данных.....	3
Установка и настройка.....	7
Установка.....	7
Настройка.....	9
Закладка «Общие».....	9
Закладка «База данных».....	11
Закладка «Порты соединений».....	12
Закладка «Синхронизация данных».....	14
Закладка «Система лояльности».....	17
Закладка «Прокси».....	19
Закладка «Сообщения».....	20
Закладка «Лицензирование».....	21
Закладка «Поставка данных».....	22
Закладка «Передача файлов».....	28
Закладка «Синхронизация с OSM».....	29
Параметры профиля синхронизации.....	31
Маска синхронизации.....	34
Управление работой.....	43
Закладки основного окна.....	44
Закладка «Клиенты».....	44
Закладка «Синхронизация клиентов».....	44
Закладка «Поставка данных».....	45
Закладка «Поставка данных с OSM».....	45
Закладка «Журнал сервера данных».....	46
Закладка «Мониторинг».....	46
Кнопки управления основного окна.....	47
Технологические особенности работы сервера.....	50
DATASYNK.....	50
DATAPUMP.....	52

Общее описание

«OpenStore.Сервер данных» (далее просто «Сервер данных») представляет собой серверное приложение, предназначенное для распространения данных (справочников и файловых каталогов от внешней системы к клиентам и результатов работы последних – к внешней системе) по всей системе, сколь бы сложной она не была.

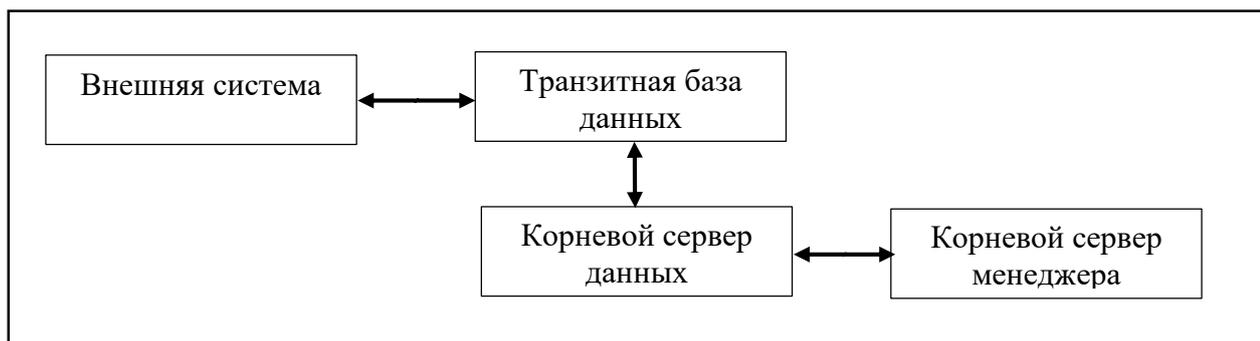
Сервер представляет собой Windows-сервис, работающий со своей собственной базой данных, доступ к которой осуществляется исключительно самим сервисом. Это позволяет повысить безопасность системы, надежно изолируя базу от внешней среды. В простейшем случае сервер баз данных и сервер данных находятся на одном физическом сервере, однако, при необходимости (например для повышения производительности) допускается их разнесение по разным

физическим серверам так как сервер данных корректно обрабатывает временный разрыв связи с сервером баз данных.

В зависимости от своего положения в иерархии архитектуры системы сервер данных может быть корневым или подчиненным.

Корневой сервер данных

Корневой сервер данных является основным посредником между внешней системой и комплексом OpenStore (есть еще возможность обмена данными с использованием соответствующего шлюза посредством XML-запросов, но такой подход используется редко и имеет ряд ограничений, в т.ч. и по производительности). В качестве источника для такого сервера данных используется «транзитная база данных», к которой ему требуется непосредственный доступ (подробнее механизм обмена описан в документе «Синхронизация с внешней системой»). Предполагается, что сервер данных получает доступ к транзитной базе по локальной сети, поэтому никаких механизмов оптимизации (сжатия, шифрования) на этом этапе не предусмотрено.



Вышеприведенная схема иллюстрирует взаимодействие сервера данных с внешней системой и корневым сервером менеджера. Передаваемые внешней системой данные сохраняются в транзитную базу, откуда в последствии забираются сервером данных. Зеркально выполняется получение данных. Кроме того, с корневым сервером данных взаимодействует корневой сервер менеджера, к базе которого сервер данных должен иметь непосредственный доступ (такой же как и к транзитной базе).

Корневой сервер менеджера предназначен для управления теми справочниками, которыми не может управлять внешняя система (например права кассиров, настройки профиля касс и т.д.). Он не является обязательным, однако отсутствие корневого менеджера исключает использование подчиненных менеджеров (более подробно о работе менеджера в документе «OpenStore.Менеджер (руководство администратора)»).

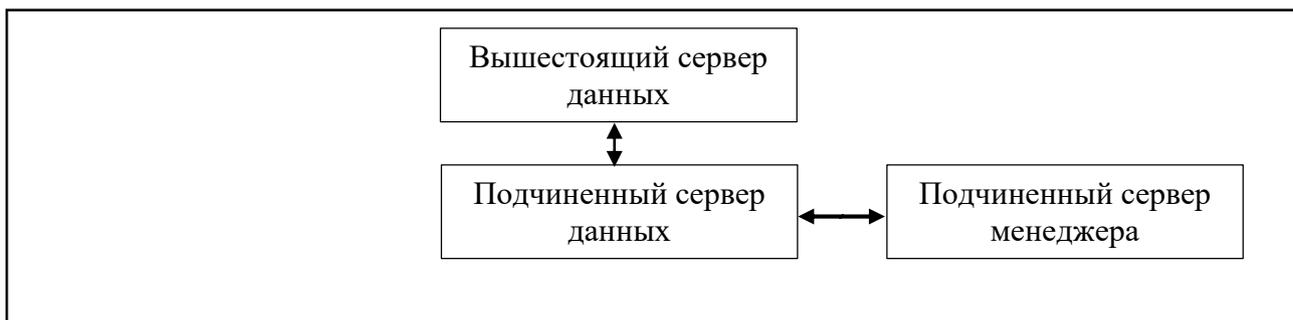
Корневой сервер данных является источником уникальных идентификаторов для элементов тех справочников, с которыми не работает внешняя система, поэтому потеря базы корневого сервера

данных достаточно критична, а ее восстановление потребует высококвалифицированного вмешательства.

Подчиненный сервер данных.

Подчиненный сервер данных отличается от корневого тем, что источником данных для него служит другой сервер данных (в т.ч. и корневой). Для взаимодействия с вышестоящим сервером данных, подчиненному серверу требуется только подключение по одному из рабочих каналов (ТСР или НТТР – подробнее описано ниже). При обмене данными между серверами данных используется потоковое сжатие. В случае разрыва соединения в момент получения данных используется «докачка» что позволяет минимизировать время передачи данных при медленных и / или не стабильных физических каналах связи.

Подчиненный сервер данных может работать с подчиненным сервером менеджера, и при том лишь с одним. Для этого требуется непосредственный доступ сервера данных к базе сервера менеджера.

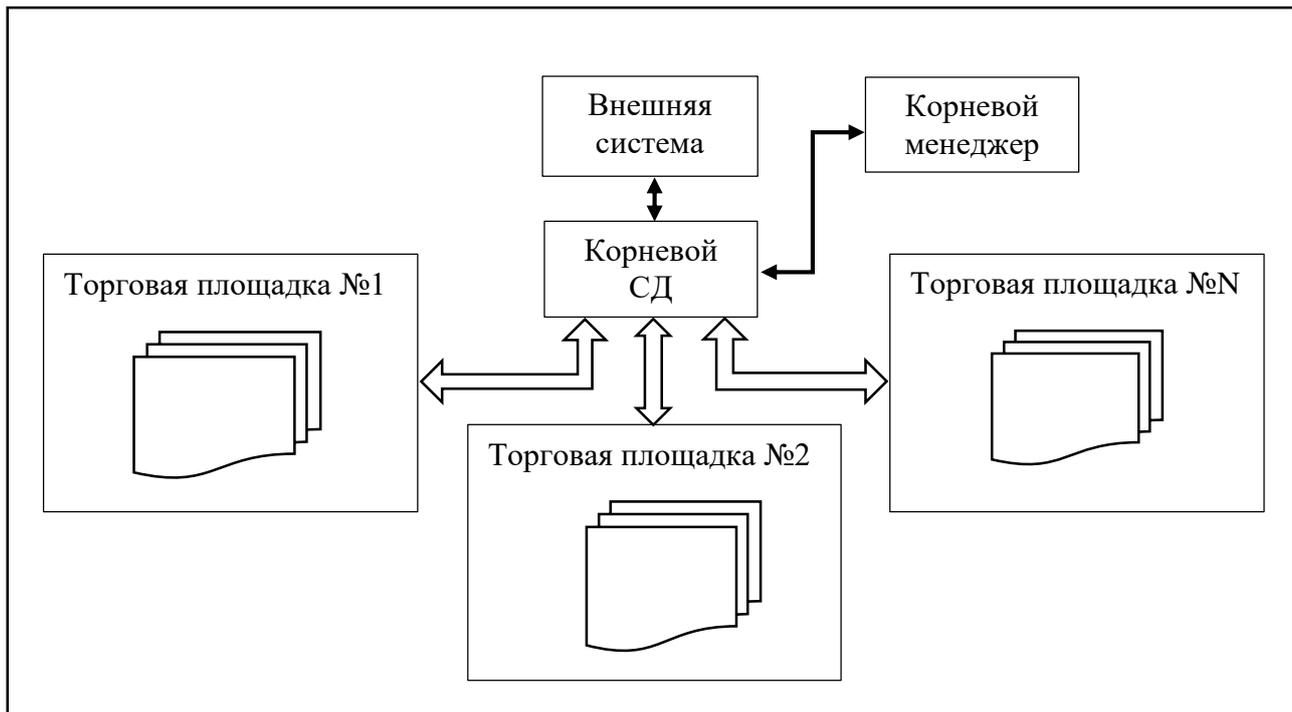


Вышеприведенная схема иллюстрирует взаимодействие подчиненного сервера данных с вышестоящим и с подчиненным менеджером. Количество последовательно включаемых серверов данных не ограничено, однако чрезмерное удлинение этой цепочки увеличивает время прохождения данных, поэтому для получения эффективной системы требуется тщательное планирование.

Использование серверов данных.

Правильное использование сервера данных позволяет сократить трафик и повысить как скорость распространения данных, так и канало-независимость сегментов системы. Для большего понимания «правильности» использования сервера данных рассмотрим несколько архитектурных решений, и опишем их достоинства и недостатки.

Простейшим решением является использование сервера данных только в качестве шлюза между внешней системой и комплексом OpenStore. Ниже приведена схема такого решения:



В такой архитектуре существует только один (корневой) сервер данных и каждый клиент (сколько бы их ни было) является клиентом этого сервера данных. К достоинствам такой архитектуры можно отнести:

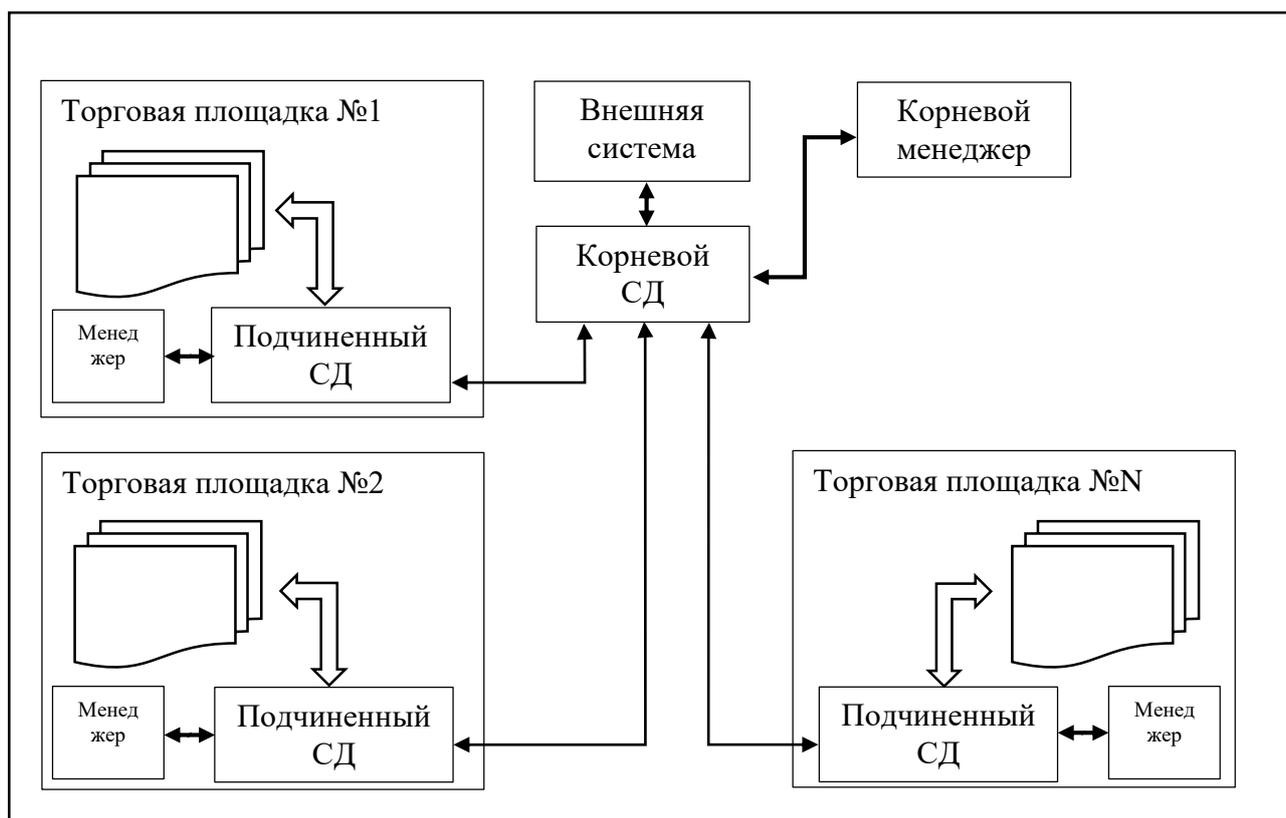
- относительную простоту развертывания рабочих мест (можно использовать копирование с предопределенными настройками, однако все равно требуется перенастраивать идентификатор клиента и идентификатор ТП),
- нет «серверного» компьютера на торговой площадке, все компьютеры равноценны и взаимозаменяемы,
- допустимо перемещение компьютера с одной торговой площадки на другую, без существенной перенастройки (только изменив собственно идентификатор ТП в настройках и обновив справочники цен),
- появляется возможность наблюдать за состоянием (синхронизацией) всех клиентов в консоли одного сервера (только при не большем количестве клиентов),
- сокращается количество промежуточных этапов синхронизации, что снижает время появления на клиентах новых данных (это верно только для не большего (до 30-50) количества клиентов, т.к. с ростом их числа издержки на параллельную синхронизацию становятся соизмеримыми с таймаутом синхронизации).

Однако, такая схема обладает и достаточно большим количеством недостатков:

- в первую очередь требуется обеспечение уникальности идентификатора клиента в пределах всей системы, что потребует принятия специальных организационных мер, а так же может вызвать проблемы связанные с человеческим фактором (ошибка уникальности идентификатора может быть обнаружена не сразу, если часть клиентов на момент запуска нового клиента находятся в выключенном состоянии или у них временно отсутствует канал связи);
- т.к. каждый из клиентов непосредственно подключается к корневому серверу данных, то объем передаваемых по каналу связи от сервера к клиентам данных увеличивается кратно количеству клиентов торговой площадки;
- в случае, если канал связи с корневым сервером отсутствует, новый клиент не может быть запущен в эксплуатацию;
- т.к. синхронизация клиентов осуществляется в порядке живой очереди (порциями, размер которых указывается в настройках сервера данных), то достаточно высока вероятность того, что на одной и той же торговой площадке на некоторый промежуток времени (возможно несколько минут, а в случае разрыва связи и на более продолжительный) окажутся клиенты с разными наборами данных (например с разными ценами);
- теряется канал-независимость при работе с менеджером (работа с журналами, формирование отчетов, управление справочниками);
- в случае использования SSL-шифрования потребуется большее количество сертификатов (для каждого клиента);
- один сервер данных может обслуживать не более 250 клиентов одновременно, но т.к. в некоторых решениях используются пользовательские запросы от кассы к корневой системе (передаваемые по каналу серверов данных, например пополнение мобильного телефона, использование талонов, сторонней системы лояльности и т.д.), рекомендуется работа одного сервера данных максимум со 125 клиентами (два подключения на клиента: основное и пользовательское).

Из вышеизложенного следует, что использование архитектуры с одним сервером данных имеет смысл только для небольших систем, где на каждой торговой площадке работает только одна касса и не используется менеджер.

Свободной от вышеизложенных недостатков является архитектура с сервером данных на каждой торговой площадке. Ниже приведена схема такого решения:



Достоинствами такой архитектуры являются:

- объем передаваемых по каналу данных не зависит от количества клиентов на торговой площадке (данные один раз принимаются подчиненным сервером данных и потом раздаются клиентам по локальной сети);
- исключается длительное расхождение данных на клиентах одной торговой площадки (в случае не большого количества клиентов (до 20-30) исключается в принципе);
- торговая площадка сохраняет полноценную работоспособность при отсутствии канала связи (можно редактировать справочники, смотреть отчеты, работать с журналами, запускать новых клиентов);
- в случае использования SSL-шифрования, достаточно защитить только канал сервер-сервер.

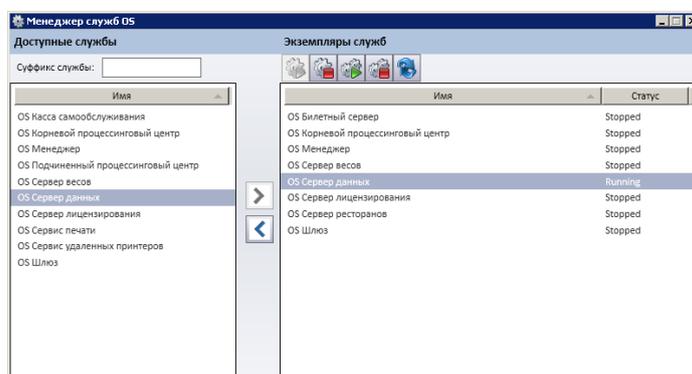
Недостатком такой схемы является необходимость наличия дополнительного компьютера, используемого в качестве сервера или выделение «главной» кассы, на компьютере которой будет работать подчиненный сервер данных (этот компьютер должен быть всегда включен) и подчиненный сервер менеджера (если требуется).

Вышеописанная архитектура является типовой и рекомендована к применению.

Установка и настройка

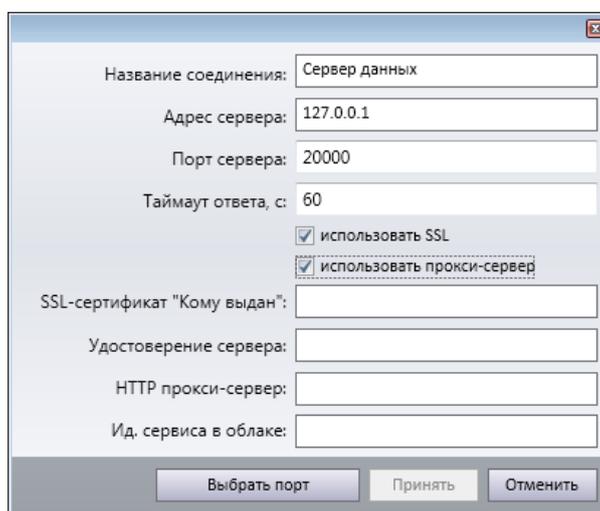
Установка.

Прежде всего сервер данных следует установить, как Windows-сервис. Для этого в стандартной поставке предусмотрен «Менеджер сервисов». Запустив его, следует выбрать «OS Сервер данных» в списке слева и добавляем в список установленных сервисов (список справа).



После этого Windows-сервис может быть запущен как при помощи «Менеджера сервисов» так и штатными средствами Windows.

Для управления сервером данных и его настройками используется «Консоль управления» так же предусмотренная в стандартной поставке. Для подключения к конкретному серверу данных в консоли управления нужно создать соответствующее «соединение», в котором указать параметры подключения к каналу управления требуемого сервера данных:

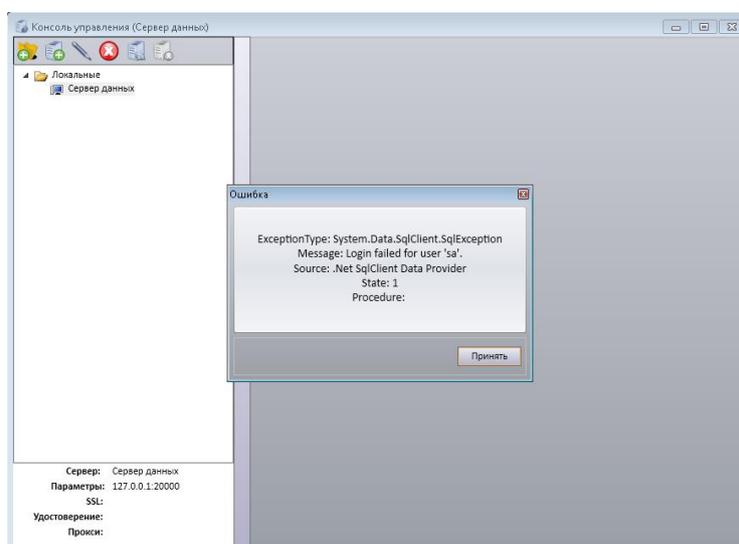


Параметры соединения сервера данных:

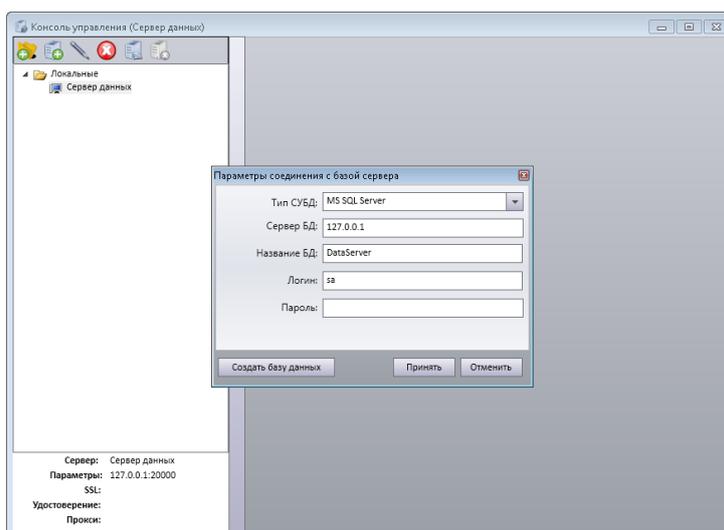
- «Название соединения» - логическое имя соединения, позволяющее отличать одно соединение от другого (один сервер от другого). Может содержать любой текст, в т.ч. с пробелами.
- «Адрес сервера» - ip-адрес компьютера, на котором запущен сервер данных (консоль обеспечивает удаленное управление серверами).

- «Порт сервера» - порт, определенный серверу данных, как порт управления (по умолчанию -20000, может быть переопределен путем редактирования файла настроек сервера данных (DataServer.xml), а именно его параметра RemoteManagementPort).
- «Таймаут сервера» - таймаут ожидания ответа от сервера, в секундах (по умолчанию - 60).
- «Использовать SSL» - позволяет устанавливать защищенный канал управления сервером данных (для этого на самом сервере данных следует включить поддержку SSL отредактировав в файле настроек (DataServer.xml) значение параметра RemoteManagementSslSubjectName, указав в качестве его значения имя сертификата (subject name) канала управления, установленного в хранилище сертификатов того компьютера, на котором запускается сервер данных (личные сертификаты)).
- «Использовать прокси-сервер» - позволяет соединению работать через прокси-сервер (канал управления работает только по HTTP-каналу).
- «SSL-сертификат «Кому выдан»» - имя сертификата (subject name) подключения консоли, установленного в хранилище сертификатов того компьютера, на котором запускается клиент (личные сертификаты).
- «Удостоверение сервера» - имя открытого сертификата канала управления сервера данных (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается клиент (доверенные сертификаты).
- «HTTP прокси-сервер» - адрес и порт (127.0.0.1:3128) прокси сервера, через которые должно работать данное соединение.

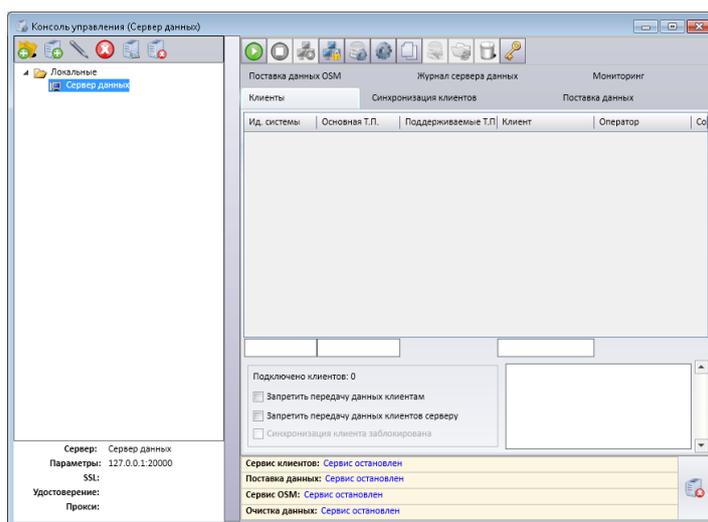
Используя созданное подключение переходим к настройке сервера данных. При первом запуске потребуется указать подключение к базе данных сервера данных или создать новую, о чем будет свидетельствовать ошибка подключения к базе данных:



После нажатия кнопки «Принять», отобразится форма подключения к базе данных:



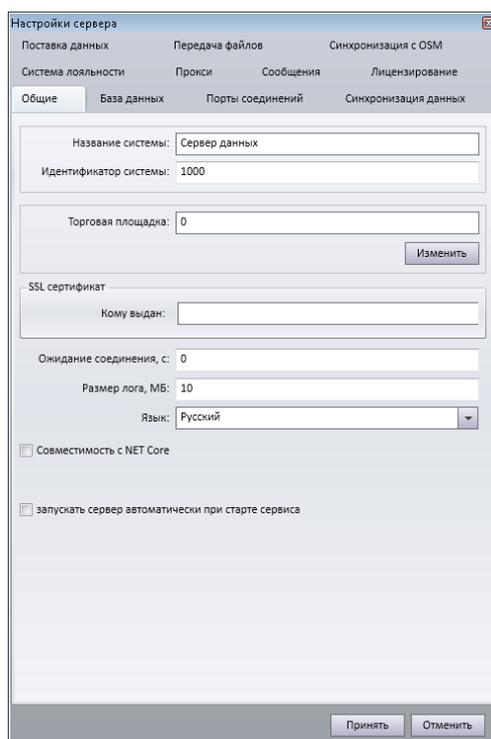
Если база данных уже существует, то после указания данных, необходимых для подключения, достаточно нажать кнопку «Принять». В противном случае следует нажать кнопку «Создать базу данных». Во обоих случаях сервер перезапустится, подключившись к базе данных и отобразится основная форма управления:



Если сервер подключен к ранее существовавшей (рабочей) базе данных, то он готов к работе. Если же база была создана только что, то прежде всего следует перейти в меню настроек, нажав соответствующую кнопку: 

Настройка.

Закладка «Общие».



Параметры закладки «Общие»:

- «Название системы» - логическое имя данного сервера, предназначенное для визуальной его идентификации в списке клиентов вышестоящего сервера данных.
- «Идентификатор системы» - уникальный в пределах всей системы идентификатор сервера данных, предназначенный для корректной передачи данных по направлению к корню системы (входит в первичный ключ ряда таблиц, в т.ч. SYSLOG).
- «Торговая площадка» - список поддерживаемых сервером данных торговых площадок (если 0 – поддерживаются все торговые площадки). Ограничение поддерживаемых торговых площадок приводит к фильтрации по ним принимаемых из внешней системы данных таблиц SAREAPRC, PACKPRC, PACKPRCHISTORY, ARTSDEPT, SAREAOFFER.
- «SSL сертификат» - имя сертификата (subject name) используемого данным сервером, как для подключения к вышестоящему серверу данных (если это подчиненный сервер), так и для работы с клиентами, и установленного в хранилище сертификатов того компьютера, на котором запускается сервер (личные сертификаты).
- «Ожидание соединения, с» - время ожидания подключения к базе данных при запуске, в секундах. При запуске компьютера складывается такая ситуация, при которой сервер базы данных (MS SQL, Postgre SQL, Oracle) может запуститься позже сервера данных, что в свою очередь не позволит последнему подключиться к базе данных при запуске, и как следствие – начать нормально работать после запуска компьютера. Для разрешения этой проблемы используется задержка перед подключением к базе данных, определяемая

данной настройкой. Значение задержки следует подбирать эмпирическим путем. Обычно достаточно 10-30 секунд (в зависимости от типа СУБД и производительности компьютера).

- «Размер лога, МБ» - размер файлового лога, по достижении которого начинается новый файл. В файловый лог пишутся в т.ч. и внутренние ошибки, возникающие на сервере данных, даже если при этом ему не удалось подключиться к базе данных, поэтому он является очень важным инструментом в диагностике проблем возникающих при работе сервера. Однако работать со слишком большим файлом не удобно, могут возникнуть проблемы со средствами просмотра лога. Значение по умолчанию – 10 МБ.
- «Язык» - позволяет выбрать язык, на котором будут создаваться сообщения сервера (не путать с языком меню консоли управления).
- «Совместимость с NET Core» - включает режим совместимости с клиентами, работающими не под .NET4.X а под «NET Core».
- «Запускать сервер автоматически при старте сервиса» - включение этой настройки приведет к тому, что сервер автоматически запустится сразу после запуска Windows-сервиса.

Закладка «База данных».

Настройки сервера

Поставка данных Передача файлов Синхронизация с OSM

Система лояльности Прокси Сообщения Лицензирование

Общие **База данных** Порты соединений Синхронизация данных

Параметры подключения к базе данных

Версия базы данных: 7.17

Тип СУБД: PostgreSQL

Сервер БД: 127.0.0.1

Название БД: DataServer33

Логин: postgres

Максимальный размер базы данных, МБ: 100000

Сохранять записи логов, суток: 10

Сохранять остальные записи, суток: 90

Расписание удаления устаревших записей...

удалять игнорируемые данные

проверять, переданы ли данные во внешнюю систему

проверять, переданы ли данные в OSM

Принять Отменить

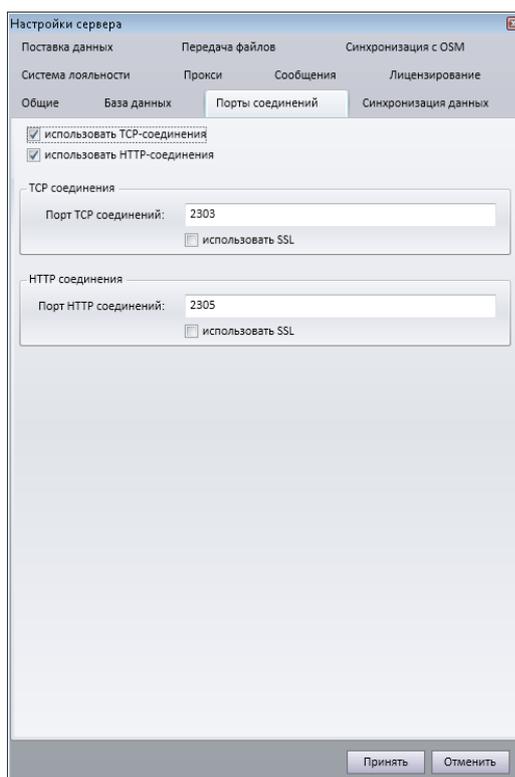
Параметры закладки «База данных»:

- «Максимальный размер базы данных, МБ» - этот параметр не ограничивает размер базы данных, а позволяет информировать администратора посредством системы

мониторинга о том, что база данных достигла определенного размера. Рост размера базы данных может свидетельствовать о том, что требуется проведение определенных профилактических процедур, или некорректно настроена очистка устаревших данных.

- «Сохранять записи логов, суток» - количество суток, в течение которых будут сохраняться данные таблицы SYSLOG.
- «Сохранять остальные записи, суток» - количество суток, в течение которых будут сохраняться данные таблиц передаваемых на корневую систему, исключая таблицу SYSLOG (например SALES, SESS, WORKDAY и т.д.).
- «Расписание удаления устаревших записей» - позволяет определить, когда и как часто будут удаляться устаревшие записи. При настройке планировщика удаления следует помнить, что во время удаления устаревших записей работа клиентов будет невозможна (а при большем объеме данных удаление может занимать несколько десятков минут и больше), но удаление не будет выполняться, если компьютер окажется выключен.
- «Удалять игнорируемые данные» - включение этой настройки приводит к тому, что при выполнении очистки устаревших данных таблицы, которые согласно настройкам не передаются ни во внешнюю систему не в базу менеджера будут удаляться без учета значений UPDATENUM и UPDATENUM2, в противном случае удаление происходит в соответствии с настройками «Проверять, переданы ли данные во внешнюю систему» и «Проверять, переданы ли данные в OSM».
- «Проверять, переданы ли данные во внешнюю систему» - включение этой настройки приводит к тому, что при очистке устаревших данных для удаления записи требуется, чтобы она была передана во внешнюю систему (UPDATENUM=1).
- «Проверять, переданы ли данные в OSM» - включение этой настройки приводит к тому, что при очистке устаревших данных для удаления записи требуется, чтобы она была передана в базу менеджера (UPDATENUM2=1).

Закладка «Порты соединений».



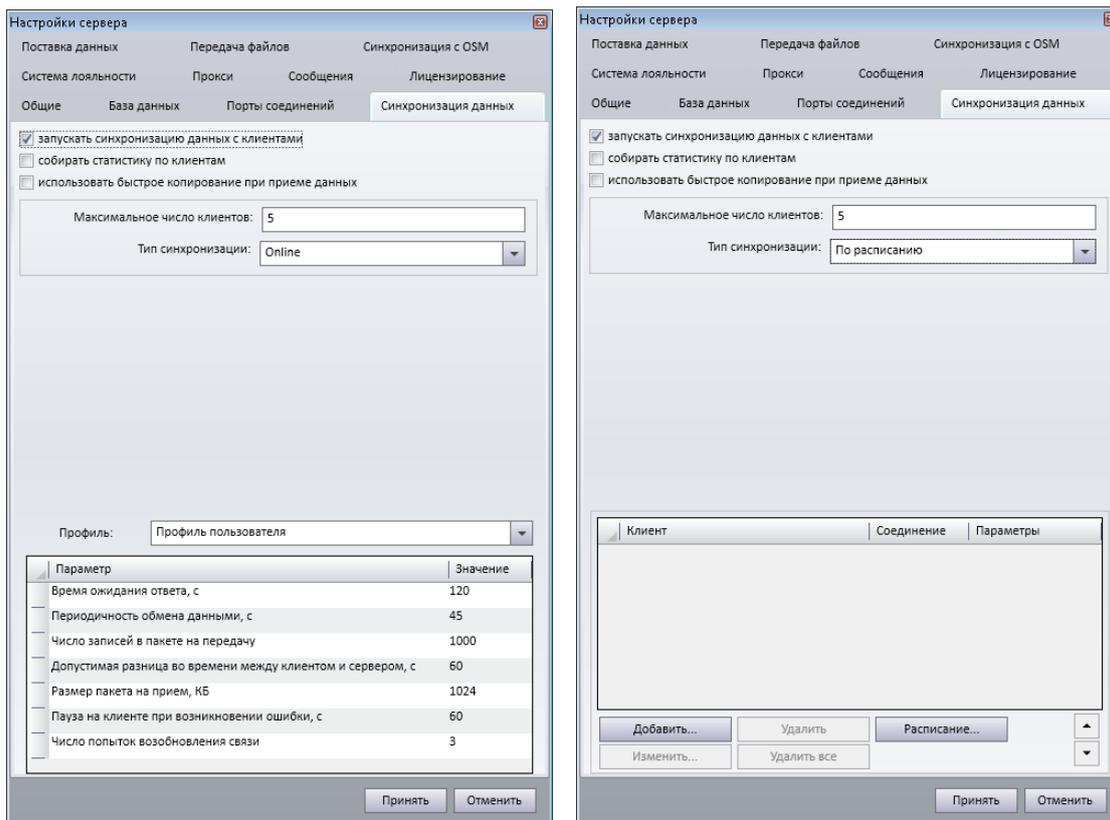
Параметры закладки «Порты соединений»:

- «Использовать TCP-соединение» - эта настройка включает TCP-канал, по которому сервер работает с клиентами.
- «Использовать HTTP-соединение» - эта настройка включает HTTP-канал, по которому сервер работает с клиентами.
- «Порт TCP соединений» - позволяет переопределить порт по которому работает TCP канал по работе с клиентами, по умолчанию – 2303.
- «TCP соединение»\«Использовать SSL» - требует установления TCP соединения только по защищенному каналу.
- «Порт HTTP соединений» - позволяет переопределить порт по которому работает HTTP канал по работе с клиентами, по умолчанию – 2305.
- «HTTP соединение»\«Использовать SSL» - требует установления HTTP соединения только по защищенному каналу.

Как следует из настроек – сервер данных может одновременно работать по обоим каналам. При этом любой из них или оба могут быть защищены шифрованием.

TCP канал рекомендован для внутренних сетей, т.к. требует прямого IP доступа от клиента к серверу. К достоинствам TCP канала можно отнести меньший создаваемый трафик и несколько большую скорость передачи данных. HTTP канал наиболее целесообразно использовать при работе через интернет.

Закладка «Синхронизация данных».

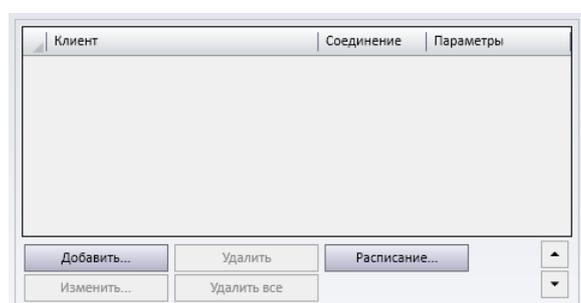


Параметры закладки «Синхронизация данных»:

- «Запускать синхронизацию данных с клиентами» - включает серверную часть сервера данных, если эта настройка не включена, то порты соединений не открываются.
- «Собирать статистику по клиентам» - эта настройка включает работу системы мониторинга (подробнее описана ниже) на данном сервере.
- «Использовать быстрое копирование при приеме данных» - эта настройка актуальна только для СУБД MS SQL2008 и выше (на всех остальных СУБД приведет к ошибке). Вместо select и в зависимости от результата update или insert, использует insert во временную таблицу с последующим merge, что приводит к увеличению скорости изменения данных в 2-4 раза.
- «Максимальное число клиентов» - максимальное число одновременно обслуживаемых клиентов. Оптимальное значение этой настройки подбирается эмпирическим путем и зависит от производительности и загруженности компьютера, на котором запускается сервер и сервер баз данных. Т.к. для каждого обслуживаемого клиента открывается свое соединение к базе данных, то при определенных лицензионных требованиях СУБД, этот параметр может быть использован для их удовлетворения (при этом к количеству клиентов следует добавить еще два соединения используемых непосредственно сервером).
- «Тип синхронизации» - предусмотрено два типа синхронизации: клиент-серверная («online») и инициированная сервером («по расписанию»). В первом случае (типовом) каждый клиент при запуске устанавливает соединение с сервером и периодически обменивается с ним

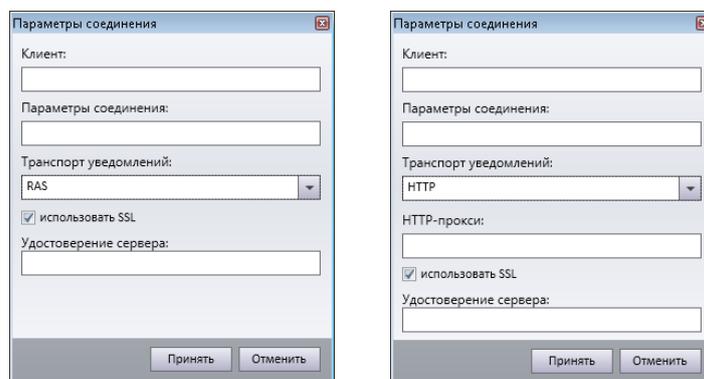
данными. Этот случай предполагает наличие у сервера известного (фиксированного) IP-адреса, а у клиентов доступ в сеть, откуда доступен этот адрес (прямо или через PROXY-сервер). Регистрация / удаление клиента на сервере в этом случае выполняется автоматически. При использовании модемов для установления канала связи или периодически возникающих каналах связи предпочтительнее, чтобы сервер обзванивал или поднимал VPN-соединение по очереди с каждым из своих клиентов (если клиенты будут звонить на сервер, то возникнут проблемы с синхронизацией этих звонков). Для решения этой задачи предусмотрен второй тип синхронизации – «по расписанию» (поддерживается только между серверами данных).

Настройка клиентов для синхронизации «по расписанию»:



Для того, чтобы сервер мог инициировать связь с клиентом, требуется вручную зарегистрировать последнего, а так же настроить расписание синхронизации.

Для добавления клиента используем кнопку «Добавить»:



Для случая RAS (соединение с использованием модема):

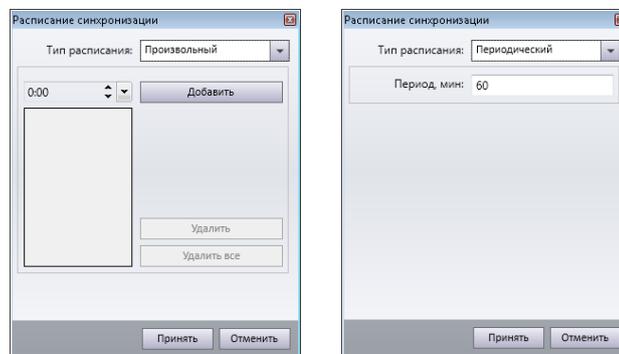
- «Клиент» - имя созданного средствами операционной системы RAS-соединения,
- «Параметры соединения» - порт, прослушиваемый подчиненным сервером данных для HTTP-канала (смотри настройки закладки «Порты соединений»).
- «Использовать SSL» - если HTTP канал подчиненного сервера требует SSL, следует включить эту настройку.

- «Удостоверение сервера» - имя открытого сертификата HTTP канала подчиненного сервера данных (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается родительский сервер данных (доверенные сертификаты).

Для случай HTTP (соединение по сети, по расписанию):

- «Клиент» - логическое имя клиента, предназначенное для того, чтобы отличить одного клиента от другого.
- «Параметры соединения» - IP-адрес и порт через двоеточие, определяющие подключение к HTTP-каналу подчиненного сервера (например 192.168.0.1:2305).
- «HTTP-прокси» - IP-адрес и порт через двоеточие, определяющие подключение к прокси-серверу (например 192.168.1.1:3128).
- «Использовать SSL» - если HTTP канал подчиненного сервера требует SSL, следует включить эту настройку.
- «Удостоверение сервера» - имя открытого сертификата HTTP канала подчиненного сервера данных (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается родительский сервер данных (доверенные сертификаты).

Для настройки расписания используем кнопку «Расписание»:



Для типа «Произвольный» указывается одна или несколько временных точек, в которые будет осуществляться синхронизация. Для типа «Периодический» - период времени в минутах, по истечении которого будет повторяться попытка синхронизации с клиентами.

Профиль синхронизации касс.

Профиль: <input type="text" value="Профиль пользователя"/>	
Параметр	Значение
Периодичность обмена данными, с	45
Число записей в пакете на передачу	1000
Допустимая разница во времени между клиентом и сервером, с	60
Размер пакета на прием, КБ	1024
Пауза на клиенте при возникновении ошибки, с	60
Число попыток возобновления связи	3
Время ожидания ответа, с	120

Все серверные приложения OpenStore имеют собственные настройки синхронизации. Однако для кассы (а так же кассы самообслуживания, кассы ресторана, кассы АЗС и билетной кассы) сделано исключение. Дело в том, что касс может быть достаточно много и поэтому для них предпочтительнее использовать централизованное (посредством сервера) управление параметрами синхронизации. Касса, на этапе регистрации на сервере получает и сохраняет себе в локальную базу данные настройки. До первой успешной регистрации на сервере касса использует настройки «по умолчанию». Более подробно настройки профиля синхронизации описаны в разделе «Параметры профиля синхронизации».

Закладка «Система лояльности».

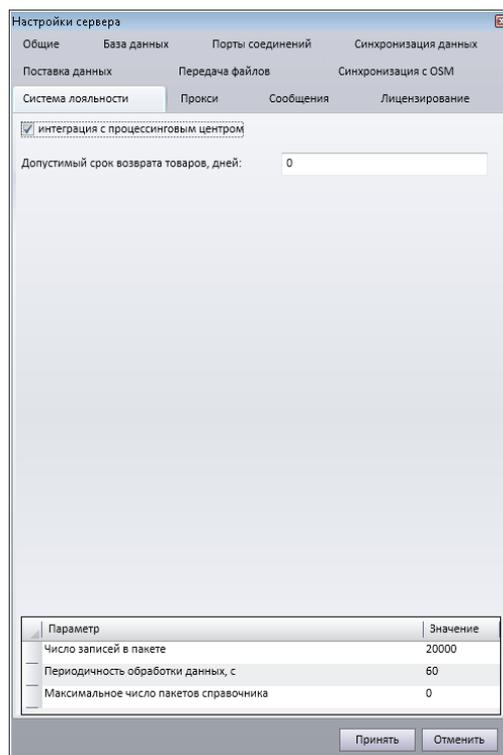
Задачи по интеграции с процессингом:

- передача начислений / списаний на бонусный / накопительный счет из таблицы SALESBONUS (поднимающейся с касс) в таблицу BACKACCOUNTTRANS, передаваемую вниз, на корневой процессинговый центр;
- передача рассчитанных остатков на бонусном / накопительном счету приходящих от корневого процессинга в таблице PCCLNTSUM в таблицу CLNTSUM, спускаемую на кассы;
- передача заявок на изменение данных, сформированных подчиненными процессингами из таблицы LOCALREQUEST в таблицу PCREQUESTEX.

При необходимости, все эти задачи могут быть решены внешней системой, что позволит ей вмешиваясь в вышеописанные процессы реализовывать те или иные пользовательские схемы работы. Однако, как правило, решение этих задач оставляют серверу данных.

Следует помнить, что режим интеграции с процессингом будет работать только на корневом сервере данных, в то время, как включение его на подчиненных серверах ни к чему не приведут.

Внимание!!! Если вы используете интеграцию с процессинговым центром на сервере данных, убедитесь, что таблица SALESBONUS исключена из маски синхронизации с внешней системой!!!



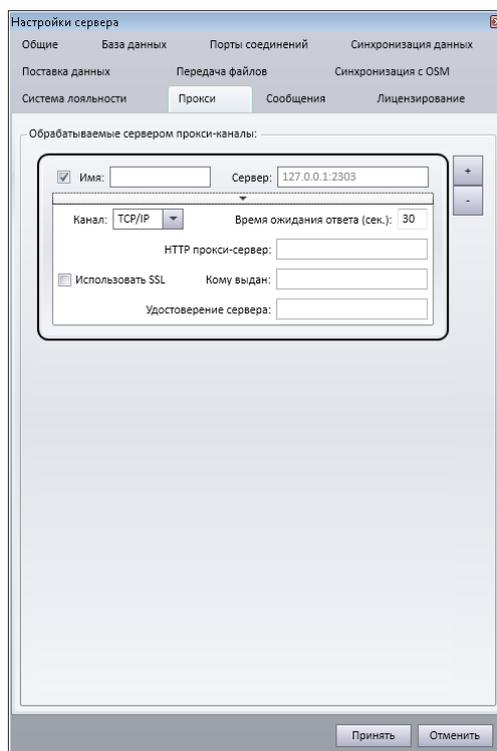
Параметры закладки «Система лояльности»:

- «Интеграция с процессинговым центром» - включает режим интеграции с процессингом на данном сервере данных (если это не корневой сервер, значение этой настройки воспринимается как «выключено»).
- «Допустимый срок возвратов, дней» - количество дней, на которое будет задерживаться начисления на бонусный счет. Эта настройка предназначена для разрешения проблемы манипуляций с бонусным счетом со стороны клиентов, а именно: приобретение дорогого товара и получение соответствующей суммы бонусов на счет, покупка на бонусы другого товара, возврат дорогого товара в течение 14-и дней (задержка начисления бонусов на 14 дней полностью исключает эту манипуляцию).
- «Число записей в пакете» - размер пакета, обрабатываемого сервером данных за один раз. Уменьшение размера пакета позволяет несколько снизить потребляемые сервером данных ресурсы, но при этом одновременно и снижается скорость обработки.
- «Периодичность обработки данных» - время в секундах, по истечении которого повторяется цикл обработки.
- «Максимальное число пакетов справочника» - по умолчанию (значение 0) обработка продолжается до тех пор, пока остаются данные для обработки. Для очень больших и нагруженных систем это может привести к закликиванию на этой операции, т.к. за время выполнения цикла будут появляться новые данные (хотя бы одна запись). Для предотвращения этой ситуации можно ограничить количество последовательно

обрабатываемых пакетов, что позволит регулярно выполняться полному циклу синхронизации.

Закладка «Прокси».

Кроме передачи данных от клиентов наверх и от внешней системы к клиентам, последовательная цепочка серверов данных может использоваться как прокси, для передачи пользовательских запросов от клиентов к шлюзам или внешней системе (используется только в пользовательских решениях, в штатном функционале не требуется). В этом случае клиенту не требуется настройки дополнительного канала. Он адресует свой запрос к тому серверу данных, клиентом которого является. Сервер данных, получивший запрос, перенаправляет его своему родителю и т.д., до тех пор, пока не будет достигнут корневой сервер, или сервер, у которого в настройках будет указан прокси-канал с именем, совпадающим с именем запроса. В первом случае (запрос достиг корневого сервера, но у него нет канала с именем, соответствующим имени запроса) будет сгенерирована ошибка, указывающая на отсутствие получателя запроса, во втором (на одном из серверов цепочки, в т.ч. и корневом найден подходящий канал) запрос будет перенаправлен на найденный канал и его передача по цепочке вверх прервется.



Параметры прокси-канала:

- «Имя» - логическое имя канала, которое должно совпадать с именем указанным в запросе (должно содержаться в документации к соответствующему пользовательскому проекту).
- «Сервер» - ip-адрес и пор конечного сервера – получателя запроса.

- «Канал» - тип канала к серверу-получателю запроса (TCP / HTTP, рекомендации по применению того или иного типа канала даны при описании закладки «Порты соединений»).
- «Время ожидания ответа (сек.)» - время ожидания ответа (в секундах) от внешнего шлюза / сервера, к которому перенаправляется запрос. По истечении времени ожидания в качестве ответа будет сгенерирована соответствующая ошибка.
- «HTTP прокси-сервер» - ip-адрес и порт через двоеточие (192.168.0.1:3128) используемого при связи по HTTP-каналу прокси-сервера.
- «Использовать SSL» - включает защиту канала шифрованием.
- «Кому выдан» - имя сертификата (subject name) подключения канала, установленного в хранилище сертификатов того компьютера, на котором запускается сервер данных (личные сертификаты).
- «Удостоверение сервера» - имя открытого сертификата канала внешнего шлюза, к которому перенаправляется запрос (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается клиент (доверенные сертификаты).

Закладка «Сообщения».

Для уведомления службы поддержки о возникновении ошибки / проблемы на сервере данных может быть использовано автоматическое уведомление по электронной почте.



The screenshot shows the 'Server Settings' dialog box with the 'Messages' tab selected. The 'Send messages by email about server status' checkbox is checked. Below it are input fields for 'SMTP-server', 'Receiver address', and 'Sender address'. A 'Time to block, min:' field is set to '60'. There is a 'Send test message' button. At the bottom of the dialog are 'Accept' and 'Cancel' buttons.

Параметры закладки «Сообщения»:

- «Отправлять сообщения по электронной почте о состоянии сервера» - позволяет включить режим отправки уведомлений.

- «SMTP сервер» - ip-адрес сервера SMTP (только адрес, порт стандартный).
- «Адрес получателя» - содержит адреса получателей сообщения. Несколько адресов электронной почты необходимо разделять запятыми (",").
- «Адрес отправителя» - формальный адрес отправителя.
- «Время блокировки, мин.» - часто на сервере данных возникают периодические ошибки (например ошибки синхронизации), которые повторяются с частотой периода синхронизации. Чтобы избежать огромного количества однотипных писем, используется данная настройка. Она определяет, через какой период письмо с одним и тем же сообщением об ошибке будет отправлено повторно.
- «Отправить тестовое сообщение» - позволяет отправить немедленное тестовое сообщение, позволяя проверить корректность настроек.

Закладка «Лицензирование».

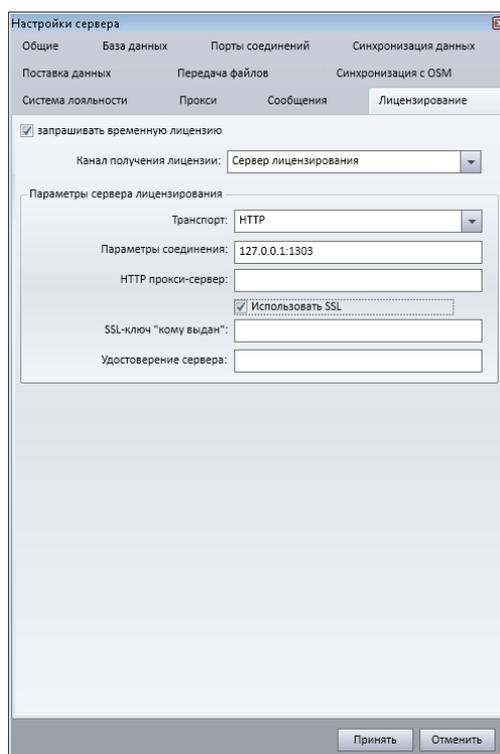
Сервер данных может сам выступать носителем лицензии, но только своей. Такой механизм оставлен по большей части для совместимости с продуктами – прототипами, и на сегодняшний день не рекомендован к применению. Основным носителем лицензии в комплексе OpenStore является «Сервер лицензирования».

Существует два способа запроса лицензии: посредством родительского сервера данных (последний выступает как прокси) и непосредственно с сервера лицензирования. Для корневого сервера данных последний способ является единственно возможным, а для подчиненных выбор способа получения лицензии определяется архитектурой системы. В любом случае, в ответ на свой запрос, сервер данных получает временную лицензию с временем жизни 30 дней, что позволяет в дальнейшем сохранять полноценную автономную работу на протяжении вышеуказанного периода. Однако повторный запрос лицензии следует через 10 минут, что дает возможность:

- поддерживать полученную лицензию всегда в актуальном состоянии (когда бы не исчез канал связи, с этого момента всегда есть 30 дней);

- учитывать внесенные на сервере лицензирования обновления (если на последнем произведено перераспределение / дополнение лицензий, то сервер данных получит эти изменения автоматически в течение 10 минут).

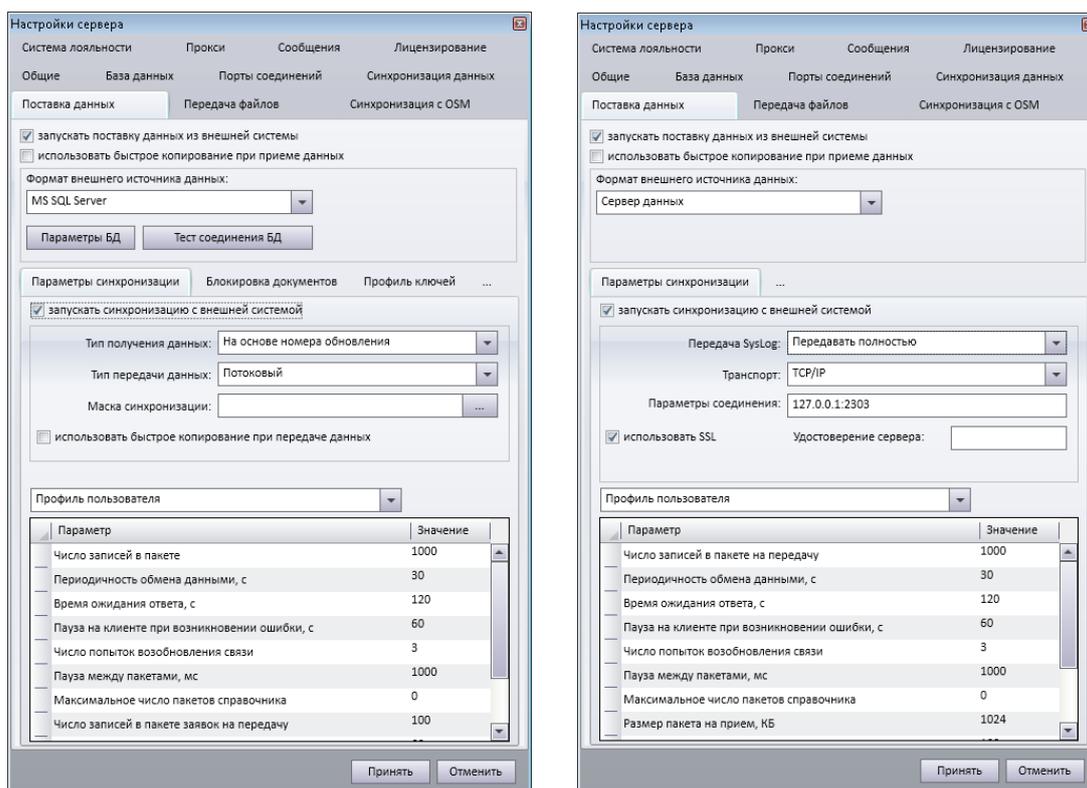
В случае, если при выполнении запроса к серверу лицензирования возникла ошибка, запросы лицензии учащаются до 1-го запроса в 2-е минуты, что позволяет быстрее обнаруживать восстановление канала связи и / или сервера лицензирования и актуализировать текущую лицензию.



Параметры закладки «Лицензирование»:

- «Запрашивать временную лицензию» - включает режим запроса лицензии у сервера лицензирования (в противном случае требуется установка лицензии непосредственно на сервер данных вручную).
- «Канал получения лицензии» - позволяет определить способ запроса временной лицензии (подробнее описано выше).
- «Транспорт» - тип канала, по которому будет осуществляться связь с сервером лицензирования.
- «Параметры соединения» - ip-адрес и порт (127.0.0.1:1303) для подключения к серверу лицензирования (по умолчанию для TCP канала – порт 1303, для HTTP канала - 1305).
- «HTTP прокси-сервер» - ip-адрес и порт через двоеточие (192.168.0.1:3128) прокси-сервера.
- «Использовать SSL» - включение этой настройки приводит к попытке установить зашифрованный канал связи с сервером лицензирования (для того чтобы попытка была успешной, сервер лицензирования должен поддерживать шифрование для канала данного типа).
- «Удостоверение сервера» - имя открытого сертификата канала сервера лицензирования (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается сервер данных (доверенные сертификаты).

Закладка «Поставка данных».



Параметры закладки «Поставка данных»:

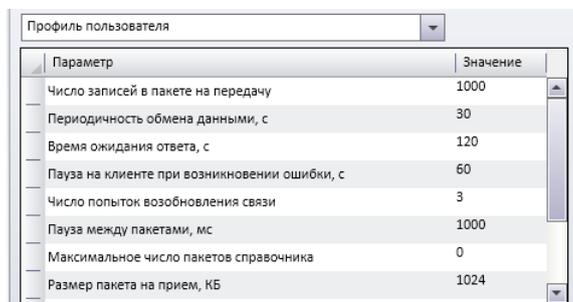
- «Запускать поставку данных» - включает работу сервиса поставки данных. Выключенное состояние не является штатным и применяется исключительно в сервисных и отладочных целях. При отключенном сервисе поставки сервер не считается корневым.
- «Использовать быстрое копирование при приеме данных» - работает только для MS SQL 2008 и выше (на всех остальных СУБД приведет к ошибке), вместо select и в зависимости от результата update или insert, использует insert во временную таблицу с последующим merge, что приводит к увеличению скорости изменения данных в 2-4 раза. Используется для принимаемых сверху данных.
- «Формат внешнего источника данных» - этот параметр определяет, что является источником данных для данного сервера данных: транзитная база данных или родительский сервер данных, что в свою очередь определяет, является ли данный сервер данных корневым или подчиненным (подробнее о корневом и подчиненном в разделе «Общее описание»).
- «Параметры БД» - позволяет указать настройки подключения к БД выбранного типа.
- «Тест соединения» - выполняет пробное подключение в указанной в настройках выше базе данных.
- «Запускать синхронизацию с внешней системой» - позволяет не запускать реальную синхронизацию, сохраняя при этом включенным как передачу данных, так и определенным

значение настройки «Формат внешнего источника». Используется для реализации работы сервера данных в качестве корневого без реальной транзитной базы и внешней системы.

- «Тип получения данных» - используется для корневого сервера, позволяет выбрать протокол обмена с транзитной базой на прием (подробнее описан в документе «Синхронизация с внешней системой»).
- «Тип передачи данных» - используется для корневого сервера, позволяет выбрать протокол обмена с транзитной базой на передачу (подробнее описан в документе «Синхронизация с внешней системой»).
- «Маска синхронизации» - используется для корневого сервера, позволяет выбрать используемую маску синхронизации, описывающую сопоставление полей базы данных и транзитной базы (более подробно о маске синхронизации в соответствующем разделе).
- «Использовать быстрое копирование при передаче данных» - используется для корневого сервера, позволяет включить режим «быстрого копирования» при вставке данных в транзитную базу данных. Работает только для MS SQL 2008 и выше (на всех остальных СУБД приведет к ошибке), вместо select и в зависимости от результата update или insert, использует insert во временную таблицу с последующим merge, что приводит к увеличению скорости изменения данных в 2-4 раза.
- «Передача Syslog» - используется для подчиненного сервера, позволяет включить / выключить фильтрацию / передачу данных таблицы системного лога. Принимаемые значения: «Передавать полностью», «Передавать события кассы» (включается фильтр, отбрасывающий служебные события, события и ошибки синхронизации и т.д.), «Не передавать». Отключение передачи системного лога позволяет сократить передаваемую вверх информацию примерно в два раза, то приводит к потере возможности получать и анализировать централизованно системный лог. На практике, в случае если существует ограничения канала связи чаще используется промежуточный вариант («Передавать события кассы»).
- «Транспорт» - используется для подчиненного сервера, тип канала, по которому будет осуществляться связь с родительским сервером данных.
- «Параметры соединения» - используется для подчиненного сервера, ip-адрес и порт (127.0.0.1:2303) для подключения к родительскому серверу данных (по умолчанию для TCP канала – порт 2303, для HTTP канала - 2305).
- «Использовать SSL» - используется для подчиненного сервера, включение этой настройки приводит к попытке установить зашифрованный канал связи с родительским сервером данных (для того чтобы попытка была успешной, родительский сервер данных должен поддерживать шифрование для канала данного типа).

- «Удостоверение сервера» - имя открытого сертификата канала родительского сервера данных (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается настраиваемый сервер данных (доверенные сертификаты).

Профиль синхронизации с внешней системой:



Параметр	Значение
Число записей в пакете на передачу	1000
Периодичность обмена данными, с	30
Время ожидания ответа, с	120
Пауза на клиенте при возникновении ошибки, с	60
Число попыток возобновления связи	3
Пауза между пакетами, мс	1000
Максимальное число пакетов справочника	0
Размер пакета на прием, КБ	1024

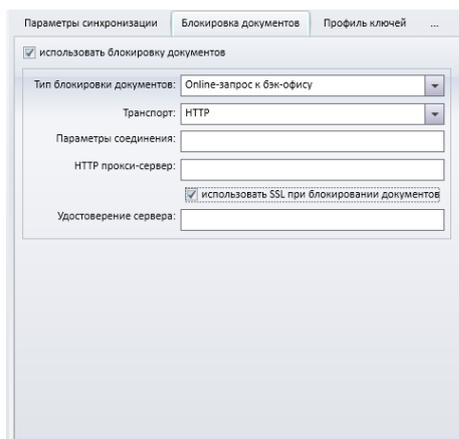
Профиль синхронизации предназначен для оптимизации механизмов передачи данных применительно к типу сервера (корневой / подчиненный), к каналу связи с вышестоящим источником и к производительности компьютеров как настраиваемого сервера, так и вышестоящего источника. Процесс оптимизации – эмпирический, требующий достаточно много времени, поэтому настройкой профиля следует заниматься только в случае реальной необходимости и детально уяснив суть работы механизмов передачи данных и настроек профиля синхронизации. Подробнее настройки профиля описаны в соответствующем разделе.

Подзакладка «Блокировка документов».

При работе с документами на подчиненных серверах менеджера используется механизм «захвата документов во владение» или «блокировки». Запрос на блокировку сервер менеджера адресует своему серверу данных, а то в свою очередь передает этот запрос по цепочке вверх, вплоть до корневого сервера данных. Действия корневого сервера определяются данными настройками.

Есть два типа передачи запроса:

- онлайн запрос (поддерживается только бэк-офисом OpenStore),
 - офлайн-запрос (передается в виде заявки в таблице BACKREQUESTEX и получает ответ через таблицу DOCOWNERBACKRESULTINFO – более подробно механизм блокировки документов описан в документе «Работа с менеджером ОС при использовании стороннего бэкофиса»).
- Если менеджер не используется для работы с документами – данные настройки значения не имеют, однако правильнее все-таки блокировку документов выключить, что очень немного, но экономит системных ресурсов.



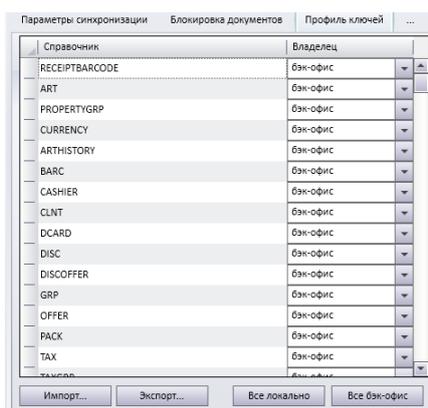
Параметры подзакладки «Блокировка документов»:

- «Использовать блокировку документов» - включает возможность обработки сервером данных запросов на блокировку документов.
- «Тип блокировки документов» - позволяет выбрать один из доступных типов запросов (описаны выше).
- «Транспорт» - используется при онлайнном запросе, тип канала, по которому будет осуществляться связь с бэк-офисом OpenStore.
- «Параметры соединения» - используется при онлайнном запросе, ip-адрес и порт (127.0.0.1:8734) для подключения к серверу бэк-офиса (по умолчанию для TCP канала – порт 8734, для HTTP канала - 8732).
- «HTTP прокси-сервер» - используется при онлайнном запросе, ip-адрес и порт (через двоеточие: 192.168.0.1:3128) прокси-сервера.
- «Использовать SSL» - используется при онлайнном запросе, включение этой настройки приводит к попытке установить зашифрованный канал связи с сервером бэк-офиса (для того чтобы попытка была успешной, сервер бэк-офиса должен поддерживать шифрование для канала данного типа).
- «Удостоверение сервера» - используется при онлайнном запросе, имя открытого сертификата канала сервера бэк-офиса (subject name), установленного в хранилище сертификатов того компьютера, на котором запускается настраиваемый сервер данных (доверенные сертификаты).

Подзакладка «Профиль ключей».

Для создания элементов справочников на подчиненных серверах менеджера используется механизм резервирования идентификаторов. В зависимости от архитектуры системы, источником пулов идентификаторов может быть корневой сервер данных или внешняя система (бэк-офис). Как правило, если внешняя система работает с тем или иным справочником, то она же и является источником идентификаторов для него, а если нет – эта задача перекладывается на корневой сервер

данных. Таким образом в настройках последнего можно указать источник идентификаторов с точностью до справочника.

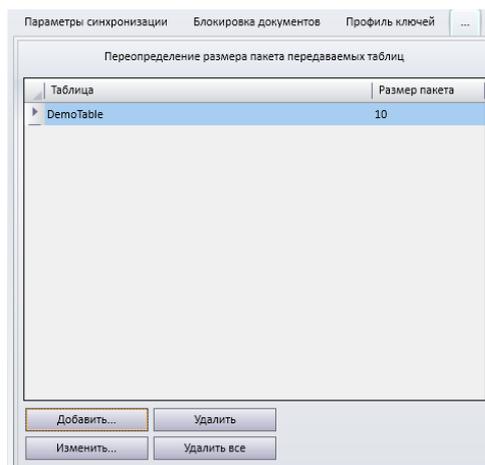


Из выпадающего списка, каждому из справочников можно выбрать одно из доступных значений: «бэк-офис» или «локально» (т.е. корневой сервер данных). Для быстрого переименования настройки для всех справочников используются кнопки «Все бэк-офис» или «Все локально» устанавливающие соответствующее значение для всех справочников сразу.

Для удобства клонирования готового решения профиль первичных ключей (идентификаторов) может быть сохранен в файл и загружен в последствии на другом корневом сервере данных (кнопки «Экспорт» и «Импорт» соответственно).

Подзакладка «...».

В случае использования пользовательских таблиц, поднимающихся вверх и содержащих большие объемы данных (например бинарных) может возникнуть необходимость переопределения размера пакета для этих таблиц (чтобы исключить чрезмерное потребление памяти).



Для этого, эти таблицы следует добавить в список на закладке «...» и указать требуемый размер пакета.

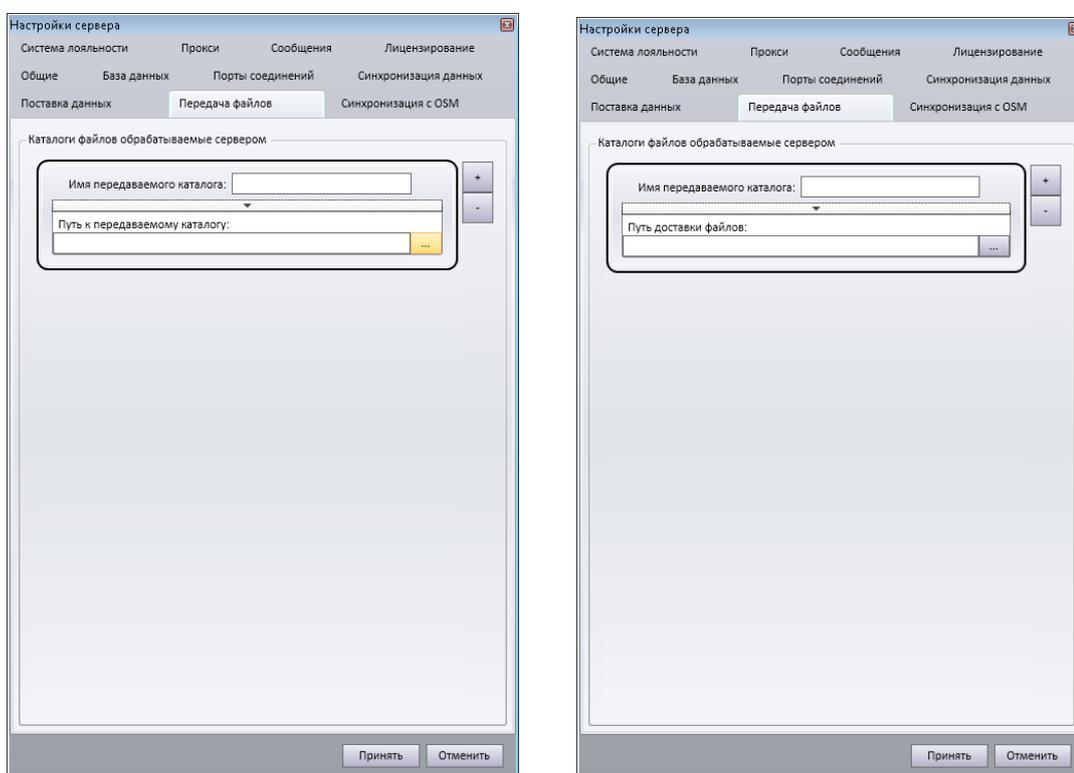
Следует отметить, что после добавления таблицы в список, изменение ее имени невозможно, т.к. последнее является первичным ключом в таблице хранения данной настройки.

Если имя таблицы указано неверно, следует удалить запись и добавить новую с правильным именем таблицы.

Для стандартных таблиц OpenStore переопределение размера пакета как правило не требуется, однако, при необходимости любая из стандартных таблиц может быть использована в данной настройке.

Закладка «Передача файлов».

Наряду с таблицами базы данных, сервер данных может передавать своим клиентам и каталоги файлов. Каталогов может быть несколько. Размер файлов и их количество, а так же уровень вложенности не ограничиваются, однако следует помнить, что передача файла большого объема займет соответствующее время.



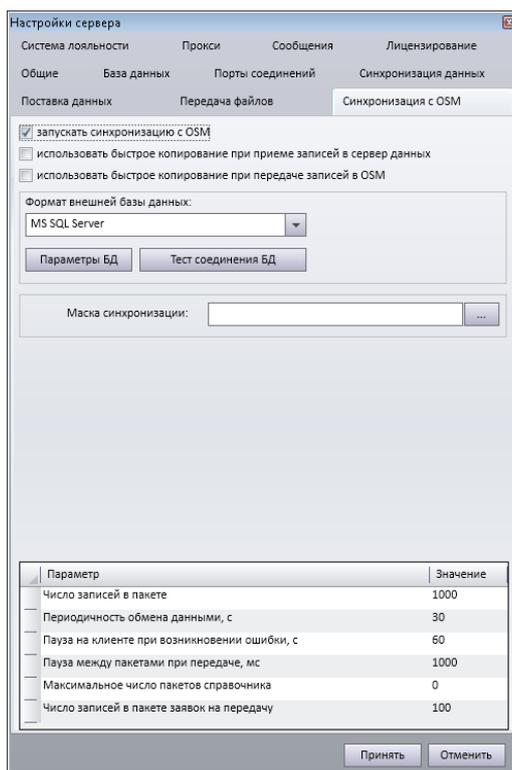
Источником файлов может служить только корневой сервер данных (слева). Подчиненный сервер может выполнять роль передаточного звена и (при необходимости) поставщика файлов по указанному пути (справа).

Для того, чтобы корневой сервер мог стать источником каталога файлов, этот каталог требуется добавить в список обрабатываемых данным сервером. «Имя передаваемого каталога» - относительное имя каталога с файлами, который требуется передавать. «Путь к передаваемому каталогу» - собственно путь до передаваемого каталога (не включая последний; например, если требуется передавать каталог «C:\Temp\MyDir», то в качестве имени следует указать «MyDir», а в качестве пути – «C:\Temp»).

Для того, чтобы подчиненный сервер мог распространять каталог файлов, этот каталог требуется добавить в список обрабатываемых данным сервером. «Имя передаваемого каталога» должно совпадать с одним из каталогов, добавленных в такой же список у вышестоящего сервера (если вышестоящий сервер не содержит каталог, запрашиваемый подчиненным сервером, то при получении данных возникнет соответствующая ошибка). Если требуется, чтобы подчиненный сервер выложил полученный каталог куда-либо, то следует указать соответствующий «Путь доставки файлов».

Закладка «Синхронизация с OSM».

OSM – это сокращение от «OpenStore менеджер», синхронизация с которым состоит из двух независимых процессов. С одной стороны, сервер менеджера является клиентом сервера данных, что определяет механизм получения им данных от внешней системы. Но, т.к. менеджер позволяет создавать и редактировать справочники, существует механизм, при помощи которого сервер данных получает данные от сервера менеджера. Этот механизм во многом аналогичен тому, при помощи которого корневой сервер данных обменивается с транзитной базой данных, только в качестве транзитной базы данных выступает база сервера менеджера. Последнее подразумевает непосредственный доступ сервера данных к базе сервера менеджера, из чего в свою очередь следует, что с одним сервером данных может работать только один сервер менеджера (сервер данных имеет только одно подключение к базе OSM). Никакого сжатия / шифрования во время такого доступа не используется, что само по себе предполагает, что доступ осуществляется по локальной сети.



Параметры закладки «Синхронизация с OSM»:

«Запускать синхронизацию с OSM» - позволяет включить / выключить получение данных от сервера менеджера.

«Использовать быстрое копирование при приеме записей в сервер данных» - позволяет включить режим «быстрого копирования» при вставке данных полученных из базы сервера менеджера.

Работает только для MS SQL 2008 и выше (на всех остальных СУБД приведет к ошибке), вместо select и в зависимости от результата update или insert, использует insert во временную таблицу с последующим merge, что приводит к увеличению скорости изменения данных в 2-4 раза.

«Использовать быстрое копирование при передаче записей в OSM» - позволяет включить режим «быстрого копирования» при вставке данных в базу сервера менеджера. Работает только для MS SQL 2008 и выше (на всех остальных СУБД приведет к ошибке), вместо select и в зависимости от результата update или insert, использует insert во временную таблицу с последующим merge, что приводит к увеличению скорости изменения данных в 2-4 раза.

«Формат внешней базы данных» - позволяет выбрать тип СУБД соответствующий тому, который используется сервером менеджера.

- «Параметры БД» - позволяет указать настройки подключения к БД выбранного типа.
- «Тест соединения» - выполняет пробное подключение в указанной в настройках выше базе данных.
- «Маска синхронизации» - позволяет выбрать используемую маску синхронизации, описывающую сопоставление полей базы данных сервера данных и базы сервера менеджера (как правило сопоставление один к одному, более подробно о маске синхронизации в соответствующем разделе).

Профиль синхронизации с сервером менеджера:

Параметр	Значение
Число записей в пакете	1000
Периодичность обмена данными, с	30
Пауза на клиенте при возникновении ошибки, с	60
Пауза между пакетами при передаче, мс	1000
Максимальное число пакетов справочника	0
Число записей в пакете заявок на передачу	100

Профиль синхронизации предназначен для оптимизации механизмов передачи данных применительно к скорости канала связи с базой сервера менеджера и к производительности компьютеров как настраиваемого сервера, так и базы сервера менеджера. Процесс оптимизации – эмпирический, требующий достаточно много времени, поэтому настройкой профиля следует заниматься только в случае реальной необходимости и детально уяснив суть работы механизмов передачи данных (раздел «Технологические особенности работы сервера») и настроек профиля синхронизации (раздел «Параметры профиля синхронизации»).

Параметры профиля синхронизации.

Как ранее было указано, сервер данных может быть корневым или подчиненным, что определяет механизм обмена данными с вышестоящим элементом системы (вышестоящий сервер данных или транзитная база). В первом случае получатель (сервер данных) выступает в роли клиента, а источник (вышестоящий элемент системы) – в роли сервера (DATASYNC). Во втором, в качестве источника (пассивного) выступает СУБД транзитной базы данных, а все операции по синхронизации выполняются непосредственно сервером данных (DATAPUMP). Более подробно работа механизмов обмена данными описана в разделе «Технологические особенности работы сервера».

- «Время ожидания ответа, с» - время (в секундах), в течение которого получатель ждет ответ от источника (клиент от сервера для DATASYNC, или сервер данных от СУБД транзитной базы для DATAPUMP), запросив данные по конкретному справочнику. В течение этого времени происходит соответствующая протоколу подготовка данных. Чем больше размер справочников и чем медленнее компьютер источника тем больше времени требуется на подготовку данных для получателя, и как следствие – большее значение таймаута ожидания ответа следует устанавливать. Однако слишком большой таймаут приведет к тому, что получатель после кратковременной потери связи с источником будет долго находится в ожидании ответа и как следствие – дольше не выйдет на новый цикл (не обнаружит, что связь с источником уже восстановлена).
- «Периодичность обмена данными, с» - время (в секундах) по истечении которого получатель повторит цикл связи с источником (клиент с сервером для DATASYNC или сервер данных с СУБД транзитной базы для DATAPUMP). Кроме того, если клиент не вышел на связь с сервером (например в результате выполнения запроса к своей базе данных при передаче данных от клиента к серверу) в течение трех периодов обмена данными, связь с клиентом считается потерянной и регистрация клиента автоматически удаляется сервером, поэтому слишком маленькое значение этой настройки может привести к невозможности обмена данными между клиентом и сервером (только для DATASYNC).
- «Число записей в пакете на передачу» - только для DATASYNC, количество записей в пакете при передаче данных от клиента к серверу позволяет оптимизировать скорость передачи данных применительно к аппаратной части и к скорости и стабильности канала связи. Так с одной стороны, увеличение размера пакета приводит к уменьшению количества запросов к локальной базе данных клиента (требуемый объем будет передан меньшим числом пакетов), но с другой – требует большего объема памяти как на клиенте, так и на сервере (для сервера, учитывая число одновременно обслуживаемых клиентов это

может стать особенно критичным). Кроме того в случае возникновения ошибки в канале связи при передаче пакета, заново передается весь пакет. Таким образом чрезмерное увеличение размера пакета при нестабильном канале связи кроме повышения требуемых ресурсов компьютера, приведет к увеличению времени передачи пакета, и как следствие, к повышению вероятности ошибки во время передачи пакета, что в конце концов существенно замедлит (один и тот же пакет будет передаваться много раз) или сделает невозможным (пакет из-за своего размера и нестабильности канала не сможет передаться) передачу данных от клиента к серверу.

- «Допустимая разница во времени между клиентом и сервером, с» - только для DATASYNC, если разница времени клиента и сервера окажется больше чем указанное в данной настройке значение, то выполнится автоматическая синхронизация времени клиента, на основании времени сервера. Этот механизм позволяет достаточно просто поддерживать единое системное время во всей системе, сколь сложной бы она не была без использования каких-либо сторонних средств. Процесс запроса и установки нового времени на клиенте занимает некоторое время (до 10-20 секунд), поэтому слишком малое значение разброса времени сервера и клиента может привести к изменению времени на клиенте на каждом цикле синхронизации. Во избежание такой проблемы не следует указывать значение этой настройки менее 60 секунд.
- «Размер пакета на прием, КБ» - только для DATASYNC, данные передаваемые сверху вниз, после запроса клиента выбираются из базы сервера и сохраняются в промежуточные файлы (пакеты), каждый из которых представляет собой таблицу с данными. Каждый файл сжимается алгоритмом ZIP. Далее клиент принимает эти файлы. Если во время приема файла возникает ошибка, то выполняется повторный прием файла с того места, на котором произошел обрыв связи (количество попыток до принять файл определяются настройкой «Число попыток возобновления связи»). Если связь не восстановилась, то прием файла начинается сначала на очередном цикле синхронизации. После приема всех файлов, клиент их распаковывает по очереди, воссоздает таблицу и синхронизирует ее со своей базой данных. Данная настройка позволяет указать размер таких файлов. Чем меньше этот размер, тем выше вероятность передачи файла по тонкому и нестабильному каналу с первого раза, однако больше число файлов, что увеличивает время, расходуемое на их создание и распаковку. Слишком большей размер потребует больше оперативной памяти как на сервере (причем для каждого из обслуживаемых клиентов), так и на клиенте.
- «Пауза на клиенте при возникновении ошибки» - при возникновении ошибки при передаче данных в лог пишется соответствующее сообщение. Если значение настройки «Периодичность обмена данными» не велико, а ошибка не устраняется (не

восстанавливается связь или целостность данных и т.п.) длительное время, то в логе окажется очень много однотипных сообщений. Для сокращения их количества предназначена данная настройка. При ошибке синхронизации следующая попытка будет выполнена не через время, указанное в «Периодичность обмена данными» а через время, указанное в данной настройке.

- «Число попыток возобновления связи» - только для DATASYNC, количество попыток возобновить связь с сервером и продолжить принимать текущий файл с места обрыва связи (подробнее в описании настройки «Размер пакета на прием, КБ»).
- «Максимальное число пакетов справочника» - число переданных пакетов, по достижении которого цикл передачи будет считаться законченным, даже если не все данные переданы, если «0» - число пакетов не ограничено. При передаче данных наверх, завершение цикла передачи по умолчанию наступает когда все данные переданы. Однако при интенсивном потоке данных (большом количестве касс и их интенсивной работе) такая ситуация может не наступать достаточно долго (за время передачи пакета будут успевать приходить новые данные), что не позволит перейти к приему данных. Для разрешения вышеуказанной проблемы используется данная настройка.
- «Время ожидания ответа при запросе лицензии, с» - время, в течение которого сервер данных ожидает от вышестоящего сервера данных или сервера лицензирования (в зависимости от настроек «Закладка «Лицензирование»») ответа на прямой или перенаправленный запрос на получение лицензии. При использовании большего количества подчиненных серверов данных и низкой производительности компьютера сервера лицензирования или канала к последнему, может потребоваться увеличение времени ожидания ответа на запрос лицензии, для чего и служит данная настройка.
- «Время ожидания ответа при блокировке документов, с» - время, в течение которого сервер данных ожидает от вышестоящего сервера данных или бэк-офиса ответа на перенаправленный запрос на блокировку документов. При использовании большего количества подчиненных менеджеров, а следовательно и серверов данных, может потребоваться увеличение времени ожидания ответа на запрос получения документа во владение, для чего и служит данная настройка.
- «Число записей в пакете» - только для DATAPUMP, максимальное число записей в пакете обмена данными с транзитной базой данных (подробнее в разделе DATAPUMP).
- «Пауза между пакетами, мс» - задержка в миллисекундах перед вычиткой данных для следующего пакета на передачу. В случае, если приток данных на передачу настолько велик, что за время передачи пакета появляются новые данные, но количество этих данных существенно меньше размера пакета, может быть полезным ввести задержку между

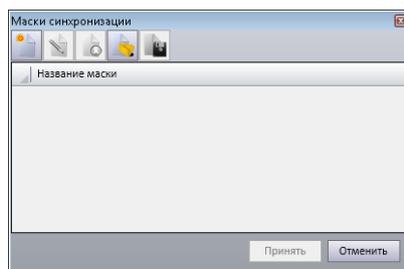
пакетами, что позволит накапливать большее количество данных для пакета, сокращая тем самым количество запросов на вычитку данных, а следовательно и нагрузку на СУБД, с которой работает сервер данных.

- «Число записей в пакете заявок на передачу» - заявка, в отличие от обычных таблиц, передаваемых наверх (SALES, SYSLOG и т.д.) представляет собой элемент справочника, а следовательно может иметь достаточно большой объем, что в совокупности с большим количеством записей может вызвать чрезмерное потребление оперативной памяти. Поэтому, для заявок размер передаваемого пакета указывается отдельной настройкой.

Маска синхронизации.

Транзитная база данных по своей структуре является посредником между внешней системой и сервером данных. Ее структура может быть близка к структуре базы сервера данных (вплоть до совпадения) или существенно от нее отличаться (если она сформирована например вьюхами на основании таблиц внешней системы). Маска синхронизации предназначена для того, чтобы согласовать эти структуры, описывая, из какой таблицы транзитной базы получает данные конкретная таблица сервера данных, и какие из таблиц сервера данных передаются в транзитную базу данных и в какие именно ее таблицы. При этом несколько таблиц сервера данных могут заполняться на основании одной таблицы транзитной базы, что позволяет упростить структуру последней по отношению к базе сервера данных. Кроме того, маска синхронизации предоставляет возможность заполнять ряд таблиц сервера данных «фиксированными данными» (жестко введенными в маске), что в свою очередь позволяет добиваться совместимости с ранее существовавшей транзитной базой в случае изменения структуры базы сервера данных на этапе обновления комплекса OpenStore.

Редактор маски синхронизации встроен в консоль управления сервером данных, что позволяет быстро и при необходимости удаленно создать / изменить маску синхронизации. Для вызова редактора маски достаточно на закладке настроек «Закладка «Поставка данных»» вызвать список доступных масок (форма «Маски синхронизации»):



Кнопки управления формы «Маски синхронизации»:



- создать маску, позволяет создать новую маску, требующую заполнения;



- редактировать маску, позволяет открыть выбранную маску на редактирование;



- удалить маску, позволяет удалить маску из базы сервера данных;

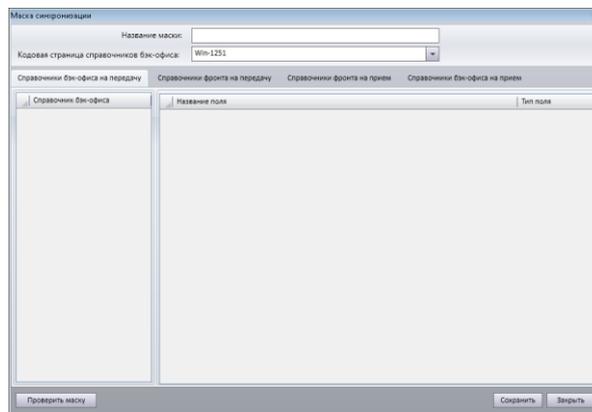


- загрузить маску из файла, позволяет загрузить ранее созданную и настроенную маску (позволяет легко реплицировать типовое решение), в т.ч. и маску из стандартной поставку;



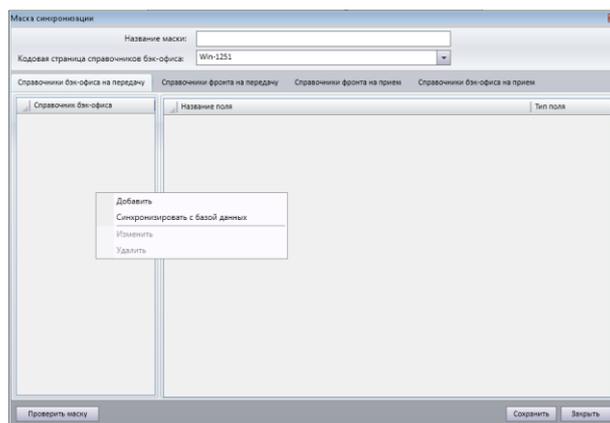
- сохранить маску в файл, позволяет экспортировать созданную маску для последующего импорта при репликации типового решения.

Для начала рассмотрим создание новой маски как наиболее сложный процесс, охватывающий все нюансы работы с масками синхронизации. При нажатии на кнопку «Создать маску» откроется редактор маски:

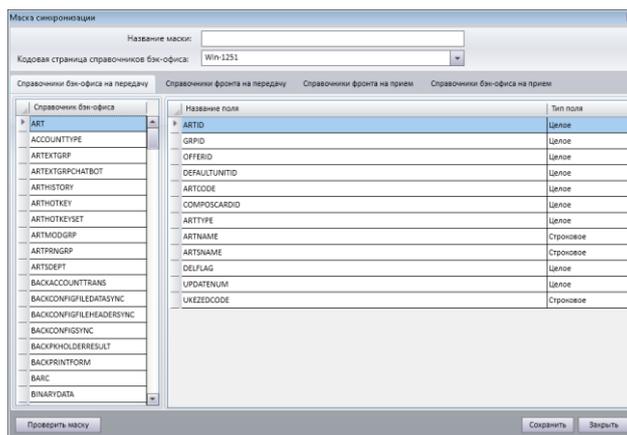


В поле «Название маски» следует указать логическое имя маски, предназначенное для того, чтобы отличить одну маску от другой. В случае, если строковые данные в транзитной базе хранятся в кодировке 866 (в настоящее время используется редко), требуется указать эту кодировку в маске синхронизации.

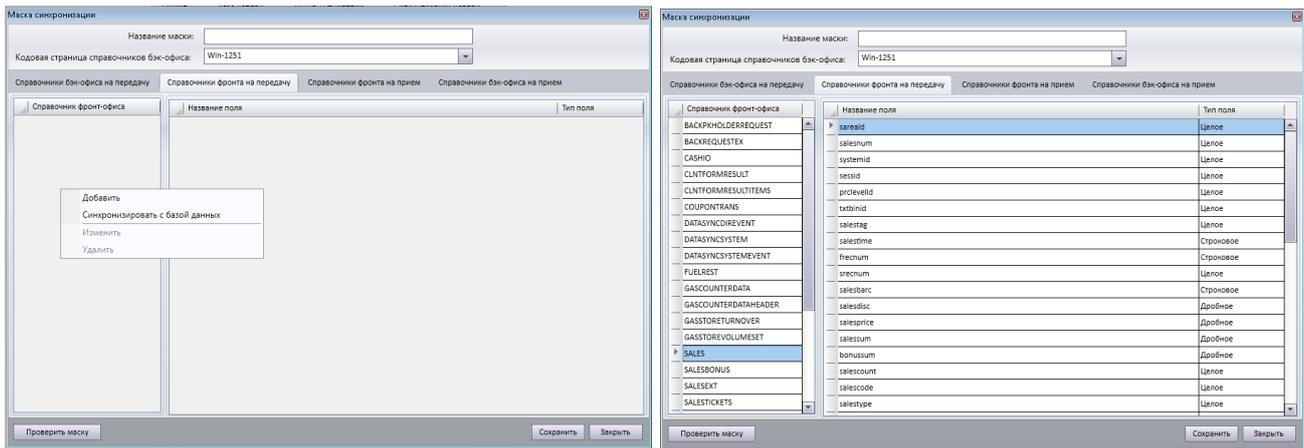
На закладке «Справочники бэк-офиса на передачу» требуется перечислить таблицы транзитной базы, передаваемые сверху вниз (из транзитной базы в сервер данных) и описать их структуры. Если транзитная база уже существует, а подключение к ней настроено и таблица DATAPUMP заполнена, то процесс заполнения списка таблиц можно выполнить автоматически. Для этого достаточно правой кнопкой мыши кликнуть на столбце «Справочник бэк-офиса»:



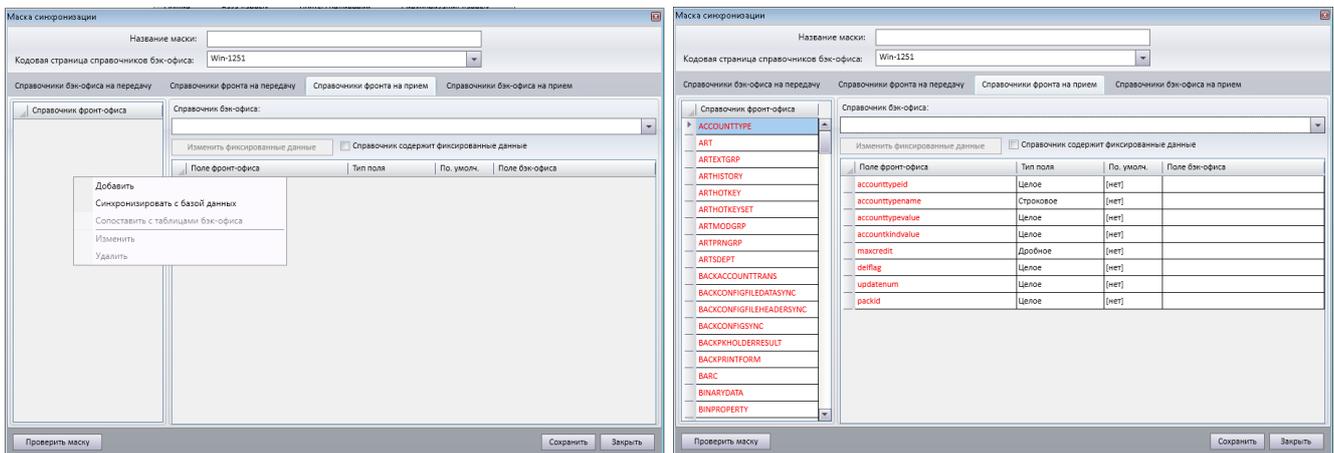
В открывшемся меню выбираем «Синхронизировать с базой данных». В случае, если вышеописанные условия (наличие транзитной базы, наличие в ней заполненной таблицы DATAPUMP и т.д.) по каким-то причинам выполнить не удалось, то придется, выбрав «Добавить» вручную указывать имя таблицы, а затем так же вручную описывать ее структуру (не рекомендуется, т.к. как правило приводит к большому количеству ошибок, обусловленному человеческим фактором). В любом случае закладка «Справочники бэк-офиса на передачу» должна быть заполнена таблицами транзитной базы, передаваемыми сверху вниз:



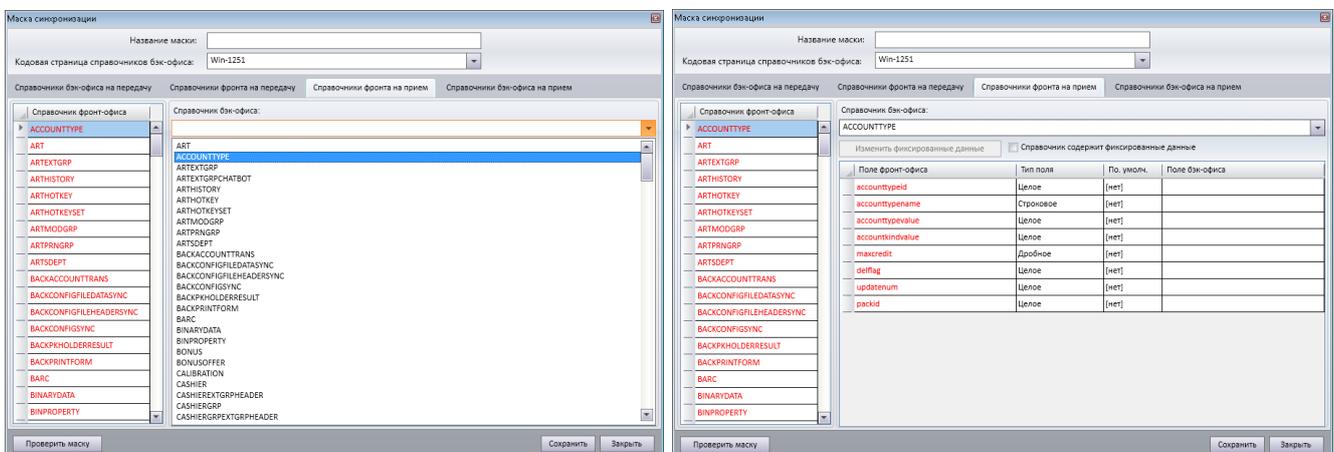
Переходим на закладку «Справочники фронта на передачу». На этой закладке нужно создать список таблиц, которые может передать фронт-офис в транзитную базу, и их структур. Т.к. передаваемые фронт-офисом таблицы однозначно определены в таблице DATAPUMPDIRS базы сервера данных то создавать их вручную нет необходимости (не смотря на то, что такая возможность существует). Для заполнения данных на этой закладке достаточно правой кнопкой мыши кликнуть на столбце «Справочник фронт-офиса» и выбрать пункт меню «Синхронизировать с базой данных»:



Переходим на закладку «Справочники фронта на прием». Заполнить данные здесь, как и в предыдущем случае можно (рекомендуется) автоматически, т.к. все принимаемые таблицы однозначно определены в таблице DATAPUMPDIRS базы сервера данных:

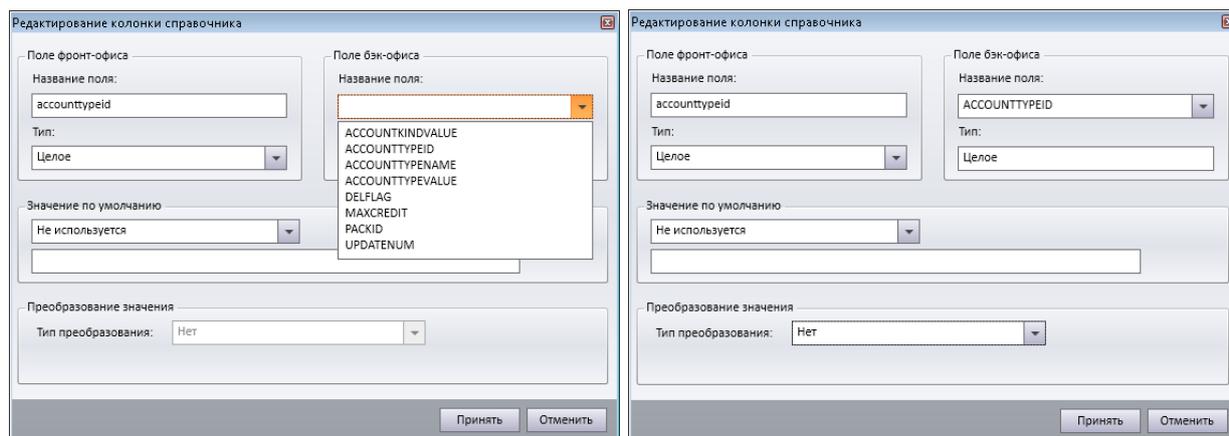


После заполнения списка таблиц и их структур нужно выполнить сопоставление таблиц фронт-офиса с таблицам транзитной базы. Для этого каждой таблице фронт-офиса выбираем соответствие из выпадающего списка «Справочник бэк-офиса»:



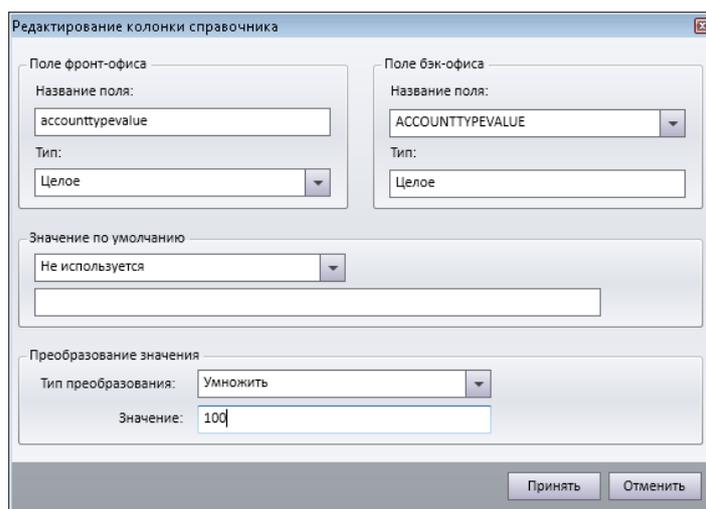
Каждому полю выбранной таблицы фронт-офиса выбираем соответствующее поле сопоставленной таблицы транзитной базы из выпадающего списка «Название поля» (двойной

щелчок левой кнопки мыши по полю или щелчок правой кнопки мыши и выбор пункта меню «Изменить»):



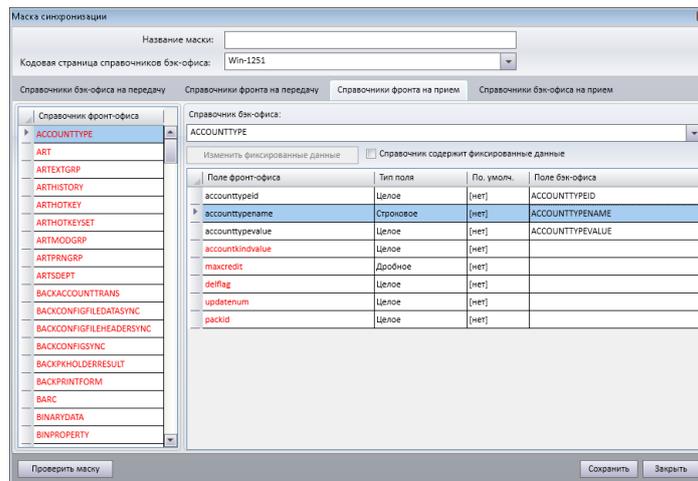
Если соответствующая таблица транзитной базы не содержит поля, которое можно / нужно сопоставить заполняемому полю таблицы сервера данных, то его можно заполнить фиксированными данными, выбрав соответствующее значение из выпадающего списка «Значение по умолчанию» («Пустое значение» - если поле может быть NULL, «Константа» или «Дата синхронизации» - в этом случае поле будет заполняться значением даты / времени на момент цикла синхронизации в формате OpenStore).

При необходимости значение поля (если его тип - целое) можно умножить или разделить на постоянное число. Для этого следует выбрать соответствующий «Тип преобразования» и указать «Значение» (множитель / делитель):

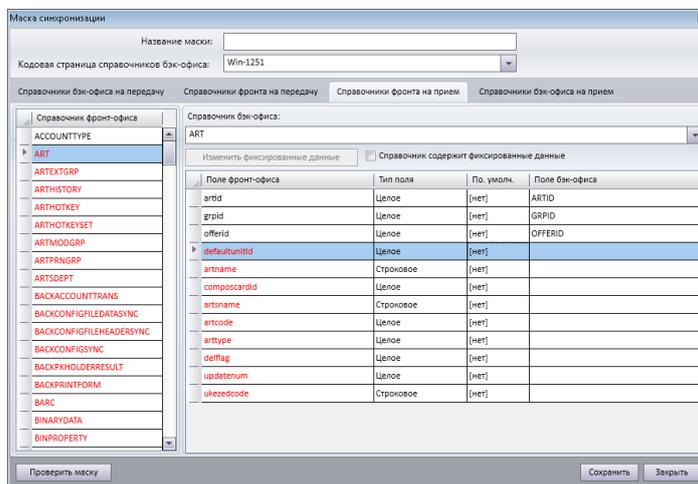


Такое преобразование может оказаться полезным для пересчета суммы / цены и т.д., а так же в некоторых случаях для обеспечения уникальности идентификаторов (в последнем случае решение следует принимать крайне взвешенно, т.к. это может привести к нежелательным последствиям).

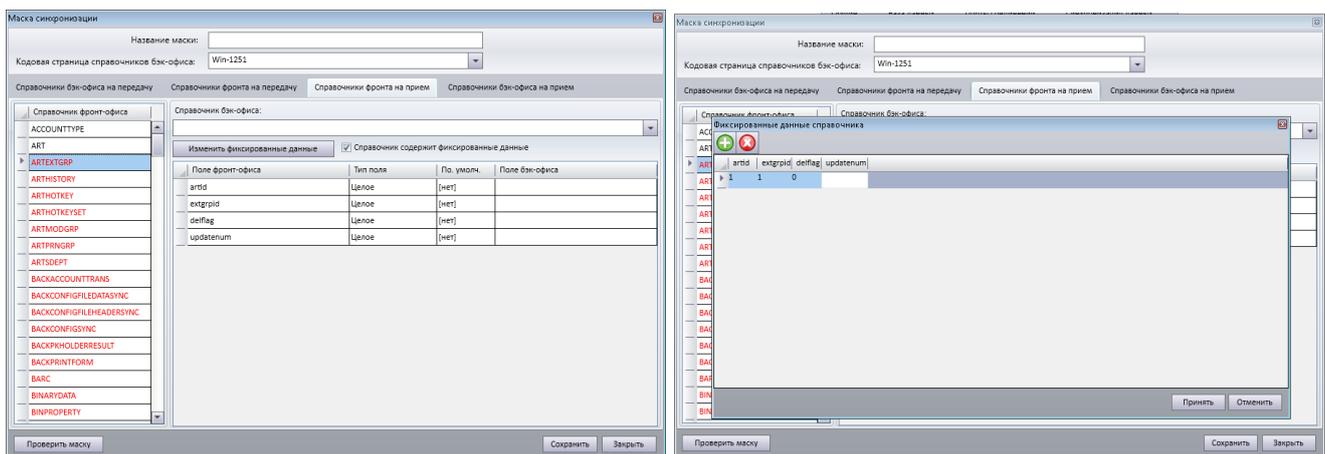
Для удобства все сопоставленные поля таблицы окрашиваются черным цветом, а не сопоставленные – красным:



Точно так же ведут себя и сопоставленные / не сопоставленные таблицы:



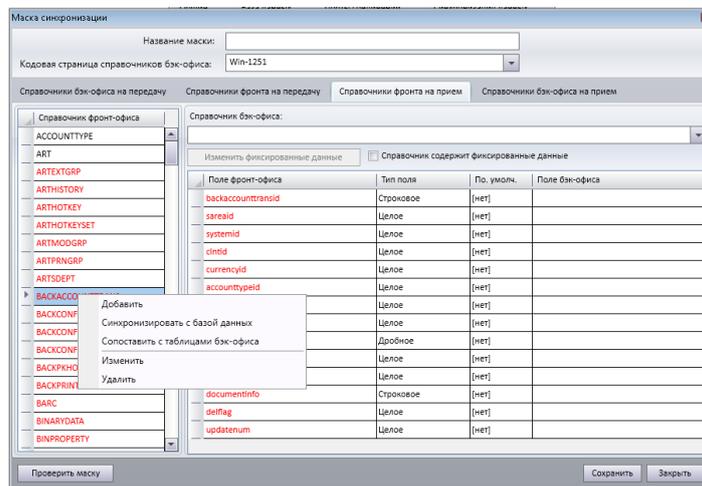
Если транзитная база не содержит таблицы, которую нужно сопоставить конкретной таблице сервера данных, последняя может быть заполнена «фиксированными данными». Для этого в настройках данной таблицы требуется установить флаг «Справочник содержит фиксированные данные» и при необходимости создать / изменить эти данные используя редактор вызываемый кнопкой «Изменить фиксированные данные»:



Таблицы помеченные флагом «Справочник содержит фиксированные данные» сохраняют красный цвет, в то время, как их поля окрашиваются черным. Это позволяет быстро обнаружить такие таблицы и при необходимости внести коррективы в маску синхронизации.

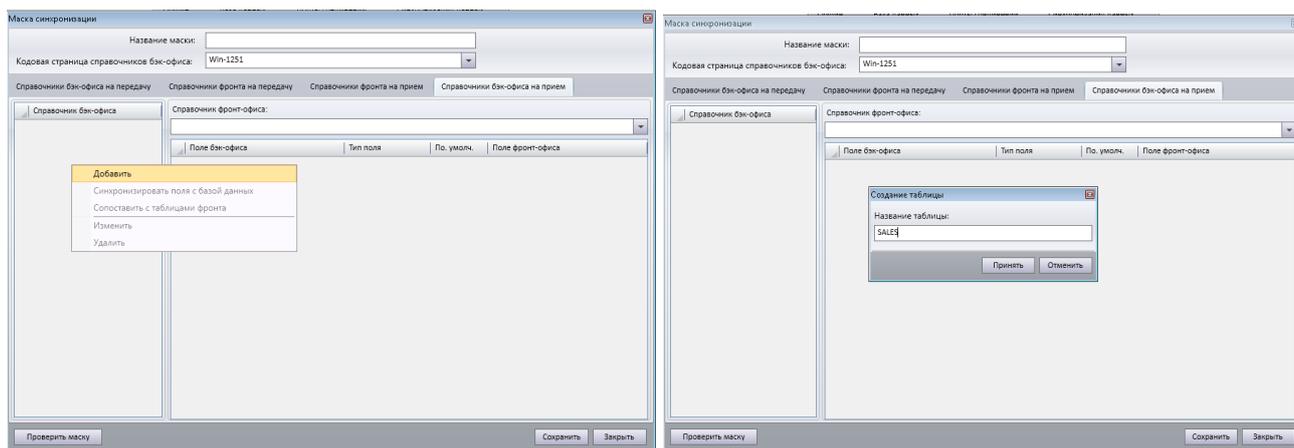
Используя вышеописанные инструменты, необходимо сопоставить все таблицы принимаемые фронт-офисом таблицам транзитной базы или фиксированным значениям, т.к. в противном случае корректная синхронизация сервера окажется невозможной.

В случае, если таблицы транзитной базы совпадают по имени с таблицами базы сервера данных, существует возможность выполнить автоматическое сопоставление таблиц и полей. Для этого достаточно правой кнопкой мыши кликнуть на списке «Справочник бэк-офиса» и выбрать пункт меню «Сопоставить с таблицами бэк-офиса»:

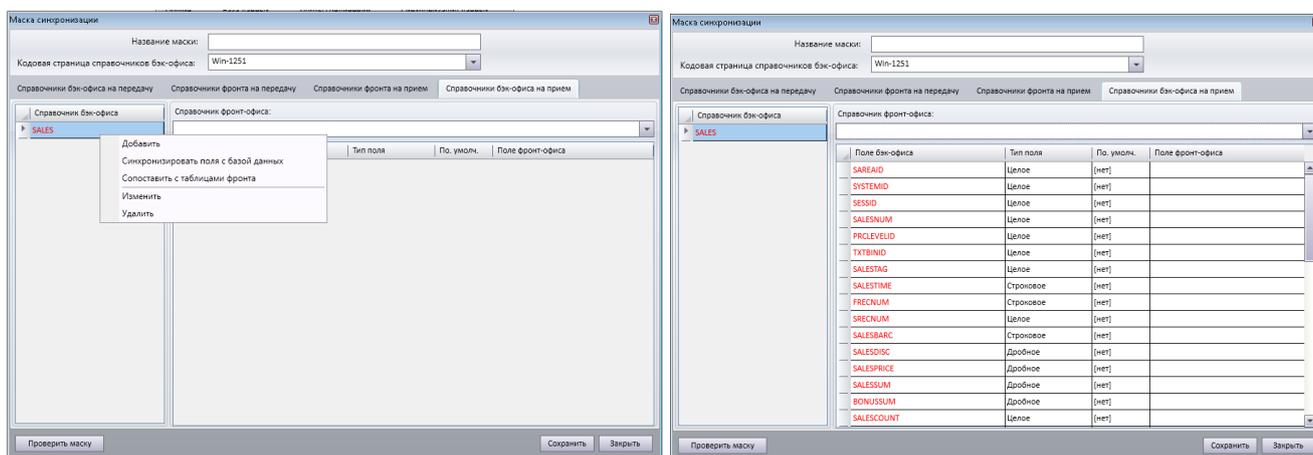


Однако следует помнить, что автоматически будут сопоставлены только одноименные таблицы, для которых будут сопоставлены только одноименные поля. Эту особенность следует учитывать при проектировании транзитной базы данных.

Переходим на закладку «Справочники бэк-офиса на прием». Так как имена таблиц, в которые сервер передает данные нигде в явном виде не указаны, автоматическое заполнение списка этих таблиц невозможно. Для добавления каждой из таких таблиц требуется правой кнопкой мыши кликнуть на списке «Справочник бэк-офиса» и выбрать пункт меню «Добавить»:



В открывшейся форме указываем имя таблицы транзитной базы данных, которую требуется заполнять. Если транзитная база существует и подключение к ней настроено, то можно (рекомендуется) сформировать структуру таблицы автоматически. Для этого достаточно правой кнопкой мыши кликнуть на ней и выбрать пункт меню «Синхронизировать поля с базой данных»:



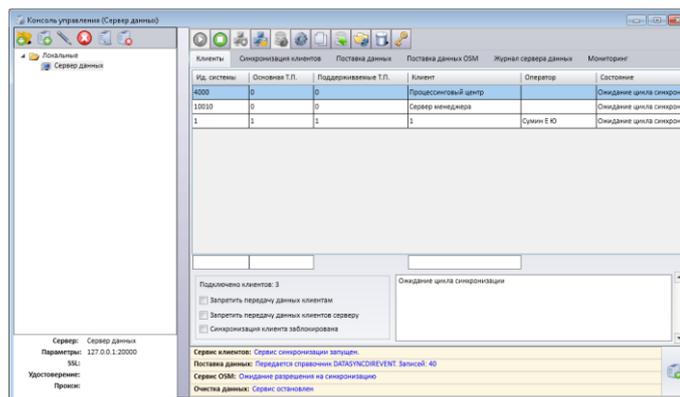
Далее требуется сопоставить эту таблицу с таблицей фронт-офиса, а ее поля – с соответствующими полями таблицы фронт-офиса (механизм сопоставления аналогичен описанному выше).

После заполнения маски синхронизации рекомендуется сделать проверку. Для этого нужно нажать кнопку «Проверить маску». Результатом такого действия станет форма со списком существующих в маске ошибок или сообщение об их отсутствии:

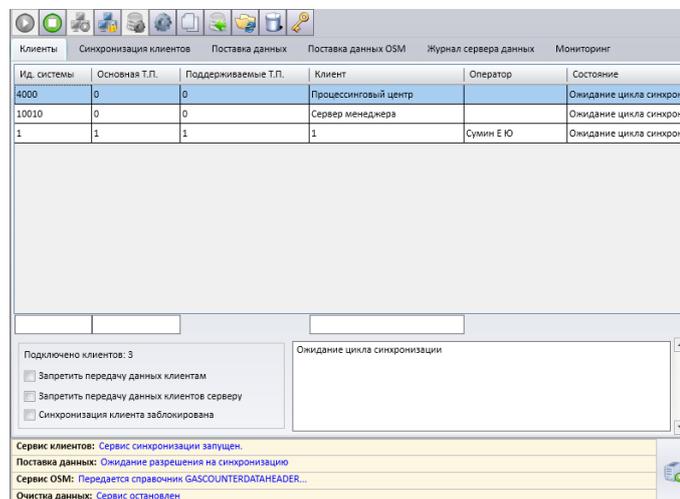
Управление работой

В нормальных условиях сервер данных работает в автоматическом режиме и в дополнительном управлении не нуждается. Однако для предупреждения возникновения внештатных ситуаций, своевременного обнаружения последних или для борьбы с их последствиями может потребоваться вмешательство администратора.

Основным инструментом для управления сервером данных является «Консоль управления».



В левой части окна расположен список всех доступных серверов (создание подключения к серверу данных описано в разделе «Установка»), что позволяет легко переключаться между последними. В правой части отображается информация о выбранном сервере.



Снизу (на светло-желтом фоне) отображается текущее состояние сервера (какие именно операции он выполняет в настоящий момент), а иконка  индицирует отсутствие / наличие ошибок в его работе ( или ). Так, если при работе сервера данных возникает ошибка (нет связи с вышестоящим сервером, нет связи с транзитной базой данных или базой менеджера и т.д.), вышеописанная иконка предупредит об этом администратора. Подведя указатель мыши к иконке, можно посмотреть текст ошибки, а выполнив двойной клик по ней – сбросить состояние ошибки.

В верхней части расположены кнопки управления и закладки содержащие логи и инструменты для более детального анализа проблем, возникших у сервера данных и их устранения.

Закладки основного окна

Закладка «Клиенты».

На закладке «Клиенты» отображается список всех активных клиентов выбранного сервера данных, а так же их текущее состояние.

Клиенты	Синхронизация клиентов	Поставка данных	Поставка данных OSM	Журнал сервера данных	Мониторинг
Ид. системы	Основная Т.П.	Поддерживаемые Т.П.	Клиент	Оператор	Состояние
1	1	1	1	Сулмин Е Ю	Ожидание цикла синхронизации
10010	0	0	Сервер менеджера		Ожидание цикла синхронизации
4000	0	0	Процессинговый центр		Ожидание цикла синхронизации

Подключено клиентов: 3

Запретить передачу данных клиентам

Запретить передачу данных клиентам серверу

Синхронизация клиента заблокирована

Для удобства поиска интересующего клиента или группы клиентов предусмотрен интерактивный фильтр.

Параметры закладки «Клиенты»:

- «Запретить передачу данных клиентам» - позволяет оперативно запретить передачу данных клиентам, может оказаться полезным при предотвращении распространения некорректных данных, выгруженных внешней системой, в плоть до исправления этих данных или для ускорения получения передаваемых клиентами данных после длительного отсутствия канала связи между последними и сервером.
- «Запретить передачу данных клиентов серверу» - позволяет оперативно прервать передачу данных от клиентов к серверу, что может быть полезным при необходимости срочно передать клиентам новые данные после длительного отсутствия канала между последними и сервером (по умолчанию клиенты сначала постараются передать данные, и только потом получать данные от сервера).
- «Синхронизация клиента заблокирована» - позволяет оперативно заблокировать обмен данными между сервером и выбранным клиентом (в обе стороны).

Закладка «Синхронизация клиентов».

На закладке «Синхронизация клиентов» можно просматривать лог событий синхронизации клиентов с сервером данных.

Дата/Время	Ид. системы	Т.П.	Описание события
30.11.2020 11:40:01	1000	0	4000#0#Справочник SYSLOG принят. Время: 00:00:01 Записей: 4
30.11.2020 11:40:00	1000	0	4000#0#Принимается справочник SYSLOG...
30.11.2020 11:39:31	1000	0	10010#0#Справочник SYSLOG принят. Время: 00:00:01 Записей: 4
30.11.2020 11:39:30	1000	0	10010#0#Принимается справочник SYSLOG...
30.11.2020 11:39:30	1000	0	4000#0#Справочник CLNT передан. Время: 00:00:00 Записей: 1
30.11.2020 11:39:29	1000	0	4000#0#Передается справочник CLNT...
30.11.2020 11:39:23	1000	0	1#1#Справочник CLNT передан. Время: 00:00:00 Записей: 1
30.11.2020 11:39:22	1000	0	1#1#Передается справочник CLNT...
30.11.2020 11:39:20	1000	0	10010#0#Справочник CLNT передан. Время: 00:00:00 Записей: 1

Фильтр: 4000#0#Справочник SYSLOG принят. Время: 00:00:01 Записей: 4

Торговая площадка: 0
Ид. системы: 1000
Показать последние: 60 мин
Обновить

Для удобства работы с логом предусмотрен фильтр в т.ч. и по описанию события, каждое из которых состоит из: [идентификатор системы]#[идентификатор торговой площадки]#[Текст сообщения].

Закладка «Поставка данных».

На закладке поставка данных можно просматривать лог событий синхронизации с вышестоящим сервером данных (в случае подчиненного сервера) или транзитной базой (для корневого сервера).

Дата/Время	Описание события
30.11.2020 11:39:04	Передан справочник SYSLOG. Время: 00:00:01 Записей: 26
30.11.2020 11:39:02	Передается справочник SYSLOG...
30.11.2020 11:38:56	Передан справочник SALESEXT. Время: 00:00:01 Записей: 31
30.11.2020 11:38:55	Передается справочник SALESEXT...
30.11.2020 11:38:55	Передан справочник SALES. Время: 00:00:01 Записей: 3
30.11.2020 11:38:54	Передается справочник SALES...
30.11.2020 11:36:58	Принят справочник CLNT. Время: 00:00:00 Новых записей: 1
30.11.2020 11:36:58	Принимается справочник CLNT...
30.11.2020 11:35:59	Принят справочник CLNTFORMPROPERTY. Время: 00:00:00 Новых записей: 0
30.11.2020 11:35:59	Принимается справочник CLNTFORMPROPERTY...
30.11.2020 11:22:22	Сервис синхронизации запущен
30.11.2020 11:22:22	Запуск сервиса синхронизации...
30.11.2020 11:22:06	Сервис синхронизации остановлен
30.11.2020 11:22:06	Останавливаю сервис синхронизации...

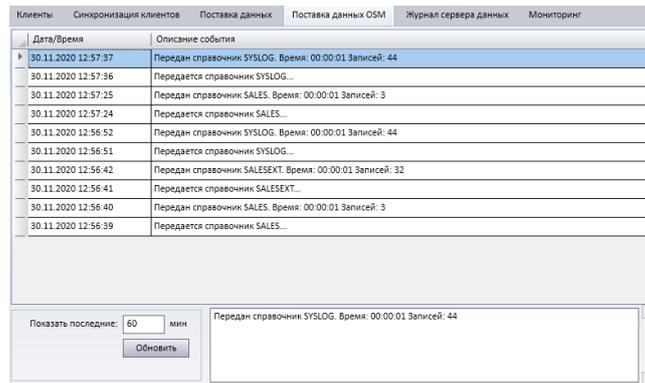
Показать последние: 60 мин
Обновить
 Передавать только справочники продаж

Передан справочник SYSLOG. Время: 00:00:01 Записей: 26

Кроме того, используя настройку «Передавать только справочники продаж», можно приостановить передачу данных таблицы SYSLOG на вышестоящий сервер данных или в транзитную базу данных, что может быть полезным при необходимости ускорить передачу данных о продажах после длительного отсутствия канала связи с вышестоящим сервером / транзитной базой данных.

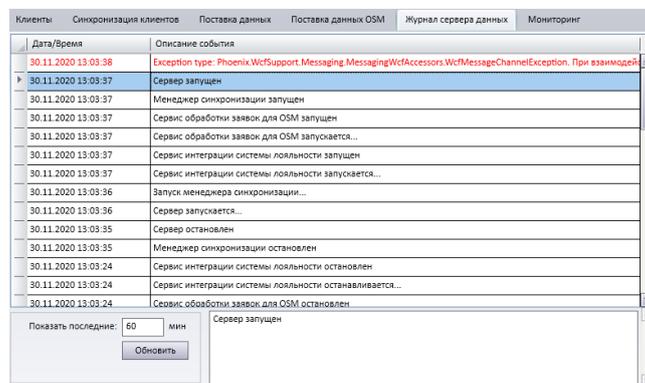
Закладка «Поставка данных с OSM».

Закладка «Поставка данных с OSM» позволяет просматривать лог событий обмена данными с базой данных сервера менеджера (OSM – OpenStore Manager).



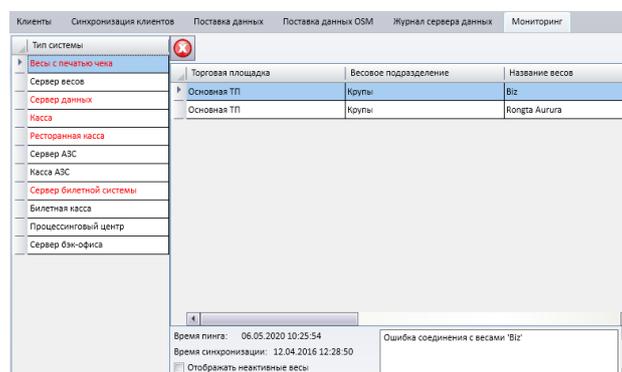
Закладка «Журнал сервера данных».

Закладка «Журнал сервера данных» позволяет просматривать события лога непосредственно сервера данных (запуск / остановка, проблема с обращением к серверу лицензирования, работа сервиса интеграции с лояльностью и т.д.).



Закладка «Мониторинг».

Закладка «Мониторинг» предназначена для наблюдения за состоянием подчиненной серверу данных частью дерева (всего дерева для корневого сервера данных).



В левой части окна расположен список типов систем. Если для конкретного типа систем существуют клиенты (как непосредственно этого сервера данных, так и всех его дочерних серверов данных), в работе которых возникли ошибки, он выделяется красным цветом. При

выборе такого типа системы, в списке слева можно увидеть проблемных клиентов, а так же описание возникших ошибок.

Данные на закладке «Мониторинг», в отличии от остальных вкладок, обновляются автоматически. Это позволяет одним взглядом оценить текущее состояние всей системы OpenStore, сколь сложной бы она не была. Следует так же отметить, что кроме проблем с синхронизацией данных, мониторинг позволяет так же отслеживать:

- отсутствие связи фискального регистратора с эквайром,
- чрезмерный размер базы данных того или иного приложения OpenStore,
- есть возможность создавать пользовательские расширения, позволяющие отслеживать требуемые параметры / состояния.

Если после возникновения ошибки произойдет действие, аннулирующее ошибку (например после пропадания связи с клиентом или весами она восстановилась), этот клиент исчезнет из списка проблемных клиентов автоматически. Однако если восстановить работу клиента не представляется возможным (например компьютер кассы безвозвратно потерян), этого клиента

можно удалить вручную при помощи кнопки .

Для того, чтобы работала система мониторинга для всех подчиненных серверов данных требуется включить настройку «Собирать статистику по клиентам» (подробнее в разделе «Настройки». «Закладка «Синхронизация данных»»).

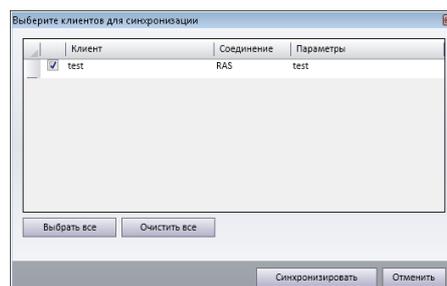
Мониторить состояние системы OpenStore можно так же сторонним приложением, работающим с таблицами DATASYNCSYSTEM, DATASYNCSYSTEMEVENT, SCALEACTIVITY. Подробнее эти таблицы описаны в документе «Синхронизация с внешней системой».

Кнопки управления основного окна



- эта кнопка позволяет принудительно (не дожидаясь расписания) запустить синхронизацию с одним или несколькими клиентами. Эта функция активна только если у сервера данных включен тип синхронизации «по расписанию» (подробнее в разделе «Настройки». «Закладка «Синхронизация данных»»).

При нажатии на вышеуказанную кнопку открывается форма:

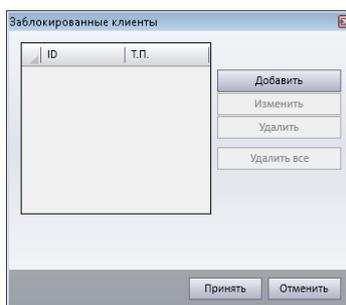


В открывшемся списке требуется выбрать тех клиентов, соединение с которыми должно быть инициировано немедленно. После нажатия на кнопку «Синхронизировать» сервер начнет выполнять синхронизацию с указанными клиентами.



- эта кнопка предназначена для управления блокировкой / разблокировкой синхронизации клиентов. Эта функция требуется когда требуется снять / поставить блокировку для одного или нескольких клиентов, даже если эти клиенты в настоящий момент не находятся на связи. Кроме того может оказаться полезной возможность увидеть список всех заблокированных клиентов.

При нажатии на вышеуказанную кнопку открывается форма:



Для блокировки синхронизации клиента требуется знать и указать идентификатор системы клиента (номер кассы) и его торговую площадку (если клиент поддерживает несколько торговых площадок – первую в списке).

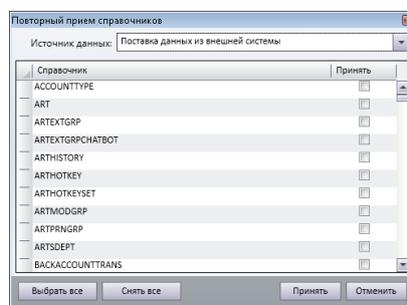


- эта кнопка позволяет принудительно начать новый файловый лог.



- эта кнопка позволяет повторно принять некоторые справочники от вышестоящего источника. Это иногда требуется для решения технических проблем, возникающих при изменении настроек (например списка торговых площадок, поддерживаемых сервером) или восстановления после технических сбоев (например после восстановления базы данных из архива).

При нажатии на вышеуказанную кнопку открывается форма:



«Источник данных» позволяет выбрать, откуда (из вышестоящий источник или база подчиненного менеджера) будет выполнен повторный прием справочников.

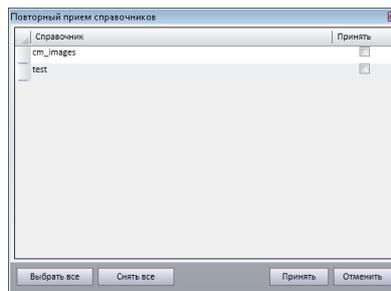
В списке справочников требуется выбрать те, которые следует принять повторно.

После нажатия на кнопку «Принять» сервер данных приступит в выполнению поставленной задачи.



- эта кнопка позволяет обновить каталоги распространяемых файлов. Эта кнопка доступна только на корневом сервере данных.

При нажатии на вышеуказанную кнопку откроется форма:



На форме отображается список всех распространяемых сервером файловых каталогов (управление этим списком описано в разделе «Настройка».«Закладка «Передача файлов»»). Требуется выбрать те каталоги, данные которых требуется обновить и нажать кнопку «Принять».

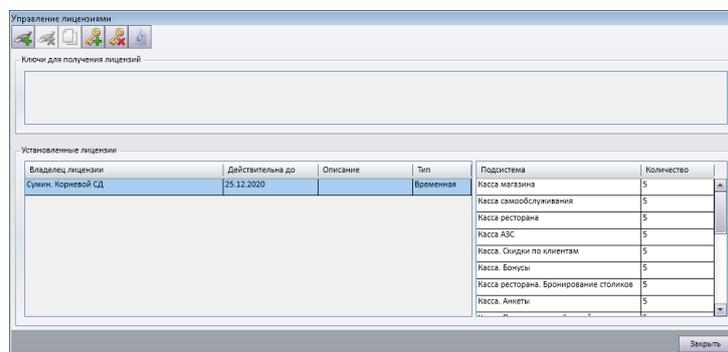


- эта кнопка позволяет инициировать внеочередную очистку устаревших данных на сервере. Следует помнить, что во время очистки сервер не может обслуживать клиентов, а сама очистка может длиться несколько часов, поэтому не следует использовать данную ф-цию без особой необходимости.



- эта кнопка позволяет войти в режим управления лицензиями.

При нажатии на вышеуказанную кнопку откроется форма:



Верхняя часть формы предназначена для инициализации аппаратного ключа (USB-flash накопителя) и получения уникального лицензионного ключа, позволяющего заказать фиксированную лицензию для сервера данных. Т.к. получение лицензии непосредственно на сервер данных считается устаревшим и не рекомендованным методом, то описывать

инициализацию аппаратного ключа в настоящем документе приведено не будем (об этом можно прочитать в документе «OpenStore.Сервер лицензий (руководство администратора)»).

В нижней части формы расположены установленные на сервере лицензии (слева) и их состав (справа).

Технологические особенности работы сервера.

Ранее было сказано о том, что в зависимости от своего положения в иерархической структуре системы, сервер данных может быть корневым или подчиненным. В первом случае сервер обменивается данными с транзитной базой данных, к которой требуется непосредственный доступ. Во втором случае, сервер данных является клиентом вышестоящего сервера данных. В терминологии OpenStore механизм обмена непосредственно с базой данных называется DATAPUMP, а механизм обмена данными с сервером данных – DATASYNK.

DATASYNK

Прежде всего, при обмене с вышестоящим сервером данных предполагается, что синхронизируются таблицы одинаковой структуры. Синхронизация производится последовательно (по таблично). При этом, вначале передаются таблицы из нижестоящего сервера к вышестоящему, а затем наоборот.

Все таблицы, участвующие в обмене данных заранее известны. Полный их список с указанием направления (поле SYNCDIR: 0 – вверх, 1 - вниз) и приоритета синхронизации (поле SYNCPRIORITY: чем меньше значение – тем выше приоритет) хранится в таблице DATASYNC_DIRS. Инициатором сеанса обмена выступает клиент (за исключением режима «по расписанию»).

В первую очередь он вычитывает из своей таблицы DATASYNC_DIRS все таблицы, которые нужно передать наверх (SYNCDIR=0), сортирует их по приоритету (SYNCPRIORITY), и последовательно, для каждой из них выбирает и передает вышестоящему серверу записи со значением UPDATENUM=0. Т.к. таких записей может быть очень много, передача осуществляется пакетами, размер которых указывается в настройках передающего (нижестоящего) сервера. После того, как пакет успешно передан вышестоящему серверу, для всех записей переданного пакета устанавливается UPDATENUM=1, что предотвращает попадание этих записей в следующую выборку. Если при передаче пакета возникает ошибка, то цикл обмена прерывается, а данные этого пакета будут переданы повторно на следующем цикле синхронизации.

После успешного завершения передачи данных, сервер переходит к приему. Для этого из своей таблицы DATASYNC_DIRS2 он вычитывает все таблицы, которые нужно принять

(SYNCDIR=1), сортирует их по приоритету (SYNCPRIORITY) и для каждой конкретной таблицы последовательно выполняется нижеописанный алгоритм синхронизации:

- подчиненный сервер сравнивает значение версии данных из своей таблицы DATASYNCDIRS2 с версией из DATASYNCDIRS вышестоящего сервера (поле UPDATENUM);
- версия подчиненного сервера может быть меньше либо равна версии вышестоящего, если это не так (может возникнуть в результате некорректного изменения данных в таблицах DATASYNCDIRS и DATASYNCDIRS2 из вне или восстановления более старой версии базы вышестоящего сервера из архива) – возникнет соответствующая ошибка, и цикл синхронизации будет прерван;
- если версия подчиненного сервера равна версии вышестоящего – считается, что данные в соответствующей таблице у серверов совпадают и синхронизация не требуется, переходим к следующей таблице;
- если версия подчиненного сервера меньше версии вышестоящего – на последний передается запрос на получение всех данных с версиями (поле UPDATENUM соответствующей таблицы) больше версии подчиненного и меньше или равно версии вышестоящего, после чего подчиненный сервер переходит в режим ожидания ответа, а вышестоящий – к подготовке данных;
- подготовка данных на вышестоящем сервере сводится к выбору записей из соответствующей таблицы с использованием вышеописанного условия, и сохранение этих данных в файлы-пакеты (размер такого файла указывается в настройках профиля синхронизации), представляющие собой заархивированные таблицы соответствующей структуры;
- после того, как данные для передачи подготовлены, вышестоящий сервер выставляет соответствующий флаг, а подчиненный, обнаружив этот флаг начинает принимать во временное хранилище подготовленные для него файлы (если во время приема данных происходит сбой канала связи, предпринимается трехкратная попытка принять файл начиная с места разрыва – т.е. выполняется «докачка»);
- приняв файлы, подчиненный сервер последовательно распаковывает их, и сохраняет данные в свою базу данных, после чего поднимает версию в таблице DATASYNCDIRS2 до уровня вышестоящего сервера, после чего синхронизация таблицы считается завершенной.

Следует отдельно отметить, что использование на подчиненном сервере таблицы DATASYNCDIRS2 вместо DATASYNCDIRS не случайно. Такой подход обусловлен тем, что подчиненный сервер данных может получать данные не только от вышестоящего сервера данных, но и от подчиненного сервера менеджера. Это требует хранения независимых значений версий данных непосредственно данного сервера (для передачи данных своим клиентам, эта версия увеличивается вне зависимости от того, откуда получены данные) и версию вышестоящего

сервера данных, для синхронизации с ним по вышеописанному алгоритму. Из этого следует, что при сохранении полученных с вышестоящего сервера данных в свою базу данных, значение поля UPDATENUM в каждой записи изменяется на значение [DATASYNC_DIRS.UPDATENUM + 1].

DATAPUMP

В отличие от DATASYNC этот протокол позволяет обмениваться данными с базой данных, структура которой в значительной степени отличается от структуры базы самого сервера данных. Синхронизация производится последовательно (по таблично). При этом, вначале передаются таблицы из сервера в транзитную базу, а затем наоборот. Все таблицы, участвующие в обмене данными со стороны сервера данных заранее известны. Полный их список с указанием направления (поле PUMPDIR: 0 – вверх, 1 - вниз) и приоритета синхронизации (поле PUMPPRIORITY: чем меньше значение – тем выше приоритет) хранится в таблице DATAPUMPDIRS базы сервера данных. Для того, чтобы сопоставить таблицы сервера данных с таблицами транзитной базы данных используется «маска синхронизации», которая, для нормальной работы сервера, должна содержать однозначное определение того, где и каким образом получаются данные для каждой входящей таблицы сервера и куда передавать, и передавать ли исходящие таблицы.

Цикл обмена с транзитной базой начинается с передачи в нее данных. Если используется тип передачи данных «По требованию», то этот этап выполняется только в случае наличия запроса в таблице COMMAND (подробнее в документе «Синхронизация с внешней системой»), в противном случае этап передачи данных выполняется на каждом цикле. В первую очередь сервер данных выбирает из таблицы DATAPUMPDIRS все таблицы, которые он может передавать в транзитную базу данных (PUMPDIR=0), сортирует их по приоритету (PUMPPRIORITY) и последовательно (по таблично) пытается передать данные в транзитную базу. Для этого выполняется следующий алгоритм:

- сервер ищет в маске синхронизации таблицу транзитной базы, сопоставленную с той, которую сервер собирается передать;
- если такой таблицы нет (на закладке «Справочники бэк-офиса на прием» нет таблицы, ссылающейся на передаваемую сервером таблицу), сервер переходит к обработке следующей таблицы из списка передаваемых;
- если в маске найдена сопоставленная таблица, то сервер данных выбирает из своей базы некоторое количество (указывается в настройках профиля синхронизации) ранее не переданных записей (UPDATENUM=0) или записей удовлетворяющих запросу (для типа передачи «По требованию»), и используя сопоставление полей формирует и вставляет в соответствующую таблицу транзитной базы записи;

- когда все выбранные записи будут обработаны, сервер данных пометит каждую из них как переданную (UPDATENUM=1) и начнет новую выборку не переданных или удовлетворяющих запросу (для типа передачи «По требованию») записей...

После успешного завершения передачи данных, сервер переходит к приему. Для этого из таблицы DATAPUMPDIRS он выбирает все принимаемые таблицы (PUMPDIR=1), сортирует их по приоритету (PUMPPRIORITY) и последовательно (по таблично) пытается принять данные, выполняя один из следующих алгоритмов (в зависимости от типа получения данных, указанного на «Закладка «Поставка данных»»):

1.) На основе номера обновления.

- сервер ищет в маске синхронизации синхронизируемую таблицу и смотрит, на какую таблицу транзитной базы она ссылается;

- если синхронизируемая таблица в маске не найдена (на закладке «Справочники фронта на прием») или для нее не указаны ссылка на таблицу транзитной базы и не стоит признак «Справочник содержит фиксированные данные», то возникнет ошибка, для устранения которой потребуется соответствующее исправление маски синхронизации;

- если указан признак «Справочник содержит фиксированные данные», то сервер сохраняет в синхронизируемую таблицу данные указанные в маске как «фиксированные», если таковые присутствуют и переходит к следующей таблице;

- найдя имя сопоставленной таблицы, сервер ищет соответствующую этому имени запись в таблице DATAPUMP (обязательная таблица) в транзитной базе;

- если запись в DATAPUMP не будет найдена, возникнет ошибка, свидетельствующая о некорректности выгруженных в транзитную базу данных или некорректности маски синхронизации;

- найдя сопоставленную таблицу в DATAPUMP, сервер сравнивает версии из DATAPUMPDIRS (поле PUMPUPDATENUM) и DATAPUMP (обязательное поле RECORDNUM);

- версия из DATAPUMPDYRS может быть меньше или равна версии из DATAPUMP, в противном случае возникнет ошибка;

- если версия из DATAPUMPDIRS равна версии DATAPUMP, считается, что данные в соответствующей таблице у сервера и транзитной базы совпадают и синхронизация не требуется, переходим к следующей таблице;

- если версия из DATAPUMPDIRS меньше то сервер выбирает все записи, версия которых больше версии из DATAPUMPDIRS и меньше или равна версии из DATAPUMP, вычитывает определенное количество этих записей (количество задается в профиле синхронизации) и при

помощи маски синхронизации составляет и сохраняет записи синхронизируемой таблицы, затем вычитывает новый пакет и т.д. до тех пор, пока все записи выборки не будут обработаны;

- после обработки всех записей выборки в таблице DATAPUMPDIRS устанавливается версия равная версии вычитанной из таблицы DATAPUMP на момент начала синхронизации данной таблицы, после чего осуществляется переход к следующей таблице.

2.) Получать весь справочник.

- сервер ищет в маске синхронизации синхронизируемую таблицу и смотрит, на какую таблицу транзитной базы она ссылается;

- если синхронизируемая таблица в маске не найдена (на закладке «Справочники фронта на прием») или для нее не указаны ссылка на таблицу транзитной базы и не стоит признак «Справочник содержит фиксированные данные», то возникнет ошибка, для устранения которой потребуется соответствующее исправление маски синхронизации;

- если указан признак «Справочник содержит фиксированные данные», то сервер сохраняет в синхронизируемую таблицу данные указанные в маске как «фиксированные», если таковые присутствуют и переходит к следующей таблице;

- найдя имя сопоставленной таблицы, сервер ищет соответствующую этому имени запись в таблице DATAPUMP (обязательная таблица) в транзитной базе;

- если запись в DATAPUMP не будет найдена, возникнет ошибка, свидетельствующая о некорректности выгруженных в транзитную базу данных или некорректности маски синхронизации;

- найдя сопоставленную таблицу в DATAPUMP, сервер проверяет наличие для ее флага наличия данных (обязательное поле RECORDNUM) отличное от 0 (равное 1);

- если флаг наличия данных установлен, то сервер выбирает все данные из сопоставленной таблицы, вычитывает определенное количество этих записей (количество задается в профиле синхронизации) и при помощи маски синхронизации составляет и сохраняет записи синхронизируемой таблицы, затем вычитывает новый пакет и т.д. до тех пор, пока все записи выборки не будут обработаны;

- после обработки всех записей выборки в таблице DATAPUMP сбрасывается в «0» флаг наличия данных и осуществляется переход к следующей таблице.