# Содержание

[Введение 4](#_bookmark0)

[Требования к выполнению лабораторных работ 5](#_bookmark1)

[Требования к отчётам о выполнении лабораторных работ 5](#_bookmark2)

[Лабораторная работа №1. ЗНАКОМСТВО С СИСТЕМОЙ](#_bookmark3) [1С:ПРЕДПРИЯТИЕ. ВИЗУАЛЬНОЕ СОЗДАНИЕ СТРУКТУРЫ](#_bookmark3) [КОНФИГУРАЦИИ 6](#_bookmark3)

[Теоретические сведения к выполнению работы 6](#_bookmark4)

[Задания к работе 29](#_bookmark5)

[Варианты выполнения лабораторной работы (тип предприятия) 35](#_bookmark6)

[Контрольные вопросы 36](#_bookmark7)

[Лабораторная работа №2. РАЗРАБОТКА КОНФИГУРАЦИИ:](#_bookmark8) [ОПРЕДЕЛЕНИЕ ПРАВ ДОСТУПА К ФУНКЦИОНАЛЬНОСТИ](#_bookmark8) [СИСТЕМЫ, НАСТРОЙКА ИНТЕРФЕЙСНЫХ МЕХАНИЗМОВ,](#_bookmark8) [СРЕДСТВА ИНТЕГРАЦИИ 37](#_bookmark8)

[Теоретические сведения 37](#_bookmark9)

[Содержание работы 66](#_bookmark10)

[Контрольные вопросы 71](#_bookmark11)

[Лабораторная работа №3. КОНФИГУРИРОВАНИЕ ПРИКЛАДНОГО](#_bookmark12) [РЕШЕНИЯ В СИСТЕМЕ 1С:ПРЕДПРИЯТИЕ В СООТВЕТСТВИИ С](#_bookmark12) [ПРИНЦИПОМ УЧЕТА «ОТ ДОКУМЕНТА» 72](#_bookmark12)

[Теоретические сведения 72](#_bookmark13)

[Содержание работы 97](#_bookmark14)

[Контрольные вопросы 103](#_bookmark15)

[Лабораторная работа №4. МЕХАНИЗМ ЗАПРОСОВ В СИСТЕМЕ](#_bookmark16) [1С:ПРЕДПРИЯТИЕ. ОБРАБОТКИ И ОТЧЕТЫ 105](#_bookmark16)

[Теоретические сведения 105](#_bookmark17)

[Содержание работы 133](#_bookmark18)

[Контрольные вопросы 138](#_bookmark19)

[Библиографический список 139](#_bookmark20)

# Введение

Знание принципов построения отраслевых информационных си- стем, методов проектирования и работы с пользовательскими интер- фейсами способствует рациональному выбору информационной си- стемы для решаемых задач в определённой области корпоративной деятельности.

Целью преподавания данной дисциплины является знакомство с базовыми понятиями отраслевых информационных систем, формиро- вание навыков разработки отраслевых решений на платформе 1С:Предприятие.

Изучение данной дисциплины позволяет выработать понимание основных принципов функционирования отраслевых информацион- ных систем, умение работать с предоставляемыми ими функциональ- ными возможностями, а также глубоко изучить область разработки прикладных решений и их сопровождение с использованием системы 1С:Предприятие.

Содержание дисциплины «Отраслевые информационные системы» основывается на двух сертифицированных курсах фирмы 1С («Азы программирования в системе 1С:Предприятие 8» и «Введение в кон- фигурирование в системе 1C:Предприятие 8. Основные объекты») и позволяет подготовить студентов к экзаменам для получения инду- стриальной сертификации 1С:Профессинал на знание основных меха- низмов технологической платформы 1С:Предприятие.

# Требования к выполнению лабораторных работ

Тематика лабораторных работ: автоматизация хозяйственной дея- тельности предприятия некого профиля.

Выполнение каждой лабораторной работы заключается в реализа- ции всех пунктов сценария, приведенного в разделе «Задания к рабо- те». Задания лабораторных работ выполняются в соответствии с пред- метной областью, определяемой вариантом студента или выбранной последним по согласованию с преподавателем.

Студент обязан перед выполнением каждой лабораторной работы самостоятельно ознакомиться с теоретическим материалом и по её результатам предоставить отчет.

# Требования к отчётам о выполнении лабораторных работ

Все отчёты о выполнении лабораторных работ оформляются в пе- чатном виде на листах формата А4.

Каждый отчёт должен содержать:

1. Заголовок лабораторной работы.
2. Цель работы.
3. Задание к лабораторной работе.
4. Ход выполнения работы.
5. Экранные формы разработанных решений.
6. Тексты программных модулей, если они необходимы при ре- шении поставленной задачи.
7. Выводы по сделанной работе.

# Лабораторная работа №1.

**ЗНАКОМСТВО С СИСТЕМОЙ 1С:ПРЕДПРИЯТИЕ. ВИЗУАЛЬНОЕ СОЗДАНИЕ СТРУКТУРЫ КОНФИГУРАЦИИ**

*Цель работы:* получение практических навыков по установке и начальной работе с системой 1С:Предприятие; знакомство с интерфей- сом программы и назначением основных объектов.

# Теоретические сведения к выполнению работы

## Общие сведения о системе 1С:Предприятие

Система 1С:Предприятие является универсальной системой авто- матизации экономической и организационной деятельности предприя- тия. Поскольку такая деятельность может быть довольно разнообраз- ной, система 1С:Предприятие имеет возможность «приспосабливать- ся» к особенностям конкретной области деятельности, в которой она используется. Для обозначения такой способности используется тер- мин конфигурируемость, то есть возможность настройки системы на особенности конкретного предприятия и класса решаемых задач.

Это достигается тем, что 1С:Предприятие – это не просто програм- ма, существующая в виде набора неизменяемых файлов, а совокуп- ность различных программных инструментов, с которыми работают разработчики и пользователи. Логически всю систему можно разде- лить на две большие части, которые тесно взаимодействуют друг с другом: *конфигурацию* и *платформу*, которая управляет работой кон- фигурации. Платформа – это среда разработки приложений (конфигу- раций) высокого уровня. Она же обеспечивает и работу приложений (выполнение программного кода). В платформе определен набор базо- вых объектов со своими свойствами и методами – это константы, до- кументы, справочники, регистры и другие. Всего набора объектов до- статочно для решения задач автоматизации учета, связанного с бизне- сом (и многих сопутствующих задач). Для работы со множеством объ- ектов платформа имеет встроенный язык 1С:Предприятие 8, SQL- подобный язык запросов, механизм расширений, утилиты для админи- стрирования, средства групповой разработки и др. Таким образом, из имеющегося в платформе набора объектов можно создавать конкрет- ные приложения (конфигурации), открытые для последующего редак- тирования.

Существует технологическая платформа 1С:Предприятие (написа- на на C++) и множество конфигураций. Для функционирования како- го-либо прикладного решения всегда необходима платформа и какая-

либо (одна) конфигурация. Имеется учебная версия технологической платформы для обучения разработке в среде 1С:Предприятие. Элек- тронная поставка учебной версии распространяется бесплатно. Акту- альную учебную версию платформы можно скачать с официального сайта фирмы 1С <https://v8.1c.ru/podderzhka-i-obuchenie/uchebnye-versii/> (без регистрации указав e-mail) либо с сайта ее учебного центра <https://uc1.1c.ru/uchebnaya-versiya-1s/>(с регистрацией).

Начиная с версии 8.2 в системе 1С:Предприятие существует не- сколько клиентских приложений: *толстый клиент*, *тонкий клиент*, *веб-клиент*. «Толстым» клиент называется потому, что может испол- нять практически всю функциональность, предоставляемую встроен- ным языком системы 1С. По этой же причине он требует значительно- го количества аппаратных ресурсов на компьютере пользователя и может обращаться к базе данных (в файловом режиме работы) или к кластеру серверов 1С:Предприятия 8 (в клиент-серверном варианте) только посредством файлового доступа или по локальной сети. В опе- рационной системе Windows исполняемый файл этого приложения – *1cv8.exe*. «Тонким» клиент называется потому, что умеет исполнять ограниченный набор функциональности встроенного языка. Вся рабо- та с базой данных, объектными данными, исполнение запросов – вы- полняется на стороне сервера. Тонкий клиент только получает готовые данные, подготовленные для отображения. В операционной системе Windows исполняемый файл этого приложения – *1cv8c.exe*. Веб-клиент (написан на JavaScript), в отличие от толстого и тонкого клиентов, ис- полняется не в среде операционной системы компьютера, а в среде интернет-браузера (*Windows Internet Explorer, Mozilla Firefox, Google Chrome* или *Safari*): любому пользователю достаточно ввести адрес веб-сервера, на котором опубликована информационная база. Невозможность ис- пользования веб-клиента может быть связана с требованиями инфор- мационной безопасности.

Развитие платформы 8.3 связано с охватом мобильных устройств. Для работы на мобильных устройствах (смартфоны, планшетные ПК) может использоваться специальная технология – *мобильная платформа* 1С:Предприятия 8. Она позволяет создавать самостоятельные прило- жения, работающие на мобильных устройствах под управлением опе- рационных систем Android или iOS. Предусмотрено три варинта плат- формы для мобильных устройств (*мобильная платформа*, *мобильный клиент*, *мобильный клиент с автономным режимом*), каждый из которых отличается возможностями по взаимодействию с информационными базами. Простейший вариант мобильной платформы использует для работы исключительно локальную базу на мобильном устройстве.

Начиная с версии 8.3.12 платформы появился следующий вариант – *мобильный клиент*. Мобильный клиент позиционируется как тонкий клиент для мобильных устройств, работает с существующими на сер- вере 1С:Предприятия информационными базами. Мобильный клиент требует постоянного интернет-соединения для взаимодействия с ин- формационными базами онлайн (аналогично работе клиентских при- ложений для десктопа). В отличие от тонкого клиента для мобильного клиента поддерживается функциональность, специфичная для мобиль- ных устройств и возможность работы с серверами разных версий:

* геопозиционирование,
* мультимедийные возможности,
* сканирование штрих-кодов и QR-кодов,
* контакты,
* календари,
* телефония,
* SMS/MMS,
* почта,
* push уведомления,
* поддержка механизма покупок и механизма рекламы,
* озвучивание текста,
* ожидание подходящего интернет-соединения,
* интеграция с сервисами сбора статистики поведения пользователей. Для мобильного клиента имеются следующие ограничения:
* не поддерживается ряд объектов и операций (по сравнению с техно- логической платформой);
* не достаточно быстрая работа через мобильный интернет;
* сложный цикл разработки, публикации и обновления по сравнению с тонким и веб-клиентом.

Начиная с версии 8.3.16 запущен гибридный вариант платформы для мобильных устройств – *мобильный клиент с автономным режимом*. Сценарий его работы зависимости от наличия интернет-соединения и его качества. При наличии соединения позволяет обращаться к инфор- мационным базам онлайн, иначе для текущей работы используется локальная база на мобильном устройстве.

Гибкость платформы позволяет применять 1С:Предприятие 8 в са- мых разнообразных областях:

* автоматизация производственных и торговых предприятий, бюджет- ных и финансовых организаций, предприятий сферы обслуживания и т.д.;
* поддержка оперативного управления предприятием;
* автоматизация организационной и хозяйственной деятельности;
* ведение бухгалтерского учета с несколькими планами счетов и про- извольными измерениями учета, регламентированная отчетность;
* широкие возможности для управленческого учета и построения ана- литической отчетности, поддержка многовалютного учета;
* решение задач планирования, бюджетирования и финансового ана- лиза;
* расчет зарплаты и управление персоналом;
* другие области применения.

Поскольку задачи автоматизации могут быть самыми разными, фирма «1С» и ее партнеры выпускают прикладные решения, каждое из которых предназначено для автоматизации одной определенной обла- сти человеческой деятельности. Реестр отраслевых и специализиро- ванных продуктов на базе 1С доступен на сайте [https://solutions.1c.ru/.](https://solutions.1c.ru/)

В качестве примеров существующих прикладных решений, кото- рых порядка трехсот, можно привести следующие:

* *1С:Управление торговлей* для автоматизации задач оперативного и управленческого учета торговых операций,
* *1С:Электронное обучение. Корпоративный университет* для дистанци- онного обучения и аттестации персонала,
* *1С:Зарплата и управление персоналом* для реализации кадрового учета и расчетов с персоналом,
* *1С:ERP Управление предприятием* для комплексного управления дея- тельностью многопрофильных предприятий,
* *1С:Бухгалтерия предприятия* для автоматизации всех основных задач бухгалтерского учета.

Прикладное решение является, по возможности, универсальным, чтобы удовлетворить потребности самых разных предприятий, рабо- тающих в одной области деятельности (отрасли). Такая универсаль- ность неизбежно приведет к тому, что на конкретном предприятии будут использоваться далеко не все возможности прикладного реше- ния, а каких-то возможностей в нем будет не доставать.

Гибкость в этом вопросе возможна за счет *конфигурируемости* си- стемы, поскольку платформа, помимо управления работой конфигура- цией, содержит средства, позволяющие вносить изменения в использу- емую конфигурацию. Более того, платформа позволяет создать свою собственную конфигурацию «с нуля», если по каким-либо причинам использование типовой конфигурации представляется нецелесообраз- ным.

Для того чтобы было возможно вносить изменения в программные модули системы, 1С:Предприятие имеет различные режимы работы: *1С:Предприятие* и *Конфигуратор*. Режим 1С:Предприятие является ос- новным и служит для работы пользователей системы. В этом режиме пользователи вносят данные, обрабатывают их и получают выходные результаты (отчеты). Режим конфигуратора используется разработчи- ками и администраторами баз данных, предоставляя инструменты для модификации существующей или создания новой конфигурации. C 2017 года появилась альтернативная среда разработки (аналог Конфи- гуратора) – *1C:Enterprise Development Tools*. 1С:EDT – это самостоятель- ное приложение на базе IDE Eclipse, устанавливаемое отдельно от платформы. Новая среда разработки ориентирована на создание мас- штабных бизнес-приложений. 1С:EDT позволяет быстро переключать- ся между разными версиями технологической платформы и информа- ционными базами в процессе работы, не покидая контекста разрабаты- ваемого прикладного решения.

## Установка системы 1С:Предприятие и вопросы лицензирования

Для лицензирования программных продуктов 1С используются технические средства аппаратной или программной защиты. Начиная с версии платформы 8.2.10 предусмотрен смешанный вариант работы, когда с одной и той же информационной базой возможно работать с одновременным использованием программной и аппаратной лицензии. Продукты системы 1С:Предприятие 8 подразделяются на несколь-

ко категорий:

* основные поставки (состав: платформа, конкретное прикладное ре- шение, лицензия на использование на одном рабочем);
* клиентские лицензии (лицензии на 1, 5, 10, 20 или 50 дополнитель- ных рабочих мест);
* лицензии на сервер приложений 1С:Предприятие, если предполага- ется использование продукта в клиент-серверном варианте;
* наборы (bundles), сочетающие несколько перечисленных категорий;
* специализированные категории, расширяющие возможности приме- нения системы 1С:Предприятие (технология 1cfresh для создания и публикации облачных решений, мобильные решения и др.)

Пользовательские поставки локализуются под международный ры- нок или конкретную страну (Азербайджан, Беларусь, Казахстан, Кыр- гызстан, Латвия, Литва, Эстония, Узбекистан, Армения, Грузия, Мол- дова).

Предусмотрены бесплатные полнофункциональные NFR-лицензии («not for resale») для партнеров фирмы 1С, предназначенные только

для внутреннего использования и демонстрации клиентам (продляются при условии, если фирма выполняется условия партнерского соглаше- ния); лицензии для разработчиков; поставки для учебных заведений по сниженным ценам; учебные версии для обучения программированию.

Для продвижения прикладных решений, созданных сторонними разработчиками, используется лицензирование тиражных решений по программе «1С-Совместно». Программа подразумевает, что тиражное решение стороннего производителя, созданное на основе технологий 1С, проходит экспертизу, при положительном результате комплекту- ется платформой 1С:Предприятие 8 и распространяется под торговой маркой фирмы 1С.

Официальные сведения по лицензированию программ 1С размеще- ны <https://v8.1c.ru/price/> и [https://v8.1c.ru/priobretenie-i-vnedrenie/otvety-](https://v8.1c.ru/priobretenie-i-vnedrenie/otvety-na-tipovye-voprosy-po-litsenzirovaniyu-1s-predpriyatiya-8/) [na-tipovye-voprosy-po-litsenzirovaniyu-1s-predpriyatiya-8/.](https://v8.1c.ru/priobretenie-i-vnedrenie/otvety-na-tipovye-voprosy-po-litsenzirovaniyu-1s-predpriyatiya-8/)

Перед установкой системы полезно ознакомиться с системными требованиями 1С:Предприятия 8 ([https://v8.1c.ru/tekhnologii/sistemnye-](https://v8.1c.ru/tekhnologii/sistemnye-trebovaniya-1s-predpriyatiya-8/) [trebovaniya-1s-predpriyatiya-8/](https://v8.1c.ru/tekhnologii/sistemnye-trebovaniya-1s-predpriyatiya-8/)).

Для начальной установки на компьютере пользователя запускается файл setup.exe из дистрибутива установки 1С:Предприятие 8 и в диа- логовом режиме выбираются компоненты для установки. На послед- нем шаге программа установки производит копирование файлов и со- здание программной группы и ярлыков. Если приобреталась поставка 1С:Предприятие с аппаратной защитой, то после окончания данных действий программа установки предлагает произвести установку драй- вера защиты от несанкционированного использования HASP Device Driver.

Установка драйвера требуется, если к USB-порту данного компь- ютера будет присоединяться аппаратный ключ защиты (локальный или серверный). При использовании сетевой лицензии или программной защиты драйвер устанавливать не имеет смысла.

Новые версии 1C:Предприятие выпускаются в виде полного дис- трибутива и в виде дистрибутива обновления (поставляется на дисках или через портал информационно-технического сопровождения <https://portal.1c.ru/>).

## Установка по образцу

При необходимости установки системы 1C:Предприятие 8 на не- сколько однотипных компьютеров (например, для подготовки компь- ютерного класса) можно выполнить установку на один компьютер, а затем провести установки на остальные компьютеры по образцу пер- вой.

Для этого необходимо выполнить запуск программы установки

*setup.exe* из командной строки с указанием ключей.

При первой установке необходимо задать ключ */r*. В результате установки в системном каталоге Windows будет создан файл *setup.iss*, содержащий ответы пользователя на вопросы программы установки (для изменения каталога файла *setup.iss* можно использовать ключ */f1* - например, */f1"*[*c:\temp\setup.iss"*](file://localhost/c:/temp/setup.iss)).

Этот файл можно использовать при последующих установках, за- давая ключ */s*, - файл *setup.iss* должен находиться в одном каталоге с *setup.exe* или же его местонахождение должно быть задано ключом */f1*.

При установке по образцу программа установки создаст файл *setup.log*, содержащий записи о ходе установки - этот файл будет со- здан в том же каталоге, что и *setup.iss*, однако можно явно задать ката- лог для *setup.log*, используя ключ */f2* - например, */f2"c:\temp\setup.log"*.

## Административная установка

Данный режим предназначен для установки системы 1С:Предприятие на компьютеры, объединённые в локальную сеть.

При использовании административной установки администратор локальной сети получает возможность устанавливать 1C:Предприятие с единого источника в сети, а затем обновлять версию системы только в месте административной установки. Дальнейшее обновление версий на локальных рабочих местах производят сами пользователи при оче- редном запуске системы.

Поскольку административная установка рассчитана на подготов- ленных пользователей, возможность такой установки не вынесена в пользовательский интерфейс программы установки *setup.exe*. Для вы- полнения административной установки необходимо запустить про- грамму установки *setup.exe* с ключом */а*. После запуска на экран выво- дится диалог, в котором необходимо указать каталог административ- ной установки. В указанном каталоге будет создан набор файлов, не- обходимый для установки по сети на локальных рабочих местах.

Для установки системы 1C:Предприятие на локальных рабочих ме- стах необходимо запустить программу *setup.exe* из каталога админи- стративной установки.

При выходе очередной версии 1C:Предприятия выполняется про- цедура обновления административной установки. При очередном за- пуске 1C:Предприятия с локального рабочего места пользователю бу- дет предложено произвести обновление локальной установки системы. При подтверждении необходимости обновления процедура обновле- ния производится в автоматическом режиме.

Если по каким-либо причинам требуется отказаться от проверки наличия обновления, то достаточно удалить файл *Admupd.cfg*.

## Установка конфигурации

Конфигурации для системы 1С:Предприятие устанавливаются из шаблонов. После запуска шаблона программа установки предлагает выбрать каталог на жестком диске, куда установится шаблон конфигу- рации.

В том случае, если в указанном вами каталоге для установки шаб- лона конфигурации уже существуют подкаталоги с установленной ра- нее конфигурацией, программа сообщает о найденной конфигурации и повторно запрашивает о необходимости установки в данный каталог. В случае подтверждения будет произведена перезапись файлов конфи- гурации.

## Отличительные особенности установки 1С:Предприятие в варианте клиент-сервер

1С:Предприятие позволяет работать с информационными базами в варианте «клиент-сервер», под которым понимается архитектура с тремя программными уровнями:

* клиентское приложение 1C:Предприятие;
* сервер 1С:Предприятие;
* сервер баз данных.

Клиентское приложение 1C:Предприятия – это программа, с кото- рой работает конечный пользователь. Для того, чтобы 1С:Предприятие получило возможность работать с информационными базами в вари- анте «клиент-сервер», обычная установка, позволяющая работать с файловым вариантом информационной базы, должна быть дополнена специализированными компонентами доступа к серверу 1C:Предприятие. Это достигается во время процедуры установки вы- бором флажка «Компоненты доступа к серверу 1С:Предпрития 8». При этом устанавливаемое 1C:Предприятие получает возможность работы с информационными базами, как в файловом варианте, так и в вариан- те «клиент-сервер».

Установка сервера 1С:Предприятия выполняется программой уста- новки 1C:Предприятия, которая копирует на компьютер необходимые файлы и настраивает сервер 1С:Предприятия. Приложение сервера 1C:Предприятия выполняется от лица специального пользователя, например «USER1CV8SERVER», который также может быть создан в процессе установки программы.

Сервер 1С:Предприятия – это специализированное серверное при- ложение, через которое осуществляется доступ к информационной базе в варианте «клиент-сервер». Он образует промежуточный про- граммный слой между клиентским приложением и сервером баз дан- ных. Клиентские приложения не имеют непосредственного доступа к серверу баз данных. При этом, помимо простой передачи данных от клиентского приложения серверу баз данных, сервер 1C:Предприятие выполняет и ряд других задач. В частности, в его среде может быть организовано выполнение достаточно сложных обработок, написан- ных на встроенном языке системы 1С:Предприятие.

Сервер 1С:Предприятие может функционировать на компьютере, работающем под управлением

* MS Windows XP/ Vista/ Server 2003/ Server 2008/ Server 2012/ Server 2012 R2/ Server 2016/ 7/ 8/ 8.1/ 10 (архитектуры x86, x86-64);
* Linux (архитектуры х86 и х86-64): Astra Linux/ CentOS/ Debian/ Mint/ Red Hat Enterprise/ Ubuntu LTS/ Альт Линукс/ Альт Рабочая станция/ Альт Сервер.

Хранение данных информационных баз в варианте «клиент-сервер» обеспечивается сервером баз данных, в качестве которого может ис- пользоваться промышленная СУБД MS SQL Server/PostgreSQL/IBM DB2/Oracle Database. При этом каждая информационная база целиком сохраняется в отдельной базе данных.

Клиентские приложения 1С:Предприятие, сервер 1C:Предприятие и сервер баз данных могут быть распределены по компьютерам доста- точно произвольным образом. Также они вполне могут работать и на одном компьютере. Сервер 1C:Предприятие и сервер баз данных при работе создают примерно одинаковую нагрузку на компьютеры, на которых они исполняются.

Сервер 1С:Предприятие защищен от несанкционированного ис- пользования с помощью ключа аппаратной защиты HASP4 или про- граммных лицензий.

## Совместное использование различных версий системы 1С:Предприятия

Начиная с версии 1С:Предприятие 8.2 вводится гибкий механизм использования множества различных версий системы на одном ком- пьютере. В этом случае каждая новая версия системы автоматически устанавливается в новую папку с именем номера версии, что не приво- дит к затиранию предыдущих версий, установленных на компьютере.

## Понятие «информационная база» и создание её резервной копии

К информационным базам относится наиболее ответственная ин- формация, включающая: конфигурацию, все данные о хозяйственной деятельности предприятия, а также административную информацию. Хранение информационной базы осуществляется в базе данных в виде набора таблиц, для чего 1С:Предприятие может использовать один из двух вариантов:

* СУБД, встроенную в 1С:Предприятие (файловый вариант информа- ционной базы). В этом случае все данные информационной базы хранятся в файле, например с именем 1Cv8.1CD.
* Внешний сервер баз данных (клиент-серверный вариант информаци- онной базы). Все данные информационной базы хранятся на соответ- ствующем сервере баз данных, например MS SQL Server.

Периодически (в зависимости от интенсивности изменения базы) рекомендуется создавать архивную копию рабочей базы данных на автономных носителях информации для возможности восстановления рабочей базы из архива при её случайном разрушении.

Резервное копирование и восстановление базы данных, хранящей информационную базу, обязательно должно выполнялось только це- ликом. С этой целью рекомендуется использование средств резервного копирования баз данных, встроенных в сервер БД. Резервное сохране- ние файлового варианта информационной базы может быть выполнено копированием файла информационной базы, например 1Cv8.1CD.

В конфигураторе есть специальная функция: Администрирование > Выгрузить информационную базу. С ее помощью можно выгрузить в указанный файл (файл выгрузки) все данные, относящиеся к информа- ционной базе. Обратная ей функция, Загрузить информационную базу, позволяет в текущую информационную базу вместо существующих загрузить все данные из файла выгрузки. Эти функции можно исполь- зовать для резервного копирования данных информационной базы как в файловом, так и в клиент-серверном варианте.

## Создание пустой информационной базы без конфигурации

В окне запуска системы 1С:Предприятие (толстый клиент) доступ- но два режима работы – 1С:Предприятие и Конфигуратор.

Добавление новой информационной базы осуществляется путём нажатия кнопки Добавить… окна запуска 1С:Предприятие. Для созда- ния пустой информационной базы во вновь открывшемся окне выби- рается пункт Создание новой информационной базы, а в следующем окне – Создание информационной базы без конфигурации….

Начиная с версии 8.0, добавлена возможность создания информа- ционных баз при помощи шаблонов, которые могут поставляться фир- мой 1С и партнерами. Это упрощает процесс создания множества ин- формационных баз.

Для создаваемой информационной базы задаётся наименование, тип расположения, каталог (если был выбран тип расположения базы На данном компьютере) и язык. Если указанный каталог не существует, то он будет создан в файловой системе. На последнем шаге создания указываются параметры запуска (вариант аутентификации пользовате- ля в системе, скорость соединения, основной режим запуска и версия 1С:Предприятия). Для учебной версии не подходит вариант аутенти- фикации Запрашивать имя и пароль, поскольку она не поддерживает хранение и работу с паролями. Изменение опции Скорость соединения требуется, если предполагается эксплуатация продукта по медленным каналам связи. За счет встроенных средств платформы возможна оп- тимизация размера трафика путем уменьшения количества обращений к серверу и отказа от передачи некоторой информации. Поле Версия 1С:Предприятия заполняется в том случае, если на компьютере установ- лены разные релизы платформы, и требуется запускать информацион- ную базу из-под конкретной версии.

## Файловая структура базы

В каталоге базы находятся (могут находиться) файлы, перечислен- ные в таблице 1.

Начиная с версии 8.0, список и настройки параметров информаци- онных баз расположены не в реестре Windows, как в предыдущих вер- сиях, а в каталогах профиля пользователя %AppData%\1C\. При уста- новке/переустановке операционной системы для восстановления спис- ка информационных баз не нужно искать и сохранять ветки реестра, а достаточно скопировать файл ibases.v8i.

Таблица 1. Структура ИБ системы 1С:Предприятие

|  |  |
| --- | --- |
| **файл** | **Описание** |
| 1Cv8.1CD | файл информационной базы (структура и данные) с именем по умолчанию |
| \*.pfl | параметры и настройки по умолчанию, необходимые для старта и работы с 1С |
| \*.st | файлы шаблонов текстов |
| \*.log | лог-файл журнала регистрации |
| \*.epf | файлы внешних обработок |
| \*.erf | файлы внешних отчетов |
| \*.dt | архивная копия базы со всеми данными (специализиро- |

|  |  |
| --- | --- |
| **файл** | **Описание** |
|  | ванный формат архива 1С 8) |
| \*.cf | файл поставки конфигурации (код и структура при- кладного решения без пользовательских данных) |
| \*.cfu | файл обновления конфигурации (только отличия новой конфигурации от ее предыдущей версии) |
| \*.pff | файл с сохраненными замерами производительности для системных администраторов и специалистов 1С |

## Понятие «конфигурация»

Функционирование системы делится на два процесса — *конфигури- рование* (описание модели предметной области средствами системы) и *исполнение* (обработку данных предметной области). Результатом кон- фигурирования является конфигурация, которая представляет собой модель предметной области.

*Конфигурацией* в системе 1С:Предприятие называется совокупность взаимосвязанных составных частей:

* структуры учетных данных, форм их ввода, выбора, печати;
* состав механизмов учета итоговых данных и движений учетных данных;
* состав различных отчетов и обработок;
* набора пользовательских интерфейсов;
* набора ролей (прав доступа);
* набора общих процедур и функций (глобальный модуль и общие модули), макетов табличных документов, и др.;
* вспомогательных объектов: картинки, шаблоны, стили и т.д.

## Объект конфигурации

Под ***объектом конфигурации*** в системе 1С:Предприятие понима- ется формальное описание группы понятий (предметной области, средств взаимодействия пользователя с системой) со сходными харак- теристиками и одинаковым предназначением.

Каждый объект конфигурации обладает уникальным набором свойств. Этот набор описан на уровне системы и не может быть изме- нен в процессе настройки конфигурации задачи. Набор свойств объек- та конфигурации определяется, в основном, его назначением в системе 1С: Предприятие.

Главным свойством любого объекта конфигурации является Имя — краткое наименование объекта конфигурации. При создании нового объекта конфигурации ему автоматически присваивается условное имя, состоящее из слова, определяемого по категории объекта, и циф-

ры (например, при создании реквизита создается реквизит с именем Реквизит1, при создании документа — Документ1, и т.д.). Это имя можно изменить в процессе редактирования свойств объекта конфигурации, при этом система отслеживает уникальность имен. Имя объекта кон- фигурации не может быть пустым.

## Основы работы в конфигураторе

После запуска программы в режиме конфигуратора открывается окно конфигуратора, с помощью которого настраивается или создается требуемая конфигурация.

Первая команда, с которой начинается работа с любой конфигура- цией – Конфигурация > Открыть конфигурацию. На экране откроется де- рево конфигурации (дерево метаданных).

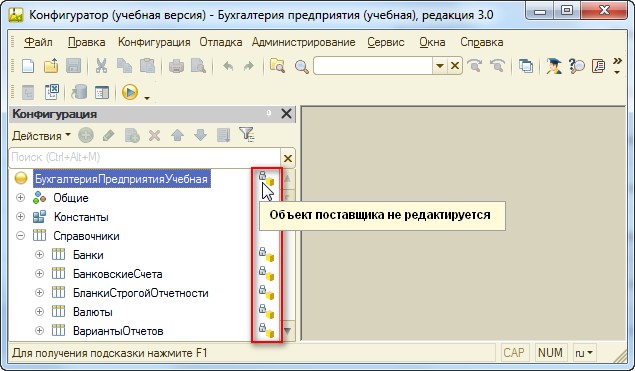
Формально дерево конфигурации представляет собой объединён- ные в классы объекты конфигурации. В таблице 2 приведена краткая характеристика основных видов объектов конфигурации системы 1С: Предприятие.

Таблица 2. Основные классы объектов конфигурации

|  |  |
| --- | --- |
| **Наименование** | **Краткая характеристика** |
| Общие | Группа вспомогательных объектов конфигурации, с помощью которых производится настройка поль- зовательского интерфейса; определяются права доступа к объектам конфигурации; разрабатывают- ся механизмы взаимодействия пользователей с данными; настраиваются механизмы обмена дан-  ными с внешними системами. |
| Константы | Постоянные (условно-постоянные) величины. Кон- станты хранят информацию, которая не изменяется или изменяется достаточно редко: название органи-  зации, ее почтовый адрес и так далее. |
| Справочники | Списки однородных элементов данных. Использу- ются для хранения нормативно- справочной ин-  формации. |
| Документы | Служат для ввода информации о совершенных хо- зяйственных операциях. |
| Журналы доку- ментов | Служат для отображения списков документов раз- ного вида. |
| Перечисления | Списки значений, задаваемые на этапе конфигури- рования. |
| Отчеты | Средство получения выходной информации. Ис- точником данных для построения отчетов служат документы, справочники и регистры, также исполь-  зуется информация, хранящаяся в константах. |

|  |  |
| --- | --- |
| **Наименование** | **Краткая характеристика** |
| Обработки | Объекты конфигурации этого вида используются для выполнения различных действий над информа- ционной базой. |

В окне конфигурации можно создавать, редактировать и удалять нужные объекты. При этом система контролирует ссылочную целост- ность структуры метаданных – если на данный объект есть ссылки из других объектов, то его нельзя удалить без разрыва связей. Для про- верки ссылок следует использовать Поиск ссылок на объект или Поиск ссылок в объекте из меню кнопки Действия дерева метаданных. Также существует Глобальный поиск из меню Правка. С помощью него можно находить заданную строку не только во всех текстах программных модулей, но и элементах интерфейса, макетах (печатных формах), внешних файлах.

Типовые конфигурации в начале внедрения и эксплуатации по умолчанию находятся на поддержке поставщика (фирма «1С») без возможности изменения объектов и их свойств. Возможность измене- ния конфигурации включается через настройку Конфигурация > Под- держка > Настройка поддержки… .

**Рис. 1 – Объекты типовой конфигурации с запретом редактирования**

Редактирование существующих свойств объекта производится в Палитре свойств. Поскольку разные объекты конфигурации имеют раз- ные свойства, содержимое этого окна будет меняться в зависимости от того, какой объект является активным. При некоторых действиях раз- работчика (например, создание табличной части) палитра свойств от- крывается автоматически. Но существует возможность открыть палит-

ру свойств активного объекта конфигурации самостоятельно, восполь- зовавшись пунктом Свойства контекстного меню или сочетанием кла- виш [Alt+Enter].

Информацию по конструкциям встроенного языка программирова- ния можно получить, используя Синтаксис-помощник из меню Справка.

## Сохранение конфигурации

После изменений в конфигурации ее следует сохранять. Для сохра- нения выбираем меню Конфигурация > Сохранить конфигурацию либо стандартная комбинация клавиш [Crtl+S].

Начиная с версии 8.0, изменение конфигурации двухступенчатое: вводится понятие Конфигурация (основная конфигурация, конфигура- ция разработчика) и Конфигурация базы данных. Теперь не нужно пре- кращать работу пользователей в запущенных конфигурациях для того, чтобы сохранить текущие изменения. Просто в нужные моменты со- храняем конфигурацию, при этом конфигурация базы данных не за- трагивается. И лишь когда всё реализовано и необходимо осуществить проверку реализации – выполняется перенос изменений из конфигура- ции в конфигурацию базы данных (Конфигурация > Обновить конфигу- рацию базы данных или клавиша [F7]). В этом случае нужен монополь- ный доступ к базе.

## Режим работы пользователя

Начиная с версии 8.2 появляется возможность выбора режима ра- боты пользователя: *обычный* и *управляемое приложение. Обычный* режим обеспечивает совместимость информационных систем, разработанных в более ранних версиях. Режим *Управляемое приложение* позволяет ра- ботать с *управляемым интерфейсом* и *управляемыми формами.* Послед- ний режим ориентирован на комфортную и эффективную работу с си- стемой, основа которой определяется возможностями web-технологий.

## Константы

В системе 1С:Предприятие *константы* предназначены для хране- ния постоянной или условно-постоянной информации. Такая инфор- мация либо совсем не изменяется в процессе деятельности предприя- тия, либо изменяется достаточно редко, но, как правило, часто исполь- зуется в работе. Например, в константах может храниться наименова- ние предприятия, его ИНН, организационно-правовая форма и другая подробная информация.

В системе может быть описано неограниченное количество кон- стант.

Основная причина использования констант заключается в том, что в них один раз заносится какая-либо информация, которая затем может использоваться при заполнении полей документов и при построении отчетных форм. Значение константы время от времени может редакти- роваться.

Создание нового объекта конфигурации (например, создание кон- станты) производится следующим образом:

* 1. установить курсор на тот тип объекта, который необходи- мо создать. В нашем случае на Константы;
  2. в командной панели окна конфигурации нажать кнопку Действия > Добавить, или, вызвав контекстное меню, вы- брать пункт Добавить.

В качестве имени константы может служить её название, написан- ное без пробелов и начинающееся с буквы. Например: НазваниеОрга- низации. После ввода имени константы автоматически заполняется синоним, в котором расставляются пробелы и заменяются заглавные буквы.

Константа является типизированным объектом конфигурации.

Например, тип константы НазваниеОрганизации будет Строка.

В 1С:Предприятии 8 тип Строка доработан: помимо фиксированной длины (дополнение пробелами) и неограниченной длины, появилась возможность задать переменную длину. Длина реквизита такого типа может быть меньше заданной. При сравнении таких строк не нужно обрезать лишние пробелы.

## Перечисления

Перечисление является прикладным объектом конфигурации и предназначено для описания структуры хранения постоянных наборов значений, не изменяемых в процессе работы конфигурации.

Набор всех возможных значений, которые содержит перечисление, задается при конфигурировании системы, и пользователь не может изменять их, удалять или добавлять новые. Пользователь видит значе- ния перечислений только в списках выбора на формах. Следовательно, на перечисления могут опираться алгоритмы работы программы.

## Справочники

При заполнении бланка какого-либо документа часто требуется указывать информацию, выбирая значение из заранее заданного спис- ка. Для работы с постоянной и условно постоянной информацией с некоторым множеством значений в системе используются объекты типа Справочник.

Справочники используются в тех случаях, когда необходимо ис- ключить неоднозначный ввод информации. Например, для того, чтобы покупатель, продавец, кладовщик, директор понимали, о каком товаре идет речь, каждый должен называть его одинаково.

Система 1С:Предприятие позволяет вести практически неограни- ченное количество необходимых справочников. Каждый справочник представляет собой список однородных объектов: сотрудников, орга- низаций, товаров и т.д. Каждый такой объект будем называть элемен- том справочника.

Следует иметь в виду, что в Конфигураторе создается структура справочника, а собственно его содержимое – элементы справочника – заносятся при работе с программой пользователем.

Справочники имеют стандартные реквизиты Код и Наименование, при этом код может иметь тип число или строка. Система 1С:Предприятие предоставляет широкие возможности по работе с ко- дами элементов справочника: автоматическое присвоение кодов, авто- матический контроль уникальности кода и другие.

Справочник в системе 1С: Предприятие может быть иерархическим. В этом случае все данные справочника будут разделяться на два вида: обычные элементы и группы справочника. Группы — это логическое объединение элементов справочника. Примером иерархического спра- вочника может служить справочник товаров, где группами являются категории товаров («Кондитерские изделия», «Напитки» и.т.д.), а эле- ментами – конкретные товары («Конфеты «Белочка», «Sprite»).

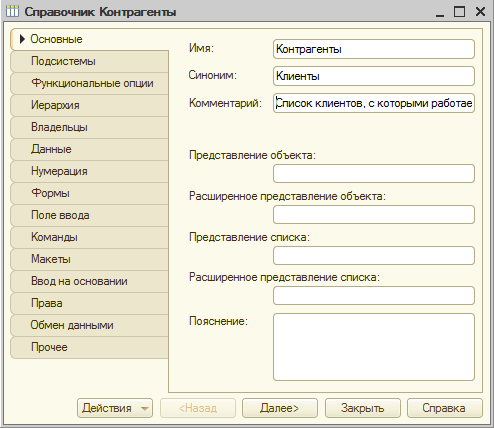
Помимо кода и наименования, для справочника можно создать произвольный набор реквизитов, позволяющих хранить дополнитель- ную информацию об элементе справочника. Например, справочник Контрагенты может содержать такие сведения как полное наименование контрагента, его ИНН, фамилии директора и главного бухгалтера, и другую информацию.

Если объект предметной области, которой соответствует справоч- ник, имеет не только «простые» свойства, например, полное наимено- вание или ИНН, но и составные (списочные) свойства, в структуру справочника может быть добавлен набор табличных частей.

## Создание справочника

Рассмотрим основную концепцию создания справочника на приме- ре создания справочника контрагентов. Под контрагентами будем под- разумевать клиентов, с которыми работает предприятие (поставщики, покупатели, партнеры).

Создадим справочник, в котором будут храниться наименования клиентов. После нажатия в командной панели дерева конфигурации

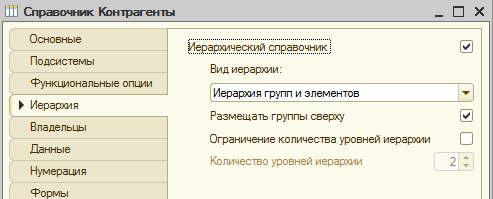
кнопки Добавить откроется окно редактирования объекта конфигура- ции (мастер объекта). Зададим Имя, Синоним и Комментарий справоч- ника (см. рисунок 2).

**Рис. 2 – Мастер объекта Справочник**

Свойство Синоним предназначено для хранения альтернативного наименования объекта конфигурации, которое будет использовано в элементах интерфейса и показано конечному пользователю.

На вкладке Основные мастера объекта можно указать различные представления в пользовательском интерфейсе для этого объекта (Представление объекта, Расширенное представление объекта, Представ- ление списка, Расширенное представление списка). Если оставить поля пустыми, во всех формах пользовательского интерфейса будет исполь- зовано значение Синонима.

По кнопке Далее > или перемещаясь по вкладкам, настроим пара- метры иерархии объекта конфигурации. Множество контрагнетов мо- жет быть разбито на несколько групп, например, поставщики и поку- патели. Если с точки зрения предметной области нет необходимости вводить группы, то создается справочник без иерархии. Для поддерж- ки иерархии в справочнике служит флаг Иерархический справочник.



**Рис. 3 – Настройки иерархии в справочнике**

Поддерживается два варианта иерархии: Иерархия групп и элементов и Иерархия элементов. Первый вариант подразумевает, что есть элемен- ты справочника и их логические (абстрактные) объединения – группы, причем набор реквизитов групп может отличаться от набора реквизи- тов элементов.

Во втором случае справочник состоит из однотипных элементов, подчиненных друг другу. Любой элемент может быть группой для набора других элементов. Тогда все элементы справочника – это одно- типные сущности, но с разным уровнем детализации, например, спра- вочник подразделений (каждое подразделение может входить в состав более крупного подразделения).

Для справочника с клиентами подходит первый вариант.

Если снять галочку с пункта Размещать группы сверху, то группы бу- дут сортироваться в порядке, установленном для элементов, например по алфавиту. Оставим по умолчанию.

В версии 8 снято ограничение на максимальную вложенность эле- ментов справочников. При необходимости ставим галочку Ограниче- ние количества уровней иерархии и указываем максимально возможное количество вложений.

На вкладке Владельцы задается список владельцев данного справоч- ника и вариант использования подчинения. Например, можно настро- ить, что элементы справочника Договоры будут принадлежать какому- то конкретному контрагенту. Использование механизма подчинения исключает ситуацию, когда договоры одного контрагента присваивают другому при совершении хозяйственных операций.

В версии 8 у справочника может быть несколько владельцев, то есть справочник может быть подчинен сразу нескольким объектам, но каждый элемент подчиненного справочника имеет только одного вла- дельца.

На закладке Данные определяется весь состав реквизитов (полей) справочника и, если нужно, состав табличных частей.

Длина кода по умолчанию нас устроит. (9 символов дают 109-1 элементов при числовом представлении кода). Длину наименования изменим на 150. Тип кода может быть числовой или строковый. Ос- новное представление определяет, в виде чего (код или наименование) будет отображаться ссылка на элемент справочника в формах выбора.

Реквизиты – это поля данных, содержащие дополнительные сведе- ния о конкретном объекте. Заведем основные данные о контрагентах:

Таблица 3. Реквизиты справочника Контрагенты

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| НаименованиеПолное | Строка | длина = 250 |
| НаименованиеСокращенное | Строка | длина = 250 |
| ИНН | Строка | длина = 12, переменная |
| КПП | Строка | длина = 9, переменная |
| ДатаРегистрации | Дата |  |
| ЮридическийАдрес | Строка | неограниченная длина |
| Телефон | Строка | в формате с кодом стра- ны и города |
| ВебСайт | Строка |  |

Для создания нового реквизита вызываем контекстное меню заго- ловка ветки Реквизиты и выбираем Добавить (либо используем кнопку  на панели инструментов). Аналогичные команды доступны и через дерево конфигурации.

Свойство Использование реквизита содержит в выпадающем списке три варианта (имеют смысл только для иерархического справочника):

* для элемента – реквизит может принадлежать только элементу спра- вочника;
* для группы – реквизит может принадлежать только группе;
* для группы и элемента – реквизит используется на всех уровнях иерархии справочника.

Свойство Индексировать: позволяет установить индексирование справочника по значениям данного реквизита для ускорения отбора по нему. Не стоит им злоупотреблять – может сказаться на производи- тельности в целом. Не используется для реквизитов с типом данных Строка неограниченной длины.

Последняя категория – Тип данных. Не стоит часто использовать строки неограниченной длины, они занимают лишние ресурсы.

В 1С:Предприятии 8 предусмотрена возможность задавать состав- ной тип данных для реквизитов справочника (и реквизитов других объектов) и возможность добавлять одну или несколько табличных частей (ТЧ). В них рекомендуется хранить связанную с элементом

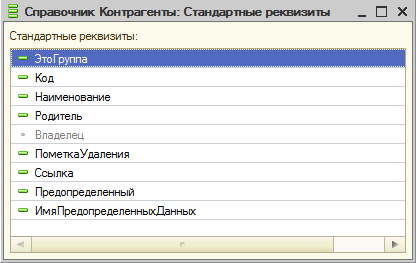
справочника информацию. Например, организуем хранение данных контактных лиц контрагентов через табличную часть справочника.

Нажимаем кнопку  и называем созданную ТЧ КонтактныеЛица.

Нажимаем кнопку  и создаем реквизиты ТЧ. Добавим реквизиты:

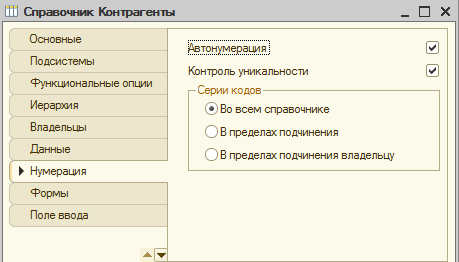
Таблица 4. Структура табличной части справочника Контрагенты

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| ФИО | Строка | длина = 50, переменная |
| Должность | Строка | длина = 30, переменная |
| РабочийТелефон | Строка | в формате с кодом стра- ны и города |
| МобТелефон | Строка |  |
| АдресЭлПочты | Строка | длина = 50, переменная |

Также через вкладку Данные доступны стандартные реквизиты справочника (по кнопке ), заложенные на уровне платформы. По мере задействования механизмов справочника (иерар- хия, владелец) эти реквизиты становятся активными. Для каждого из них можно вызвать Палитру свойств [Alt+Enter].

**Рис. 4 – Стандартные реквизиты объекта конфигурации Справочник**

Нажимаем кнопку Далее > и переходим к настройке параметров нумерации справочников. Второй и третий варианты серий кодов имеют смысл только для подчиненных справочников: для них можно выбрать, чтобы каждая группа элементов, подчиненная одному вла- дельцу, имела свою нумерацию. Иначе нумерация будет сквозная (ва- риант Во всем справочнике). Все остальное оставляем без изменения и переходим на вкладку Поле ввода.

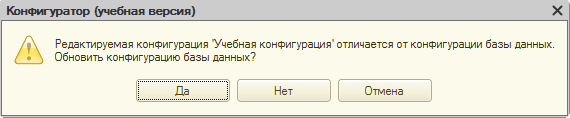


**Рис. 5 – Настройки нумерации элементов справочника**

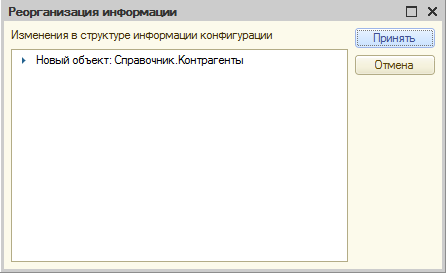
Вкладка Поле ввода позволяет задействовать механизм Ввода по строке для пользовательского режима (по части слова, введенной в по- ле ввода на форме, пользователю будут показаны подходящие вариан- ты подстановки для быстрого выбора нужного элемента справочника).

Знакомство со свойствами справочников продолжим на этапе вы- полнения практического задания к работе.

После того как были сделаны изменения в конфигурации (после со- здания справочников и других объектов конфигурации), из конфигура- тора выполним пункт меню Отладка > Начать отладку [F5].

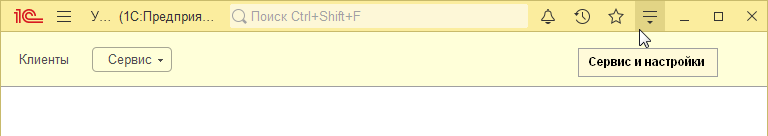
На появившейся вопрос

**Рис. 6 – Этап сохранения конфигурации БД**

ответим Да и в следующем окне нажмем Принять:

**Рис. 7 – Список изменений в структуре конфигурации**

Откроется окно системы в режиме 1С:Предприятие, через которое осуществляется навигация по функциональности созданной конфигу- рации.

Для просмотра результатов работы и заполнения справочников можно воспользоваться навигационными ссылками в панелях, либо пунктом Функции для технического специалиста… из меню Сервис и настройки:

**Рис. 8 – Доступ к настройкам и сервисным функциям режима 1С:Предприятие**

Выбираем справочник Клиенты. Система откроет одну из основных форм справочника – форму списка, сгенерированную автоматически, поскольку при создании справочника на вкладке «Формы» мы остави- ли настройки по умолчанию.

**Рис. 9 – Справочник Клиенты в пользовательском режиме**

Подробно элементы пользовательского интерфейса и принципы его построения будут рассмотрены в следующей работе.

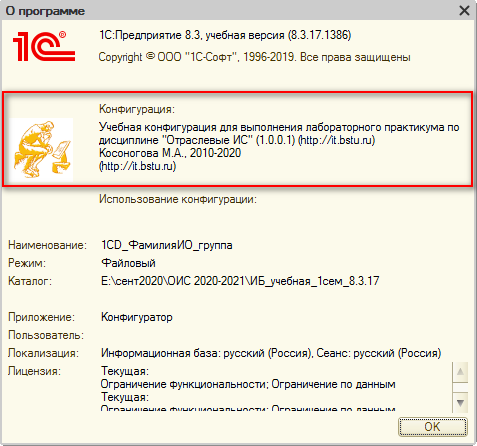
В процессе отладки допускается редактирование текущей конфигу- рации и сохранение изменений. Но для отладки новых изменений по- требуется завершить работу в режиме 1С:Предприятие, сохранить из- менения, обновить конфигурацию базы данных и повторно запустить конфигурацию в режиме откладки.

# Задания к работе

**Обязательная часть (макс. оценка за выполнение - «хорошо»)**

1. Установите систему 1С:Предприятие 8. Учебная версия в папку по умолчанию технологической платформы («сетевая папка установки»\platform\setup.exe).
2. Ознакомьтесь с элементами окна Запуск 1С:Предприятия (учебной версии).
3. Установите конфигурацию Бухгалтерия предприятия (учебная)

(«сетевая папка установки»\accounting\setup.exe).

1. Создайте новую информационную базу Бухгалтерия предприятия учебная (демо) на основе одноименного шаблона.
2. Создайте новую чистую информационную базу без конфигура- ции, наименование информационной базы – 1СD\_ФамилияИО\_группа (база будет использоваться в дальнейшем в течение всего семестра для конфигурирования по индивидуальным заданиям).
3. Для корневого элемента дерева метаданных (по умолчанию – Конфигурация) через палитру свойств заполните свойства Имя, Синоним, Краткая информация, Подробная информация, Логотип (64х64 пикс.), За- ставка (600х256 пикс.), Авторские права, Адрес информации о поставщи- ке, Адрес информации о конфигурации, Поставщик, Версия. Изменения отразятся в разделе О Программе (команда Справка > О программе…).

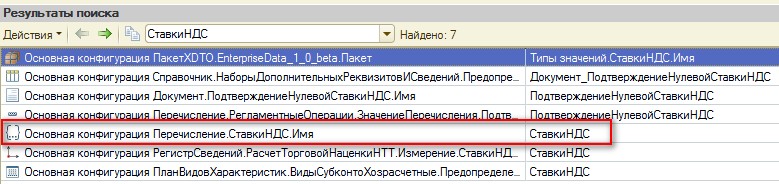
**Рис. 10 – Информация о программе**

1. Добавьте общий реквизит Комментарий (тип Строка, Неограни- ченная длина, включен Многострочный режим).
2. Создайте константы с общими сведениями об организации:

Таблица 5. Набор констант для конфигурации

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| НаименованиеПолное | Строка | длина = 250 |
| НаименованиеСокращенное | Строка | длина = 250 |
| ИНН | Строка | длина = 12, переменная |
| КПП | Строка | длина = 9, переменная |
| ДатаРегистрации | Дата |  |
| ЮридическийАдрес | Строка | неограниченная длина |
| Телефон | Строка | в формате с кодом стра- ны и города |
| ВебСайт | Строка |  |
| ПрофильПредприятия | Строка | неограниченная длина |

1. Создайте общую форму с типом Форма констант, разместите на ней все созданные константы. Для каждой константы заполните свой- ство Основная форма.
2. В режиме 1С:Предприятие заполните созданные константы по ва- риантам заданий. Как повлияло свойство Основная форма на способ редактирования констант?
3. Создайте перечисление СтавкиНДС.

**Совет.** Значения перечисления можно найти в конфигурации Бухгалте- рия предприятия учебная (демо). Для этого в режиме Конфигуратор запу- стите Глобальный поиск по текстам (Правка > Глобальный поиск…), в поле Искать достаточно указать СтавкиНДС и на вкладке Типы текстов оста- вить флаг Свойства. Для быстрого копирования найденных значений используйте drag-and-drop.

**Рис. 11 – Результаты поиска по свойствам объектов типовой конфигурации**

1. Создайте перечисления ТипыСкладов, ВидыДоговоров, ВидыЗаня- тости. Значения перечислений можно найти в конфигурации Бухгалте- рия предприятия учебная (демо) аналогично п.11.
2. Создайте перечисление ПолФизическогоЛица.
3. Создайте справочник Контрагенты по примеру, рассмотренному в описании лабораторной работы №1.
4. Создайте справочник Договоры. Владелец – справочник Контр- агенты (свойство Использование подчинения = Элементам). Реквизиты справочника:

Таблица 6. Структура справочника договоров

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| Дата | Дата |  |
| Номер | Строка | длина = 20, переменная |
| СрокДействия | Дата |  |
| ВидДоговора | ПеречислениеСсылка. ВидыДоговоров |  |
| Комментарий | Используйте общий реквизит из п.7 | |

Реквизиты Дата, Номер, ВидДоговора сделайте обязательными для за- полнения. Здесь и далее если рекомендуемые параметры по заданию не указаны, то допускается использовать настройки по умолчанию.

1. Создайте справочник ЕдиницыИзмерения. Нужно предусмотреть набор предопределенных основных единиц (в соответствии с профилем предприятия по варианту). Предопределенные единицы измерения задаются на вкладке Прочее. Реквизиты справочника:

Таблица 7. Структура справочника единиц измерения

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| НаименованиеПолное | Строка | длина = 40, переменная |

1. Создайте справочник Номенклатура. Справочник иерархический (иерархия групп и элементов), ограничение количества уровней иерархии снято. Реквизиты справочника:

Таблица 8. Структура справочника товаров и услуг

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| НаименованиеПолное | Строка | длина = 1000 |
| Артикул | Строка | длина = 25, переменная |
| ЕдиницаИзмерения | СправочникСсылка. ЕдиницыИзмерения |  |
| СтавкаНДС | ПеречислениеСсылка. СтавкиНДС | значениеЗаполнения = НДС20 |
| Услуга | Булево |  |
| Комментарий | Используйте общий реквизит из п.7 | |

Какие реквизиты целесообразно включить в построение индекса (свойство Индексировать)?

Задействуйте в механизме Ввод по строке созданный реквизит Наимено- ваниеПолное (Способ поиска строки = Любая часть). Какие требования предъявляются к реквизитам, участвующим в Вводе по строке?

1. Создайте справочник Подразделения. Для справочника устано- вите иерархию элементов, без ограничения количества уровней. Рек- визиты справочника:

Таблица 9. Структура справочника подразделений

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| НаименованиеПолное | Строка | длина = 250 |

1. Создайте справочник ФизическиеЛица. Для стандартного рекви- зита Наименование установите следующие свойства: Длина наименова- ния = 50, Синоним = ФИО. Добавьте реквизиты:

Таблица 10. Структура справочника физических лиц

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| ДатаРождения | Дата |  |
| Пол | ПеречислениеСсылка. ПолФизическогоЛица |  |
| ИНН | Строка | длина = 12, фиксированная |
| СтраховойНомерПФР | Строка | длина = 14, фиксированная |
| Адрес | Строка | длина = 240 |
| МестоРождения | Строка | длина = 240 |

Для реквизита СтраховойНомерПФР необходимо предусмотреть ввод в формате ‘ЧЧЧ-ЧЧЧ-ЧЧЧ ЧЧ’, где Ч – число.

**Совет.** Решить поставленную задачу можно с помощью свойства Мас- ка реквизита справочника. О том, какие символы может содержать Маска – читайте Справку (по запросу «ПолеВвода.Маска» в разделе Поиск справки).

1. Создайте справочник Должности. Длину кода для элементов справочника установите в 0. Реквизиты справочника:

Таблица 11. Структура справочника должностей

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Параметры** |
| НаименованиеКраткое | Строка | длина = 25, переменная |

1. Создайте справочник Сотрудники. Для стандартного реквизита Наименование установите следующие свойства: Длина наименования = 50, Синоним = ФИО. Набор реквизитов:

Таблица 12. Структура справочника сотрудников

|  |  |
| --- | --- |
| **Имя** | **Тип** |
| ФизическоеЛицо | СправочникСсылка.ФизическиеЛица |
| Подразделение | СправочникСсылка.Подразделения |
| ВидЗанятости | ПеречислениеСсылка.ВидыЗанятости |
| Должность | СправочникСсылка.Должности |
| ДатаПриема | Дата |
| ДатаУвольнения | Дата |

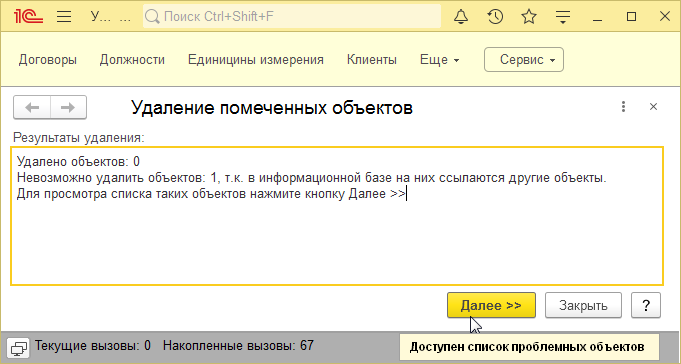
1. Создайте справочник Склады с реквизитами:

Таблица 13. Структура справочника складов

|  |  |
| --- | --- |
| **Имя** | **Тип** |
| ТипСклада | ПеречислениеСсылка.ТипыСкладов |
| Ответственный | СправочникСсылка.Сотрудники |

Включите использование общего реквизита Комментарий для данного справочника.

1. В режиме 1С:Предприятие заполните справочники согласно вари- антам задания правдоподобными сведениями. Количество элементов справочников не менее 5. Обратите внимание, как работают механиз- мы иерархии, подчинения и ввода по строке. Поэкспериментурийте с предопределенными элементами справочников (удалите элементы, измените наименование в режимах 1С:Предприятие и Конфигуратор).
2. В режиме 1С:Предприятие пометьте на удаление несколько эле- ментов справочника Должности. Удалите их, используя стандартный механизм платформы (Функции для технического специалиста… > Стан- дартные > Удаление помеченных объектов). Если возникают конфликты (см. рисунок 12), следует найти ссылки на помеченный объект, кото- рые препятствуют его удалению. Для этих целей освойте еще один стандартный механизм – Поиск ссылок на объекты.
3. Из режима Конфигуратор поменяйте Режим совместимости интер- фейса (свойство корневого элемента дерева метаданных). Запустите приложение, обратите внимание, как выглядит приложение с пользо- вательским интерфейсом Такси и Версия 8.2.
4. Запустив приложение с интерфейсом Такси, освойте команды Настроить список… и Изменить форму… (командная панель формы списка любого справочника > кнопка ).
5. Произведите архивирование (выгрузку) рабочей информацион- ной базы (из режима конфигурирования команда Администрирование > Выгрузить информационную базу…). Выгрузка базы выполняется после каждой лабораторной работы.



**Рис. 12 – Конфликт при удалении помеченных объектов**

# Факультативная часть (макс. оценка за выполнение - «отл.»)

1. Модифицируйте созданные структуры данных согласно пред- ставлениям о нормализованных БД.
2. Добавьте в конфигурацию новые структуры данных исходя из логики функционирования предприятия, профиль которого задан по варианту. В отчёте указать изменённые/добавленные объекты и обос- новать целесообразность внесенных изменений.

# Варианты выполнения лабораторной работы (тип предприятия)

Таблица 14. Тематика конфигураций

|  |  |
| --- | --- |
| № | Вариант задания (профиль предприятия) |
| 1 | Автосервис (производство запчастей, ремонт и модернизация авто- мобилей, продажа запчастей) |
| 2 | Жилищно-эксплуатационная организация (производство деталей, ремонт, продажа оборудования для ЖКХ) |
| 3 | Издательство (производство печатной продукции, её ремонт (пере- плёт), продажа и торговля своей продукции |
| 4 | Лесохозяйственное предприятие (рубка и посадка леса, первичная переработка древесины, продажа сырого леса и брёвен, торговля) |
| 5 | Общественное питание (производство блюд, ресторан, торговля продуктами питания) |
| 6 | Производственно-торговое предприятие (производство железобе- тонных изделий для строительства многоэтажных зданий) |
| 7 | Производственно-торговое предприятие (производство стеновых материалов для индивидуального жилищного строительства) |
| 8 | Производственно-торговое предприятие (производство кровельных материалов для строительства) |
| 9 | Производственно-торговое предприятие (производство материалов деревообработки для строительства) |
| 10 | Производственно-торговое предприятие (производство отделочных материалов для строительства) |
| 11 | Производственно-торговое предприятие (производство металличе- ских и пластиковых окон и дверей) |
| 12 | Машиностроительный завод (производство и торговля спортинвен- тарём) |
| 13 | Машиностроительный завод (производство и торговля газового оборудования) |
| 14 | Машиностроительный завод (производство и торговля сантехниче- ского оборудования) |
| 15 | Машиностроительный завод (производство и торговля садово-  огородного инвентаря) |
| 16 | Машиностроительный завод (производство и торговля сварочного оборудования) |
| 17 | Предприятие пищевой промышленности (производство и торговля мясопродуктами) |
| 18 | Предприятие пищевой промышленности (производство и торговля молочными продуктами) |
| 19 | Предприятие пищевой промышленности (производство и торговля кондитерскими изделиями) |
| 20 | Производственно-торговое предприятие (производство и торговля парфюмерно-косметическими изделиями) |

# Контрольные вопросы

1. Что такое система 1С:Предприятие?
2. Каковы области применения системы 1С:Предприятие?
3. Что понимается под конфигурируемостью системы?
4. Что такое прикладное решение (концигурация) и какие прикладные решения Вам известны?
5. Какие режимы работы существуют в 1С:Предприятии, в чем за- ключаются их особенности?
6. Какие существуют рекомендации по выбору оборудования для ра- боты системы 1С:Предприятие?
7. Выбор каких параметров производится во время установки систе- мы 1С:Предприятие?
8. Для чего используется установка по образцу?
9. Что такое административная установка?
10. Какие отличительные особенности установки 1С:Предприятия в варианте клиент-сервер?
11. Что такое информационная база?
12. Каким образом создается резервная копия информационной базы?
13. Что такое шаблон конфигурации и для чего он используется?
14. Как создаётся пустая информационная база?
15. Какими файлами представлена структура файловой информацион- ной базы?
16. Где хранятся настройки параметров информационных баз?
17. Что такое конфигурация и объект конфигурации?
18. Что такое дерево метаданных?
19. Для чего используется синтаксис-помощник?
20. Что такое палитра свойств и для чего она используется?
21. В чем отличие конфигурации от конфигурации базы данных в 1С:Предприятие?
22. Что такое константа в системе 1С:Предприятие?
23. Что такое справочник и как он создается?
24. Что такое табличная часть справочника?
25. Для чего используется свойство «синоним» в объектах конфигура- ции?
26. Перечислите стандартные функции для технического специалиста, доступные из режима 1С:Предприятие.

# Лабораторная работа №2.

**РАЗРАБОТКА КОНФИГУРАЦИИ: ОПРЕДЕЛЕНИЕ ПРАВ ДО- СТУПА К ФУНКЦИОНАЛЬНОСТИ СИСТЕМЫ, НАСТРОЙКА ИНТЕРФЕЙСНЫХ МЕХАНИЗМОВ, СРЕДСТВА ИНТЕГРАЦИИ**

*Цель работы:* освоение основных приемов работы с системой 1С:Предприятие, правил работы с формами, знакомство с назначением ролей и прав доступа, освоение методов проектирования интерфейса пользователя, изучение средств интеграции и взаимодействия с други- ми программными системами.

# Теоретические сведения

В предыдущей лабораторной работе были созданы константы, определяющие основные данные организации, а также ряд перечисле- ний и справочников. Платформа 1С:Предприятие 8 автоматически ге- нерирует формы объектов, имеющихся в составе конфигурации, для просмотра, ввода и изменения данных о хозяйственной деятельности предприятия. В лабораторной работе рассмотрим инструменты и ме- ханизмы, доступные разработчикам и пользователям при необходимо- сти переопределить стандартный вид и поведение элементов форм и пользовательского интерфейса.

## Концепция управляемого приложения и управляемого интерфейса

Начиная с версии 8.2 появляются два режима запуска 1С:Предприятие: Управляемое приложение и Обычное приложение. Ре- жим запуска задается через свойство конфигурации Основной режим запуска. Другое свойство, Режим совместимости, устанавливает возмож- ность использования механизмов выбранной версии платформы и предшествующих ей версий без необходимости отдельной установки нужной версии программы.

**Замечание**. Только толстый клиент может быть запущен в режиме обычного приложения. Конфигурация с установленным свойством Основной режим запуска = Обычное приложение по умолчанию запуска- ется на отладку из конфигуратора в данном режиме.

**Замечание**. Запрещена комбинация значений свойств конфигура- ции Основной режим запуска = Управляемое приложение и Режим совме- стимости = Версия 8.1.

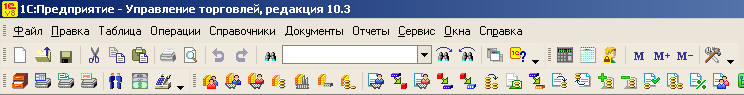
**Замечание**. Включение режима совместимости с версией 8.2.13 и ниже несовместимо с наличием в конфигурации общих реквизитов.

Принципиальные отличия между управляемым и обычным прило- жением состоят в следующем:

* для управляемого приложения предусмотрено либо физическое раз- деление на сервер и клиент, либо программная эмуляция;
* в управляемом приложении для просмотра, ввода и обработки дан- ных используются управляемые формы, в обычном приложении – обычные формы. Управляемые и обычные формы имеют разные принципы построения и функционирования;
* концепция построения пользовательского интерфейса для управляе- мого и обычного приложения координально отличаются.

Концепция пользовательского интерфейса в системе 1С:Предприятие неоднократно претерпевала изменения.

*Пользовательский интерфейс* – средство доступа пользователя к функциональности приложения, навигации между формами и выпол- нения различных действий.

*Обычный интерфейс*. В версиях платформы ниже 8.2 для создания пользовательских интерфейсов имеется специальный объект конфигу- рации Интерфейс. С его помошью все действия и объекты, к которым пользователь должен иметь доступ, можно сгруппировать в выпадаю- щие списки, а часть команд продублировать в командных панелях. Если конфигурация включает несколько сотен объектов, для разработ- чика отразить вручную попиксельно каждый из них в интерфейсах весьма трудоемко. Разработка обычного интерфейса ведется следую- щим образом: интерфейс «рисуется» отдельно, права на доступ к объ- ектам настраиваются отдельно. Поэтому вероятно возникнование си- туации, при которой объект вынесен в интерфейсе, но права на доступ к нему отсутствуют. Пользователь, в свою очередь, не имеет гибкого механизма изменения такого интерфейса (только возможность пере- ключаться между готовыми интерфейсами, включенными в состав конфигурации разработчиком).

**Рис. 13 – Обычный интерфейс (платформа 8.1) типовой конфигурации**

С появлением платформы 8.2 механизм формирования пользова- тельского интерфейса становится более гибким. Для обозначения но- вого интерфейса введен термин *командный интерфейс*, *управляемый интерфейс*, который нет необходимости прорисовывать вручную.

*Командный интерфейс* – это основное средство навигации пользова- теля по функциональности конфигурации. В системе 1С:Предприятие он строится на основе объектов Подсистемы. Разработчик должен со- здать в конфигурации иерархию подсистем, отражающую для пользо- вателя структуру функциональности прикладного решения. Все при- кладные объекты конфигурации (константы, справочники, документы, отчеты и т. д.) привязываются разработчиком конфигурации к подси- стемам. Иерархия подсистем должна отражать структуру функцио- нальности именно так, как она будет представляться пользователю. На основе списка подсистем и их состава платформа автоматически стро- ит командный интерфейс для пользователя. По умолчанию пользова- телю отображается структура прикладного решения с разделами и предоставляются стандартные команды доступа к функциональности прикладных объектов (вызов списков справочников, документов, от- крытие формы констант, отчетов, обработок и т. д.). Однако разработ- чик может настроить предлагаемое системой построение командного интерфейса (изменить порядок, видимость команд). Для этого предна- значен набор редакторов командного интерфейса, которые позволяют настраивать различные части интерфейса программы. В версиях плат- формы 8.2 и 8.3 конечный пользователь через режим 1С:Предприятие тоже имеет некоторые возможности по настройке внешнего вида ко- мандного интерфейса (отсюда термины *управляемое* приложение, *управляемый* интерфейс, *управляемые* формы).

Состав стандартных команд, доступных в командном интерфейсе (закрытие форм, запись данных, открытие списков, создание новых элементов справочников, открытие отчетов и т. д.), заложен на уровне технологической платформы. Но разработчик может создать свои ко- манды, которые будут включаться в командный интерфейс.

**Важно!** Платформа с версии 8.2 автоматически учитывает при по- строении интерфейса права пользователя, заданные с помощью объек- та конфигурации Роли. В соответствии с этим, из командного интер- фейса исключаются объекты и действия, на которые пользователю не были назначены права. Поэтому говорят, что командный интерфейс

«порождается» из прав пользователей на доступ к объектам конфигу- рации.

В версии платформы 8.3 интерфейсные механизмы вновь претерпе- вают изменения, развиваясь в русле концепции командного и управля- емого интерфейса. Основные преимущества нового интерфейса Такси:

* адаптация под мобильные устройства;
* улучшение возможностей навигации по приложению;
* настраиваемое пользователем рабочее пространство (по части взаим- ного расположения панелей и их состава команд);
* улучшение юзабилити отдельных элементов интерфейса;
* современный дизайн.

С версии платформы 8.3 появилась возможность настраивать Режим совместимости интерфейса (одноименное свойство конфигурации). Свойство определяет, какой интерфейс по умолчанию использует кли- ентское приложение и возможность/невозможность изменения этого режима средствами встроенного языка. Например, при режиме совме- стимости Такси. Разрешить Версия 8.2 клиентское приложение по умол- чанию использует интерфейс Такси, но возможно изменение режима средствами встроенного языка. Для режима совместимости Версия 8.2 клиентское приложение работает в интерфейсе версии 8.2, и измене- ние режима не возможно. Наличие режимов совместимости позволяет выполнять постепенный перевод конфигураций на новые версии.

В связи с тем, что развитие технологической платформы 1С:Предприятие 8.3 связано с охватом мобильных устройств, свойст- сва конфигурации пополнились пунктом Назначения использования. Свойство определяет, что конфигурация может быть использована на ПК и/или мобильном устройстве.

**Замечание**. При разработке конфигурации под мобильное устрой- ство доступны не все классы объектов (подробнее о мобильной плат- форме 1С:Предприятия [https://v8.1c.ru/platforma/mobilnaya-platforma-](https://v8.1c.ru/platforma/mobilnaya-platforma-1s-predpriyatiya/) [1s-predpriyatiya/](https://v8.1c.ru/platforma/mobilnaya-platforma-1s-predpriyatiya/)).

## Подсистемы как составляющая командного интерфейса

Управляемый интерфейс пользователя Такси, который генерируется платформой на основе имеющихся объектов, созданных в рамках 1-й лабораторной работы, требует доработки.

Средства платформа 1С:Предприятие 8.3 позволяют настроить ин- терфейс из пользовательского режима (с помощью команд Настройка навигации и Настройка действий из меню Функции, а также команд из меню Сервис и настройки > Настройки). Но значительных изменений в интерфейсе не добиться без использования режима Конфигуратор.

Объекты, влияющие на пользовательский интерфейс приложения, сгруппированы в ветви Общие дерева конфигурации. Основой постро- ения интерфейса являются объекты Подсистемы. Кроме этого, для настройки пользовательского интефейса служат Группы команд, Общие картинки, Элементы стиля, Стили.

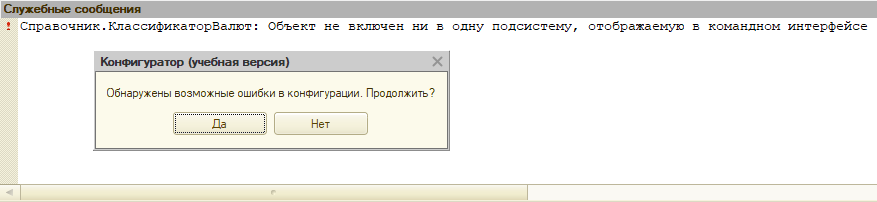
Подсистемы имеют разное назначение в обычном и управляемом приложении. В первом случае подсистемы позиционируются как ин-

струмент разработчика: распределив объекты по подсистемам, можно включить в дереве конфигурации отбор по подсистемам, тем самым скрыв ряд объектов. В управляемом приложении подсистемы опреде- ляют структуру функциональности прикладного решения именно так, как она будет представляться пользователю.

Подсистемы могут иметь дочерние элементы (количество уровней в иерархии подсистем технически не ограничено). Основные свойства подсистем – флаг Включать в командный интерфейс и Состав. Если флаг снят, то подсистема не будет отображаться в пользовательском интер- фейсе и останется лишь контейнером для группировки объектов в Конфигураторе. Для разрабатываемой конфигурации определим сле- дующие подсистемы:

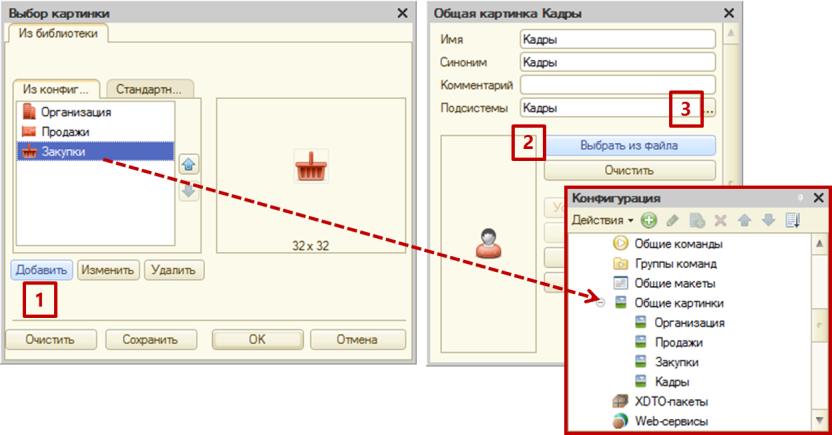
Таблица 15. Перечень подсистем для разработки учебной конфигурации

|  |  |
| --- | --- |
| **Подсистема** | **Состав подсистемы** |
| ОбОрганизации | Константы, форма констант |
| ОтделЗакупок | Справочники Контрагенты, Договоры, Едини- цыИзмерения, Номенклатура, Склады |
| ОтделПродаж | Справочники Контрагенты, Договоры, Едини- цыИзмерения, Номенклатура, Склады |
| Кадры | Справочники Подразделения, ФизическиеЛица, Должности, Сотрудники |

Если объект не включить ни в одну подсистему, то он не будет отображаться в пользовательском интерфейсе. Поэтому при сохране- нии конфигурации разработчик получает предупреждение, если новый объект не был добавлен в состав имеющихся подсистем.

**Рис. 14 – Случай, когда новый объект не включен в состав подсистем**

Для подсистемы можно установить картинку через одноименное свойство. Выбор производится из стандартного набора или путем до- бавления внешних графических файлов в конфигурацию. Картинки, добавленные в состав конфигурации, становятся доступны в пределах любых объектов, для которых предусмотрено сопоставление им гра-

фических изображений. Добавленные внешние файлы отображаются в ветви Общие > Общие картинки дерева конфигурации.

**Рис. 15 – Добавление картинки для подсистемы**

*1* – вызов диалога для добавления картинки в состав конфигурации;

* 1. – выбор графического файла с помощью стандартного диалога Windows;
  2. – поле, связывающее картинку с одной или несколькими подсистемами

## Определение прав доступа к объектам конфигурации

Настройка прав доступа к объектам конфигурации является одним из этапов конфигурирования. Для описания полномочий различных категорий пользователей на доступ к информации, обрабатываемой системой, предназначены общие объекты Роли. Как правило, роли со- здаются отдельно для каждого вида деятельности, и каждому пользо- вателю системы ставится в соответствие одна или несколько ролей. При создании ролей исходят из того, какие полномочия требуются различным группам пользователей на доступ к информации.

Механизм ролей, который изначально (до версии 8.2 платформы) служил только для назначения полномочий пользователям, приобрел в управляемом приложении дополнительный смысл: управление интер- фейсом. Другими словами, управляемый интерфейс не прорисовыва- ется разработчиком, а порождается из прав пользователей на доступ к объектам.

Редактор Ролей (вызывается автоматически при создании или ре- дактировании нового объекта Роль) содержит две области: слева, в списке объектов, перечислены все объекты и виды объектов конфигу- рации, а справа, в окне прав, перечислены доступные права для вы- бранного объекта или их совокупности.

Права бывают двух типов: основные (проверяются всегда, незави- симо от способа обращения к объекту ИБ) и интерактивные (проверя- ются при выполнении интерактивных операций, т.е операций в фор- мах). Осознанно нужно подходить к назначению права Интерактивное удаление, которое позволяет удалять данные в пользовательском ре- жиме без контроля ссылочной целостности.

Для одной из ролей обязательно следует установить полные права на доступ ко всем объектам и группам объектов конфигурации. Для разрабытываемой конфигурации определим набор ролей в соответ- ствии с таблицей:

Таблица 16. Набор ролей и прав доступа к объектам конфигурации

|  |  |  |
| --- | --- | --- |
| **Роль** | **Объекты** | **Права** |
| Администратор | Полные права на все созданные объекты   * Устанавливать права для новых объектов * Устанавливать права для реквизитов и табличных частей по умолчанию | |
| ОтделПродаж | * Устанавливать права для реквизитов и табличных частей по умолчанию | |
| Корневой объект | * Тонкий клиент * Веб-клиент * Мобильный клиент * Все режимы основного окна * Сохранение данных пользователя * Вывод |
| Подсистемы ОбОр- ганизации, Отдел- Продаж | * Просмотр |
| Общий реквизит  Комментарий | * Просмотр * Редактирование |
| Общая форма с константами | * Просмотр |
| Все константы | * Просмотр * Чтение |
| Справочники Контрагенты, Дого- воры, ЕдиницыИз- мерения, Номен- клатура, Склады,  Сотрудники | Все права, кроме прав на любое ин- терактивное удаление и работу с историей данных |
| ОтделЗакупок | Настройки аналогично роли ОтделПродаж | |
| Бухгалтерская | Корневой объект | Аналогично роли ОтделПродаж |
| Подсистемы ОбОр- | * Просмотр |

|  |  |  |
| --- | --- | --- |
| **Роль** | **Объекты** | **Права** |
|  | ганизации и Кадры |  |
| Общий реквизит  Комментарий | * Просмотр * Редактирование |
| Общая форма с констрантами | * Просмотр |
| Все константы | Полные права |
| Все справочники | Все права, кроме прав на любое ин- терактивное удаление и работу с  историей данных |

Чтобы иметь возможность отличать друг от друга пользователей, работающих с информационной базой, в системе 1С:Предприятие су- ществует штатный механизм ведения списка пользователей. В этом режиме администратор базы имеет возможность создавать и удалять пользователей системы, назначать им роли, язык интерфейса и др.

**Важно!** Перед добавлением пользователей в конфигурацию необхо- димо выполнить обновление конфигурации базы данных.

Добавление новых пользователей информационной базы выполня- ется в окне Список пользователей (меню Администрирование > Пользова- тели). Диалог для создания/изменения пользователя содержит поля Имя пользователя (это идентификатор, который будет появляться в окне выбора пользователей при запуске системы в режиме 1С:Предприятие); Полное имя (строка, которая может быть использо- вана внутри конфигурации при выводе различной справочной инфор- мации); варианты аутентификации пользователя. На второй вкладке для пользователя можно назначить одну или несколько ролей. Для пользователя, которому сопоставлено несколько ролей, интерфейс строится исходя из того, что права из разных ролей объединяются.

**Важно!** Первый пользователь ИБ обязательно должен иметь роль с правами на административные функции, иначе он не будет создан. Аналогично, нельзя удалить пользователя с административными функциями, если у всех остальных этих прав нет.

Для разрабатываемой конфигурации создадим следующих пользо- вателей:

Таблица 17. Пользователи и роли для них

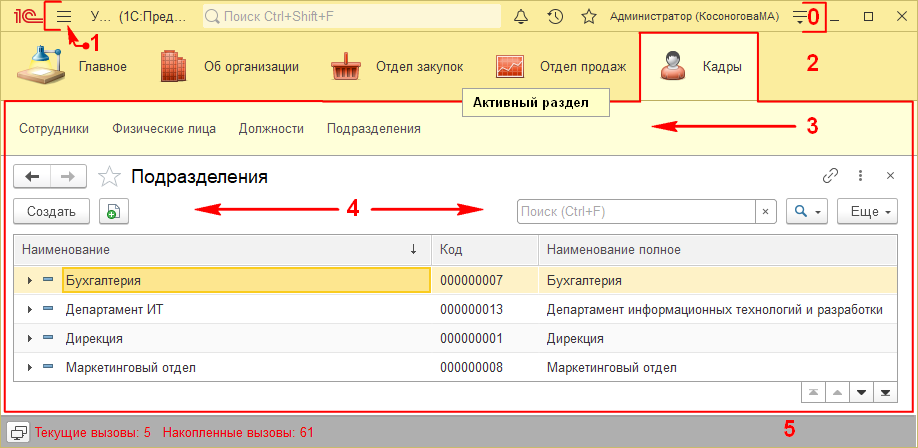
|  |  |
| --- | --- |
| **Полное имя пользователя** | **Назначенные роли** |
| Администратор (ФамилияИО) | Администратор |
| Менеджер по закупкам | ОтделЗакупок |
| Менеджер по продажам | ОтделПродаж |
| Главный бухгалтер | Бухгалтерская |
| Директор | Бухгалтерская |

После сохранения конфигурации и перезапуска приложения по- явится запрос имени пользователя и пароля.

Вся информация о пользователях хранится в общем файле базы данных, хотя до версии 8 пароли пользователей хранились в отдель- ном каталоге.

**Важно!** Учебная версия платформы 1С:Предприятие не рассчитана на работу с паролями пользователей.

## Конструирование пользовательского интерфейса

Добавление в конфигурацию подсистем приводит к появлению в пользовательском интерфейсе панели разделов. Заметим, разделы од- нозначно соответствуют первому уровню подсистем конфигурации.

**Рис. 16 – Панели интерфейса Такси**

*0 –* область системных команд; *1 –* меню Функции; *2* – панель разделов;

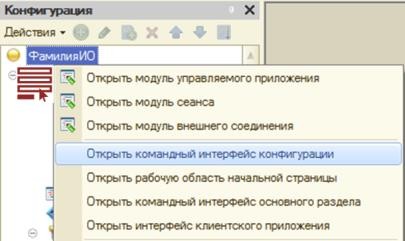
*3* – панель функций текущего (активного) раздела; *4* – командная панель формы списка;

*5* – информационная панель

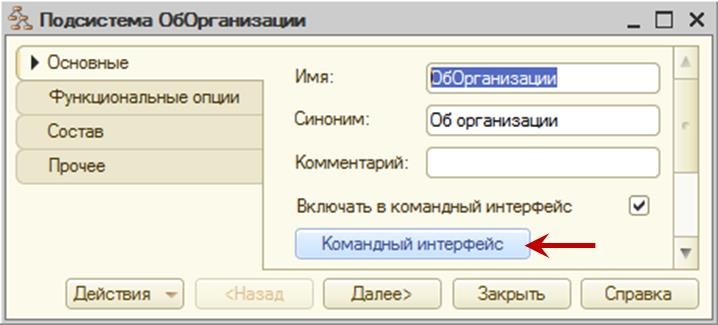
Первым в панели разделов расположен раздел Главное. Это аналог рабочего стола в операционных системах, предназначен для размеще- ния наиболее важной, часто используемой информации.

По умолчанию подсистемы в панели разделов и команды в панели текущего раздела располагаются в алфавитном порядке. Изменить по- рядок на произвольный можно как из конфигуратора через командный интерфейс, так и в пользовательском режиме:

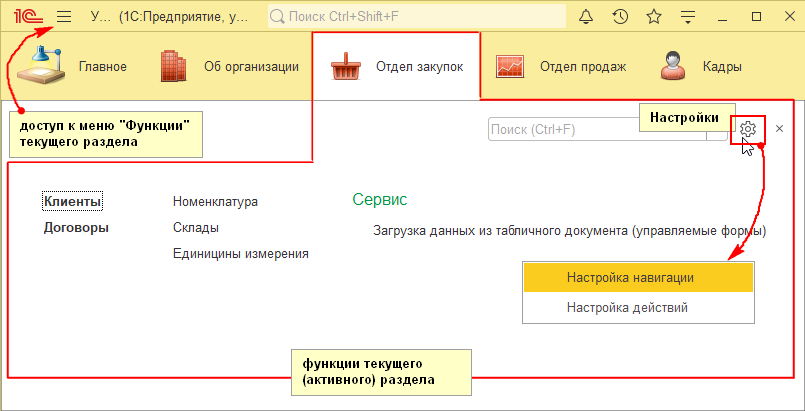
* Открыть командный интерфейс конфигурации (в режиме Конфигуратор, для изменения порядка следования разделов);



**Рис. 17 – Доступ к командному интерфейсу конфигурации**

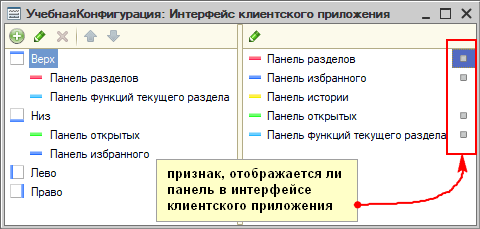
* Командный интерфейс подсистем (в режиме Конфигуратор, для измене- ния порядка следования команд внутри раздела, соответсвующего подсистеме);

**Рис. 18 – Доступ к командному интерфейсу отдельной подсистемы**

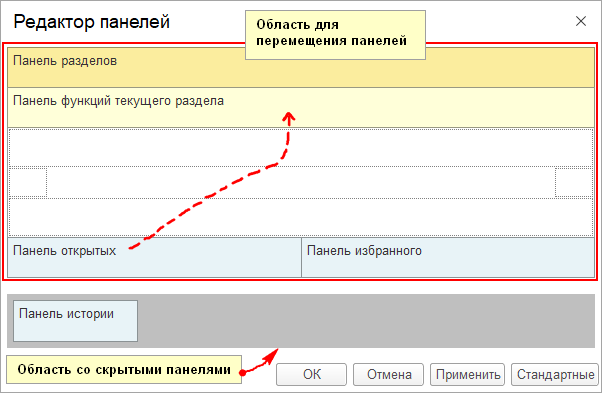
* Сервис и настройки > Настройки > Настройка панели разделов… (в ре- жиме 1С:Предприятие для изменения порядка следования разделов);
* Меню Функции > Настройки > Настройка навигации (в режиме 1С:Предприятие для изменения порядка следования команд текущего раздела).

**Рис. 19 – Доступ к настройкам функций текущего раздела в режиме 1С:Предприятие**

Взаимное расположение панелей в пользовательском интерфейсе также можно поменять (из Конфигуратора – командой Открыть интер- фейс клиентского приложения контекстного меню корневого элемента дерева метаданных; из режима 1С:Предприятие – командой Сервис и настройки > Настройки > Настройка панелей … ).

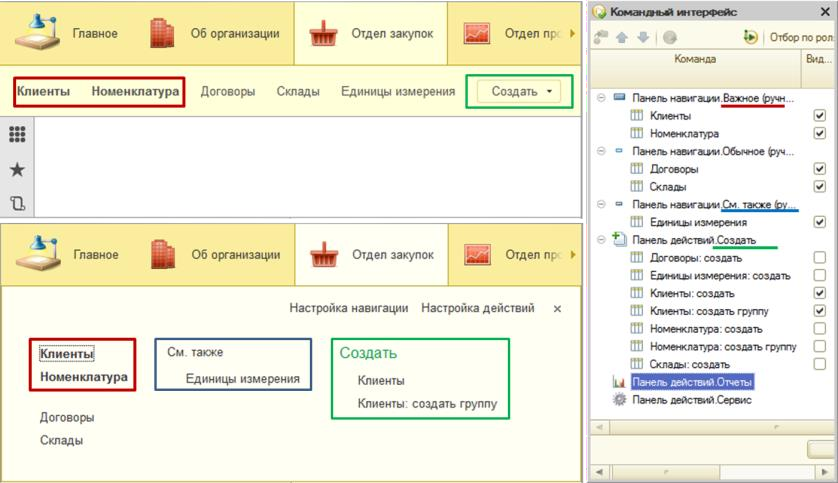
Настройки командного интерфейса и расположения панелей, вы- полненные в Конфигураторе, становятся стандартными настройками всех пользователей. Далее в режиме 1С:Предприятие каждый пользова- тель может самостоятельно конструировать своё рабочее простран- ство, изменяя стандартные настройки (к ним всегда можно вернуться).

**Рис. 20 – Редактор интерфейса со всеми стандартными панелями (инструмент разработчика)**



**Рис. 21 – Редактор панелей (инструмент пользователя)**

## Команды и группы команд

Обратимся к командному интерфейсу любой подсистемы. Для него характерно наличие стандартных групп команд (Важное, Обычное, См. также, Создать, Отчеты, Сервис), внутри которых помещены стандарт- ные команды (по созданию элементов и групп справочников, по от- крытию списков элементов справочников). Отметим, что с помощью объекта Группы команд (ветвь Общие дерева метаданных) к перечню стандартных групп можно добавить и свои.

**Рис. 22 – Стандартные группы команд в пользовательском интерфейсе**

Перемещение команд между группами команд позволяет влиять на вид пользовательского интерфейса.

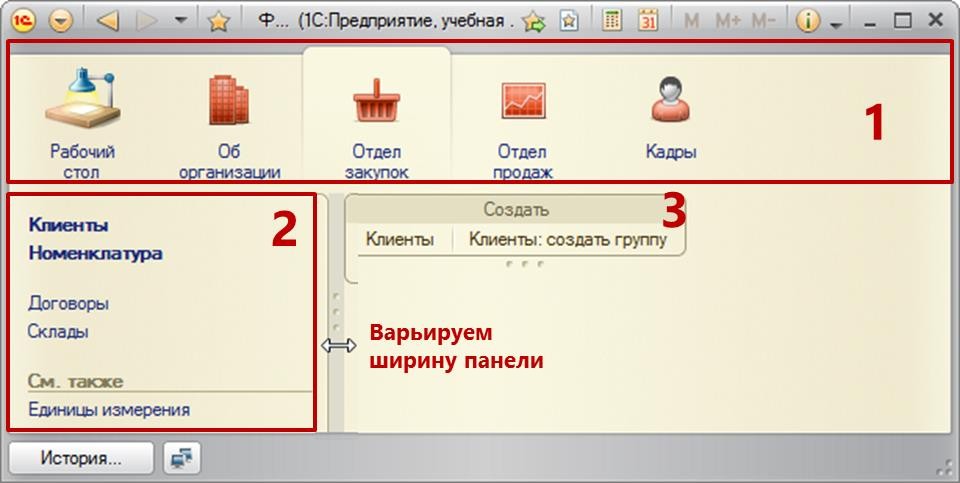
Далее целесообразно рассмотреть несколько признаков, по кото- рым можно классифицировать команды в системе 1С:Предприятие 8.

По отношению к тому, каким образом команда появилась в конфи- гурации, все команды можно разделить на стандартные и созданные раз- работчиком. Стандартные команды поставляются автоматически, платформой. Поставлять стандартные команды могут объекты конфи- гурации. Например, по рис. 22 видно, что справочник Клиенты может поставлять команды Клиенты, Клиент: создать и Клиент: создать группу. Разработчик может сам создать произвольные команды в конфигура- ции: добавить Общие команды в ветвь Общие, добавить объект Команда в конкретный объект конфигурации или описать команду в виде про- цедуры на встроенном языке в конкретной форме.

По отношению к тому, что является результатом выполнения ко- манды, все команды можно разделить на навигационные и команды дей- ствий. Навигационные команды открывают формы в текущем окне. Как правило, таким образом выполняется переход к каким-либо спискам. Например, навигационная команда Договоры (см. рис. 22) откроет спи- сок договоров. В результате выполнения команды действия открывается вспомогательное окно. Как правило, таким образом выполняется пере- ход к форме объекта или к форме отчета/обработки. Например, коман- да действия Клиенты: создать группу (см. рис. 22) откроет вспомогатель- ное окно для редактирования данных новой группы справочника Кли- енты.

Позже, при знакомстве с механизмом работы форм, будет рассмот- рен еще один признак классификации команд – по отношению команд к некоторой форме.

С точки зрения проектирования интерфейсов фирма 1С предлагает систему своих стандартов ([https://its.1c.ru/db/v8std#browse:13:-1:7](https://its.1c.ru/db/v8std#browse%3A13%3A-1%3A7)).

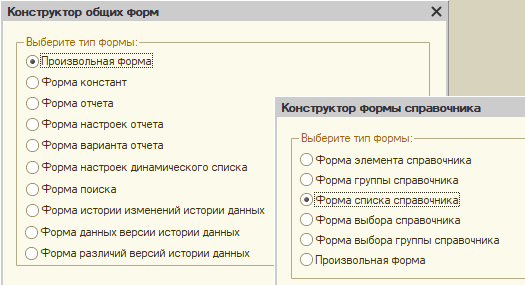
Для сравнения интерфейса Такси и интерфейса версии 8.2 покажем вид клиентского приложения с включенным режимом совместимости (Основной режим запуска = Управляемое приложение, Режим совместимо- сти интерфейса = Версия 8.2, Режим совместимости = Версия 8.2.16). Для пользовательского интерфейса версии 8.2 предусмотрена настройка панели разделов, панели навигации, панели действий – можно изме- нить порядок, видимость команд (через контекстное меню соответ- ствующей стандартной панели). Взаимное положение стандартных панелей жестко зафиксировано на уровне платформы (не может быть изменено ни пользователем, ни разработчиком), варьируется их шири- на.

**Рис. 23 – Вид управляемого интерфейса для версии 8.2**

*1* – панель разделов; *2* – панель навигации; *3* – панель действий

## Понятие «форма»

Форма – это специально созданный объект для ввода и просмотра какой-либо информации, а также для выполнения других действий исходя из контекста формы. Другими словами, форма – интерфейсный объект, посредством которого пользователь взаимодействует с данны- ми информационной базы.

Форма может быть подчиненной объекту конфигурации, либо об- щей (например, форма констант, распологающаяся в ветви Общие > Общие формы дерева метаданных). Создание форм выполняется в Кон- фигураторе с помощью одноименного конструктора. На этапе констру- ирования формы задается ее тип, который определяет дальнейшее по- ведение формы по умолчанию.

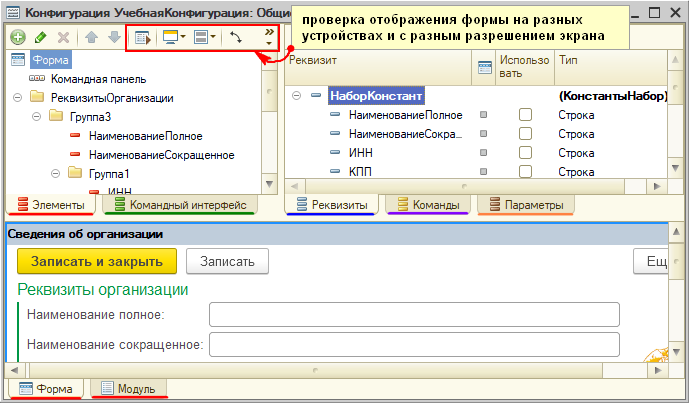
**Рис. 24 – Типы общих форм и форм, подчиненных справочнику**

На втором шаге в конструкторе следует указать набор реквизитов, которые будут отображаться на форме.

Все дальнейшие манипуляции выполняются в редакторе формы. При создании новой формы редактор вызывается автоматически (по- сле завершения работы конструктора). Редактор для существующей формы вызывается при ее открытии.

Редактор управляемой формы представлен несколькими вкладками (см. рис. 25).

Реквизиты. Вкладка содержит список реквизитов формы, в которых содержатся данные. Реквизиты формы используются, чтобы связать форму с реквизитами и табличными частями объекта, для которого конструируется форма. Жирным шрифтом выделяется так называемый Основной реквизит формы. Он определяет основную функциональность формы, стандартные команды, размещаемые на ней, и некоторые ав- томатические действия, которые будет делать платформа относительно этой формы. Основной реквизит зависит от типа формы, который ука-

зывается в конструкторе при первичном создании. В нашей учебной конфигурации на данный момент имеется форма с типом Форма кон- стант. (имя ее основного реквизита по умолчанию НаборКонстант).

**Рис. 25 – Редактор управляемой формы**

Элементы. Вкладка содержит дерево элементов управления, каждый из которых имеет связь с одним из реквизитов формы. Таким образом, элементы управления определяют способ отображения данных, содер- жащихся в реквизитах формы. Любая управляемая форма модифици- руется путем добавления элементов в дерево, перетаскивания их в пределах дерева. Можно воспользоваться следующими элементами управления: Группа, Поле, Кнопка, Таблица, Декорация (надпись, картин- ка), Дополнение элемента формы. Элемент управления характеризуется набором свойств и событий, настраиваемых в панели свойств [Atl+Enter].

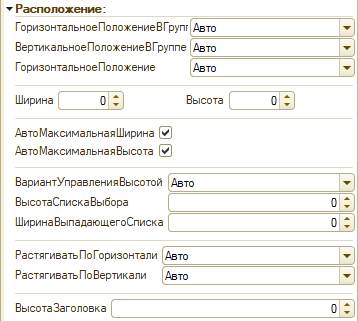
**Замечание**. Любой элемент управления – это контейнер для отобра- жения данных, и пустые элементы на форме не отображаются. Если элемент управления присутствует в дереве элементов, но не отобража- ется на форме, то у элемента отсутствует связь с реквизитом формы. В таком случае элементу попросту нечего отображать на форме. Для свя- зи элемента с данными используется свойство ПутьКДанным. Если пе- ретащить реквизит формы на вкладку Элементы, то элемент управления на форме размещается автоматически, а также устанавливается Путь- КДанным.

Форма. Вкладка представляет собой область предварительного про- смотра и отображает форму так, как ее увидит пользователь.

Модуль. Через вкладку доступен редактор текста программного мо- дуля. Программный модуль формы может содержать процедуры- обработчики событий формы и функции, написанные на встроенном языке системы 1С:Предприятие.

У форм, как и у конфигурации и подсистем, имеется командный ин- терфейс, который настраивается на одноименной вкладке редактора формы. Через эту вкладку можно модифицировать панель навигации формы и ее командную панель.

Таким образом, вид управляемых форм зависит от структуры дере- ва элементов управления и структуры прав доступа. При этом форма является не графическим объектом, а декларативным описанием в виде дерева. В связи с этим для управляемых форм не действуют абсолют- ные координаты; ширина и высота элементов управления задается не в пикселях, а в символах; положение элементов определяется относи- тельно друг друга. Все эти аспекты облегчают передачу управляемых форм по слабым каналам связи.

**Совет**. Для изменения взаимного расположения элементов на создан- ной форме нужные элементы необходимо объединить с помощью эле- мента управления Группа, для Группы настроить свойства Поведение, Отображение, Группировка, флаг Объединенная, СквозноеВыравнивание, ВыравниваниеЭлементовИЗаголовков, ГоризонтальныйИнтервал, Верти- кальныйИнтервал, ГоризонтальноеПоложение, ВертикальноеПоложение, РастягиватьПоГоризонтали, РастягиватьПоВертикали. Расположение само- го элемента в группе или на форме настраивается с помощью свойств категории Расположение:

**Рис. 26 – Свойства, определяющие расположение элемента на управляемой форме**

**Совет**. Чтобы посмотреть на создаваемую форму в рабочем режиме, можно обойтись без сохранения конфигурации базы данных и запуска приложения в режиме отладки. Для тестирования/проверки формы (в Конфигураторе) доступно сочетание клавиш [Ctrl+R].

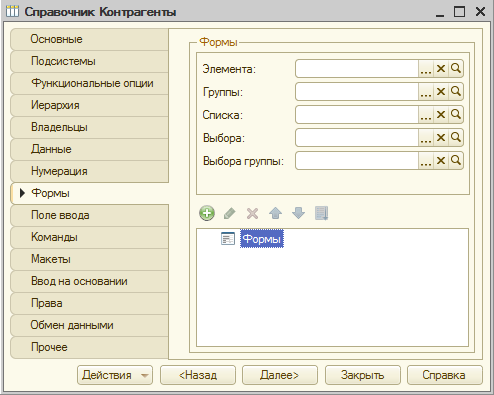
Для элементов управления на формах имеются свойства, которые влияют на отображение элементов только в мобильном клиенте, например, ВажностьПриОтображении. Такие свойства вводятся для упрощения адаптации управляемых форм под мобильный клиент. По- дробнее в разделе «Адаптация форм» ([https://its.1c.ru/db/metod8dev/](https://its.1c.ru/db/metod8dev/%0bcontent/5948/hdoc) [content/5948/hdoc](https://its.1c.ru/db/metod8dev/%0bcontent/5948/hdoc)).

Самый быстрый способ удалить форму (и любой другой объект конфигурации) – клавиша [Delete]. Однако нужно помнить: при удале- нии любого объекта конфигурации происходит автоматический кон- троль ссылочной целостности, и система не даст удалить объект, на который ссылаются другие объекты конфигурации. Если сейчас попы- таться удалить имеющуюся в конфигурации форму констант, плат- форма проинформирует, что «Объект не может быть удален, так как на него имеются ссылки в других объектах». И это верно, ведь ссылка на форму констант содержится в свойстве Основная форма всех констант. Поэтому перед удалением любого объекта конфигурации рекоменду- ется проверить наличие ссылок на него командой Поиск ссылок на объ- ект (Поиск ссылок в объекте) из контекстного меню объекта.

Важной особенностью системы 1С:Предприятие 8 является меха- низм форм по умолчанию. Этот механизм освобождает разработчика от необходимости создания всех возможных форм для каждого из объ- ектов прикладного решения. Разработчику достаточно создать новый объект прикладного решения, а система сама сгенерирует в нужные моменты работы пользователя необходимые формы по умолчанию для отображения информации, содержащейся в этом объекте. Таким обра- зом, разработчику нужно создавать собственные формы объектов при- кладного решения с нуля лишь в том случае, если они должны иметь отличия (другой дизайн или специфическое поведение) от форм, гене- рируемых системой по умолчанию. В предыдущей работе использова- лись именно формы по умолчанию (например, формы типа Список для отображения элементов справочника). Форм, создаваемых платформой по умолчанию «на лету», достаточно для выполнения простейших операций - заполнения созданных объектов данными, а также их про- смотра и редактирования.

## Настройка форм справочников

Рассмотрим процесс разработки экранных управляемых формы на примере справочника Контрагенты.

Откроем в конфигураторе справочник контрагентов и перейдём на вкладку Формы мастера объекта.

**Рис. 27 – Вкладка Формы мастера справочника**

Вкладка Формы справочника разделена на две области. В нижней области содержится список форм разных типов, подчиненных данному справочнику. В верхней области назначаются основные формы каждо- го типа: основная форма элемента, основная форма списка и т.д. (вы- бор осуществляется из нижнего списка). Назначить форму основной значит указать платформе, что именно эту форму нужно показывать пользователю. Если основную форму не назначить, то система создает ее «на лету» в процессе работы пользователя (см. конец раздела *Поня- тие «форма»*).

Создадим форму элемента справочника. Формы создаются в ниж- ней части окна, методика такая же, как и при создании реквизитов. По нажатию кнопки Добавить (из командной панели нижней области окна) запускается конструктор формы справочника.

Для справочника предусмотрены формы 6-ти типов (элемента, группы, списка, выбора, выбора группы, произвольная). В зависимости от выбранного типа формы некоторые поля настройки могут стано- виться недоступными. Выбираем тип Форма элемента справочника. Ва- риант Произвольная форма отличается от других тем, что не имеет Ос- новного реквизита, который определяет основную функциональность

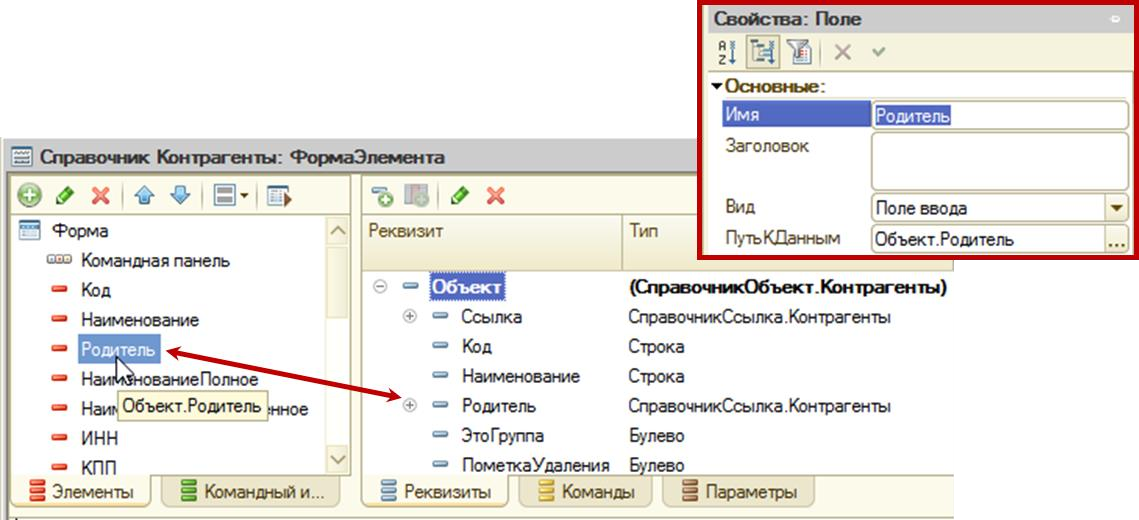
формы. Если установить флаг Назначить форму основной, она после создания автоматически будет присвоена соответствующему полю в списке основных форм. По флагу Основная форма элемента и группы данная форма станет основной формой и для элемента, и для группы иерархического справочника (продублируется).

Если в настройках параметров (Сервис > Параметры > Общие) ука- зан режим редактирования конфигурации Управляемое приложение и обычное приложение, то в конструктор формы добавляется еще одно поле выбора типа создаваемой формы (управляемая или обычная). При создании обычной формы становятся доступны флажки для размеще- ния на форме командных панелей.

Нажимаем кнопку Далее для настройки списка реквизитов, или Го- тово. Настройка списка реквизитов нужна тогда, когда требуется изме- нить состав реквизитов, отображаемых на форме. При создании обыч- ной формы на этом этапе доступно изменить тип элемента управления формы, связанного с реквизитом. По умолчанию, создается элемент Поле ввода, но можно поставить переключатель, флажок, надпись и т.п. в зависимости от типа данного реквизита.

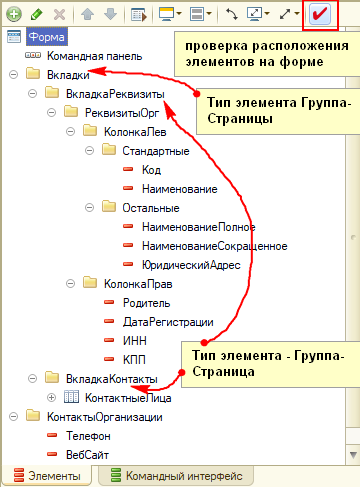
В открывшемся редакторе формы можем настроить ее внешний вид, изменив состав дерева элементов управления и их свойства.

Ранее отмечалось, что элементы управления выполняют свою зада- чу только в случае, если установлена связь между элементом управле- ния и реквизитом формы. Для этого предназначено свойство ПутьКДан- ным элемента управления.



**Рис. 28 – Иллюстрация связи элемента управления с реквизитом формы**

Отредактируем стандартный вид формы элемента справочника. Разместим элементы более компактно, разнеся данные о клиенте и его контактах по разным вкладкам. Для решения задачи дерево элементов управления должно иметь следующую структуру:



**Рис. 29 – Иллюстрация связи элемента управления с реквизитом формы**

Элементы управления распределяются по вкладкам с помощью вспомогательных элементов Группа-Страницы и Группа-Страница. Все типы элементов управления доступны по кнопке Добавить  (команд- ная панель вкладки Элементы). Перемещение элемента в группу осу- ществляется по методу drag&drop. Порядок элементов внутри группы можно изменить с помощью управляющих стрелок  и .

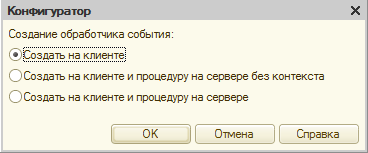
Управляемые формы с версии 8.3 платформы по умолчанию стро- ятся с расчетом на большие интервалы между полями. Если требуется привести форму к более компактному виду (например, из-за большого количества полей и других элементов управления), можно использо- вать настройку Вариант масштаба (Обычный, Компактный) в ее свойствах в редакторе формы.

Механизм управляемых форм позволяет проектировать интерфейс для толстых, тонких, мобильных, веб-клиентов в одном IDE. Управля- емые формы адаптивные и подстраиваются под текущий сеанс само- стоятельно.

Ни разработчику, ни пользователю нет необходимости заботиться о том, как на программном уровне создаются элементы и группы спра-

вочников, происходит открытие списков элементов справочников и запись их в базу данных. Все эти и другие команды поставляются платформой автоматически. Для изменения стандартного поведения объектов и их форм предназначен механизм обработки событий. Код обработчиков событий создается на встроенном языке программиро- вания системы 1С:Предприятие. Познакомимся с этими механизмами, организовав в разработанной форме автоматический запуск почтовой программы для отправки письма контактному лицу клиента.

*События* – это определенные изменения или действия, произошед- шие в системе, с формой или определенным элементом формы. Например: запуск системы, открытие формы, нажатие кнопки, обнов- ление и т.п. События для каждого объекта системы заранее жестко определены на уровне платформы.

Список предопределенных событий для любого элемента управле- ния приведен в его палитре свойств (категория События). Кнопка , расположенная справа от поля с наименованием события, служит для создания / просмотра процедуры-обработчика этого события. Нажатие на «лупу» запускает конструктор создания обработчика события (если процедура-обработчик события еще не созадана), либо позволяет пе- рейти в модуль формы к нужному фрагменту кода.

**Рис. 30 – Конструктор для создания обработчика события**

Отметим, что конструктор создания обработчика события появился с версии 8.3 платформы. Ранее в модуле формы автоматически созда- валась одна процедура, выполняющаяся на клиенте.

После выбора одного из вариантов в модуль формы добавляется каркас процедур(ы)-обработчика события.

При использовании конструктора название процедуры-обработчика события формы складывается по шаблону «Имя элемента управле- ния+Имя события», но может быть и произвольным.

Описанию каждой процедуры/функции в модуле формы предше- ствует *директива компиляции* (&НаКлиенте, &НаСервере, др.). Эта конструкция определяет место выполнения программного кода:

&НаКлиенте

Процедура КонтактныеЛицаАдресЭлПочтыНачалоВыбора

(Элемент, ДанныеВыбора, СтандартнаяОбработка)

// Вставить содержимое обработчика.

КонецПроцедуры

Если директива компиляции не указана, по умолчанию считается, что подпрограмма выполняется &НаСервере.

Вернемся к реализации возможности отправки письма контактному лицу клиента. Для этого открываем свойства элемента управления Кон- тактныеЛицаАдресЭлПочты табличной части КонтактныеЛица и в катего- рии Использование устанавливаем свойство КнопкаВыбора = Да. Далее создаем обработчик для события НачалоВыбора с командой:

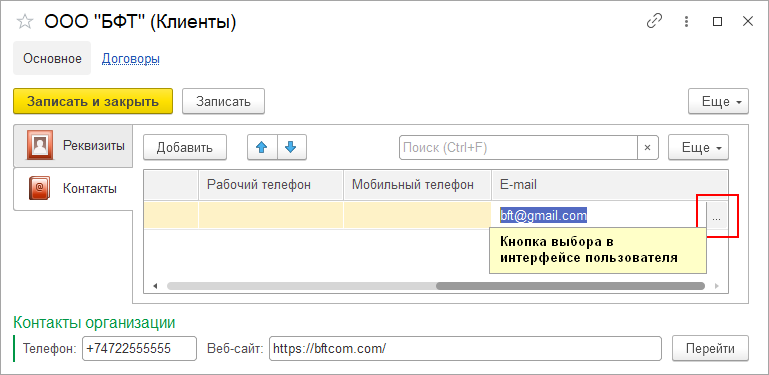
//далее обращение к данным табличной части КонтактныеЛица

//через объект Элементы АдресПочты=Элементы.КонтактныеЛица.ТекущиеДанные.АдресЭлПочты; Если НЕ ЗначениеЗаполнено(АдресПочты) Тогда

Сообщить("Укажите адрес почты контактного лица!"); Иначе

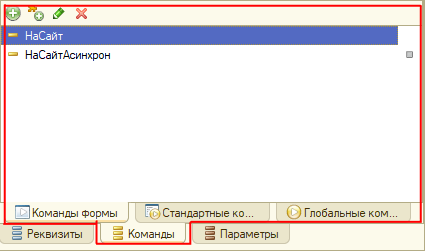
Оповещение = Новый ОписаниеОповещения(); НачатьЗапускПриложения(Оповещение,"mailto:"+АдресПочты,,);

КонецЕсли;

Теперь при выборе конкретного e-mail при двойном щелчке мыши будет появляться кнопка выбора, нажав на которую мы запустим поч- товый клиент.

**Рис. 31 – Кнопка для запуска почтового клиента**

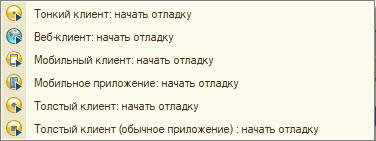
Единственным элементом управления, для которого не предусмот- рено списка событий в свойствах, является Кнопка. Для обработки нажатия элементу Кнопка сопоставляется команда (через свойство эле-

мента управления ИмяКоманды). Команды создаются на одноименной вкладке редактора управляемой формы.

**Рис. 32 – Источники команд для кнопок**

Как видно на вкладке Команды, по отношению команд к некоторой форме, все команды можно разделить на локальные и глобальные. Гло- бальные команды не являются частью формы, они поставляются плат- формой и могут быть включены в эту или в другие формы. Локальные команды формы являются частью самой формы и не могут быть вклю- чены в другие формы. Локальные команды поставляются формой и ее элементами (Стандартные команды), а также создаются разработчиком в форме (Команды формы).

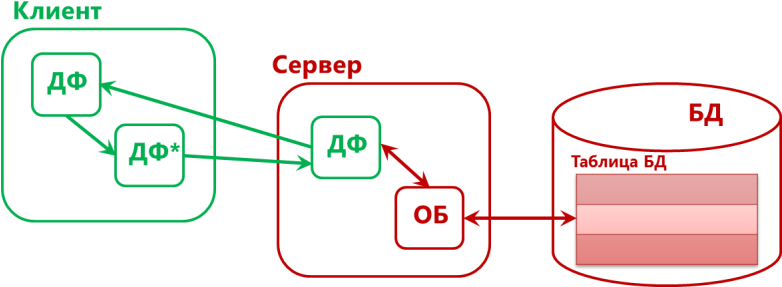
Если перетащить команду в область дерева элементов управления, то кнопка будет создана автоматически.

Поскольку разработка сложных конфигураций связана с програм- мированием, платформа предоставляет средства отладки программных модулей (в Конфигураторе меню Отладка): точки останова, точки ос- нова с условием, замер производительности и др. Запуск приложения в режиме отладки производится по нажатию [F5]. Режим отладки преду- смотрен для всех видов клиентских приложений (меню Отладка > Начало отладки).

**Рис. 33 – Варианты запуска в режиме отладки**

## Механизм работы управляемых форм

Рассмотрим механизм работы управляемой формы на примере формы элемента справочника. Пользователь дает команду на открытие формы. Далее выполняются следующие действия:



**Рис. 34 – Механизм работы управляемой формы**

*БД* – база данных; *ОБ* – объект БД;

*ДФ* – данные формы; *ДФ\** – модифицированные на клиенте данные

Сервер 1С:Предприятие обращается к БД, происходит выборка за- писи (соответствующей элементу справочника) из таблицы (соответ- ствующей справочнику).

При сборке формы на сервере платформа:

* получает объект БД (который не может существовать на стороне клиента),
* автоматически преобразует ОБ в некое «представление» (ДФ), до- ступное и на сервере, и на клиенте,
* помещает ДФ в основной реквизит формы элемента справочника.

Преобразование ОБ в ДФ/ДФ в ОБ автоматическое, но есть и явные методы.

Далее копия формы предается на сторону клиента, пользователь изменяет данные на форме. При выполнении команды Записать и За- крыть происходит серверный вызов, и форма в своем основном рекви- зите «приносит» с клиента на сервер все изменения, сделанные поль- зователем. ДФ\* и ДФ синхронизируются. Выполняется автоматиче- ское преобразование ДФ в ОБ. Посредством ОБ изменения сбрасыва- ются в базу.

**Важно!** Таким образом, форма содержит собственную копию данных из базы (это контекст формы). У любого объекта несколько состояний: на форме на стороне клиента, на форме на стороне сервера, в базе.

При программировании управляемых форм (и разработке управля- емого приложения в целом) необходимо помнить, что программный код должен разрабатываться в рамках клиент-серверного подхода.

Как было сказано выше, данные формы содержатся в ее основном реквизите. В модуле формы &НаКлиенте к этим данным можно напрямую обращаться, изменять их. Иначе происходит работа с поля- ми ссылочного типа. Например, на форме элемента справочника Со- трудники поле ФизическоеЛицо имеет тип СправочникСсыл- ка.ФизическиеЛица. Эта ссылка представлена на форме в виде наимено- вания. В модуле формы &НаКлиенте по этой ссылке напрямую мы не сможем получить никаких данных (пол, дату рождения, ИНН и др.), поскольку в реквизите формы есть только ссылка на данные, но не сами данные. Данные по ссылке можно получить, обратившись на сер- вер, например, так:

&НаКлиенте

Процедура ФизическоеЛицоПриИзменении(Элемент) Сообщить(ПолучитьИнфоНаСервере(Объект.ФизическоеЛицо));

КонецПроцедуры

&НаСервереБезКонтекста

Функция ПолучитьИнфоНаСервере(ссылкаНаФизлицоСФормы) Возврат ссылкаНаФизлицоСФормы.МестоРождения;

КонецФункции

Разберем приведенный листинг.

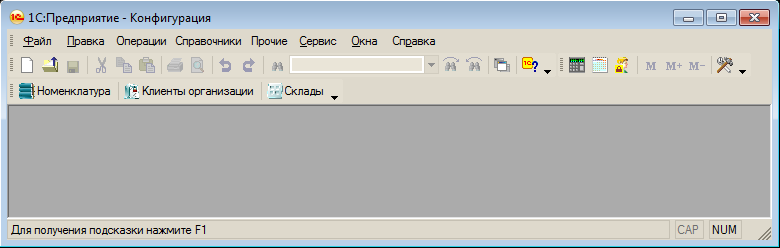
Клиентская процедура – это обработчик события ПриИзменении элемента ФизическоеЛицо. В этой процедуре вызывается встроенная функция Сообщить (выводит сообщение пользователю) с параметром, содержащим ссылку, взятую из реквизита формы. По ссылке на серве- ре возможно получить место рождения физического лица. Серверная процедура описана директивой &НаСервереБезКонтекста. Это означа- ет, что при ее вызове на сервер передается только управление и пара- метр процедуры, но не больше (контекст формы не передается на сер- вер при таком описании).

## О разработке обычного интерфейса и обычных форм

Как отмечалось ранее, управляемый интерфейс, появившийся с версии 8.2 системы 1С:Предприятие, формируется следующим обра- зом. Разработчик описывает возможность отображения команд, по- ставляемых платформой или созданных самим разработчиком, в зара- нее определенных элементах интерфейса прикладного решения (име- ется в виду панель разделов, панель действий, панель навигации, ко- мандные панели форм объектов конфигурации и т.п.). Окончательное же построение конкретного пользовательского интерфейса возлагается на платформу, которая «анализирует» роли, назначенные пользовате- лю, параметры, настроенные в конфигурации, а также настройки, вы-

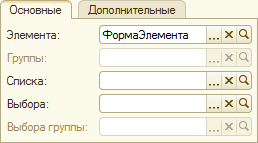
полненные самим пользователем, формируя индивидуальный интер- фейс с исключением ненужной для данного пользователя функцио- нальности.

Кратко рассмотрим процесс построения обычного интерфейса и обычных форм. Средства работы с обычным интефейсом и формами становятся доступны при изменении свойства Редактирование конфигу- рации для режимов запуска с варианта Управляемое приложение на Управляемое приложение и обычное приложение (меню Сервис > Пара- метры > Общие). Кроме того, следует изменить свойство конфигурации Основной режим запуска на Обычное приложение. После этого, во- первых, в дереве конфигурации появляется дополнительный объект – Интерфейсы. В обычном приложении эти объекты используются для проектирования индивидуальных пользовательских интерфейсов в виде наборов команд главного меню и панелей инструментов.

При таком подходе интерфейсы являются статическими наборами картинок, с которыми связаны те или иные действия. Такой подход к организации интерфейса не позволяет динамически его перестраивать в зависимости от функционирования прикладного решения.

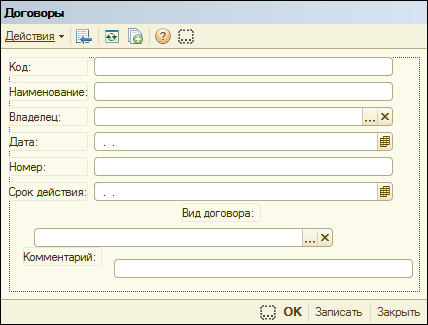
**Рис. 35 – Обычный интерфейс**

Как правило, для каждой категории пользователей создается свой интерфейс, который ставится в соответствие конкретному пользовате- лю. В отличие от ролей, каждому пользователю можно назначить только один интерфейс по умолчанию, однако средствами встроенного языка можно управлять видимостью других интерфейсов.

Во-вторых, на вкладке Формы мастера справочника появляется об- ласть дополнительных форм:

**Рис. 36 – Основные и дополнительные формы справочника**

В разделе Формы > Дополнительные можно назначить обычные формы элемента, группы, списка справочника и т.д., которые будут использоваться в случае запуска конфигурации в режиме обычного приложения.

Редактор обычной формы состоит из трех вкладок: Форма, Модуль, Реквизиты. Элементы управления можно свободно и абсолютно произ- вольно перемещать (Drag&Drop) непосредственно на вкладке Форма редактора:

**Рис. 37 – Вид обычной формы элемента справочника Договоры**

При добавлении новых элементов на обычную форму можно поль- зоваться мастером Вставки элемента управления - кнопка  главной панели управления, а также мастером размещения данных - кнопка .

При оформлении дизайна обычной формы очень удобно пользо- ваться выравнивающими линиями, которые можно настроить, вызвав контекстное меню формы.

В версии 8 появилась возможность настраивать поведение элемен- тов обычной формы при изменении ее размеров. Это делается при по- мощи механизма привязок. Привязки конкретного элемента формы настраиваются в его палитре свойств в разделе Расположение > Привяз- ка границ.

## Механизмы интеграции

Система 1С:Предприятие предоставляет объекты конфигурации, механизмы и средства встроенного языка для интеграции конфигура- ций между собой и со сторонними прикладными системами. Типовые конфигурации, решающие смежные задачи, содержат готовые обра- ботки по обмену данными между собой. Кроме того, поддерживается интеграция конфигураций 1С с торговым оборудованием (сканеры штрих-кодов, контрольно-кассовая техника), налоговыми (проверка контрагентов) и банковскими сервисами (выгрузка/загрузка банков- ских документов).

Технологическая платформа 1С:Предприятие предоставляет сле- дующие механизмы интеграции.

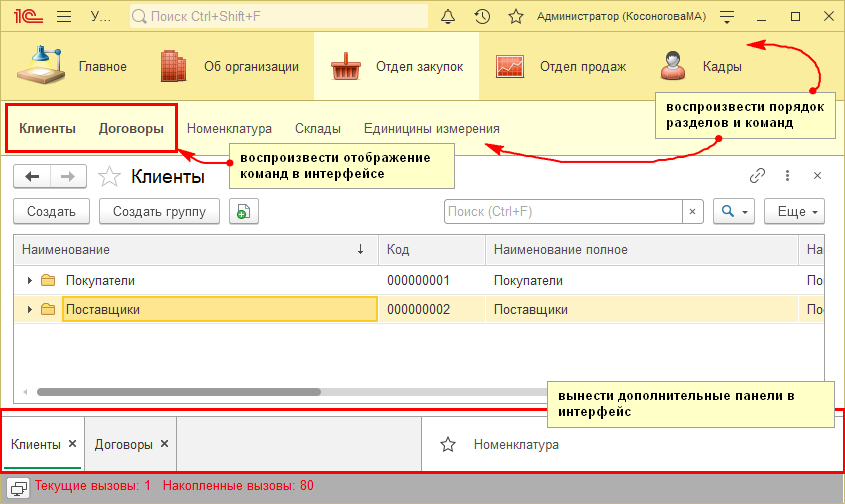
* Импорт/экспорт файлов (текстовые, XML, JSON, …) между прило- жением 1С и сторонним приложением. Например, реализован меха- низм сериализации прикладных объектов конфигурации в XML как напрямую, через методы встроенного языка ЗаписатьXML/ЧтениеXML, либо с помощью вспомогательного объекта XDTO-пакета (XML Data Transfer Objects).
* Объекты конфигурации HTTP-сервисы, Web-сервисы. Позволяют орга- низовать двухсторонний обмен данными со сторонним приложением
  + конфигурации могут реализовывать свои сервисы и вызывать внешние. Набор HTTP- и Web-сервисов, созданный в конфигурации, выступает в роли API, которое доступно внешним приложениям для обмена данными с 1С.
* REST интерфейс. Объект конфигурации можно сделать доступным для получения и модификации через REST интерфейс путем публи- кации на веб-сервере. Используется протокол доступа OData 3.0. Удобство механизма в том, что для работы с конфигурацией 1С че- рез протокол OData не требуется ее доработка.
* Объект конфигурации Внешние источники данных. Позволяет взаимо- действовать с таблицами любой ODBC-совместимой базы данных на чтение/запись.
* Объект конфигурации Планы обмена. Закрывает задачи по созданию распределенных информационных баз и обмену данными с инфор- мационными системами, не основанными на технологиях 1С:Предприятие. План обмена описывает состав данных, участвую- щих в обмене, и узлы обмена (отдельные конфигурации или прило- жения). Информационный обмен построен на основе XML- сообщений, которые помогают синхронизировать узлы.
* Технология внешних компонент. Позволяет создавать динамически подключаемые модули (С++), расширяющие функции 1С:Предприятия.

Система 1С:Предприятие 8.3 продолжает поддержку нескольких устаревших механизмов интеграции для обратной совместимости с приложениям, написанными под более ранние версии платформы. Среди них работа с файлами DBF из встроенного языка и два способа интеграции с Windows-приложениями на основе COM-технологии (Automation-сервер, внешнее соединение).

Подробности о перечисленных механизмах описаны в блоге ком- пании 1С (<https://habr.com/ru/company/1c/blog/308420/>) и на официаль- ном сайте (<https://v8.1c.ru/platforma/integraciya/>).

# Содержание работы

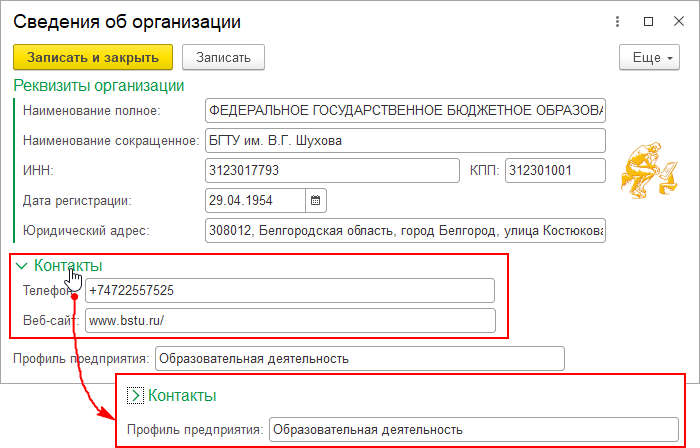
Задания Intermediate (макс. оценка за выполнение - «удовлетворит.»)

1. Добавьте в состав конфигурации подсистемы (по табл. 15).
2. Определите для конфигурации набор ролей и список пользо- вателей. Опирайтесь на таблицы 16, 17.
3. Приведите пользовательский интерфейс к следующему виду:

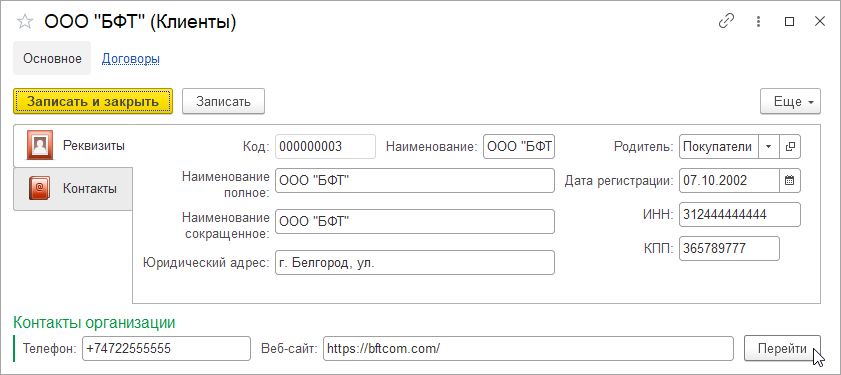
**Рис. 38 – Настроенный интерфейс Такси для пользователя Администратор**

Для остальных пользователей настройте интерфейс на свое усмотре- ние. Необходимые настройки доступны как из режима Конфигуратор (команды для корневого элемента дерева конфигурации), так и в ре- жиме 1С:Предприятие (команды из меню Сервис и настроки).

1. Измените стиль оформления приложения. Для этого добавьте в состав конфигурации объект Стиль (ветка Общие) и настройте все элементы (цвет активности, акцентирования, навигации, дополнитель- ной навигации и важного). Назначение нового стиля вместо стиля по умолчанию выполняется через свойство конфигурации ОсновнойСтиль. Изменение стиля оформления удобно в том случае, когда требуется использовать корпоративные цвета в пользовательском интерфейсе.
2. Измените стандартный вид формы констант, созданной в ла- бораторной работе №1. Элементы управления расположите в соответ- ствии с рисунком. При нажатии на заголовок Контакты группа полей Телефон, Веб-сайт должна сворачиваться.



**Рис. 39 – Вид в пользовательском режиме созданной формы констант**

1. Для справочника Контрагенты создайте форму элемента. Эле- менты формы и их представление настройте согласно рисунку. Обра- тите внимание, что на форме расположено несколько вкладок, рекви- зиты разбиты на два столбца, заголовки полей выравнены по правому краю.

**Рис. 40 – Вид формы элемента справочника**

Также требуется организовать на форме обработку нескольких собы- тий: 1) выполните пример из текста по запуску почтового клиента из строки табличной части КонтактныеЛица элемента справочника Контр- агенты; 2) организуйте из формы переход на веб-сайт контрагента по нажатию кнопки.

//использование синхронных1 методов

//Предупреждение() и ЗапуститьПриложение()

//требует предварительного изменения свойств конфигурации:

//Режим модальности = Использовать

//Режим использования синхронных вызовов… =Использовать &НаКлиенте

Процедура НаСайт(Команда)

Если НЕ ЗначениеЗаполнено(Объект.ВебСайт) Тогда Предупреждение("Укажите веб-сайт!");

Иначе

ЗапуститьПриложение(Объект.ВебСайт); КонецЕсли;

КонецПроцедуры

//Асинхронный вариант процедуры НаСайт &НаКлиенте

Процедура НаСайтАсинхрон(Команда)

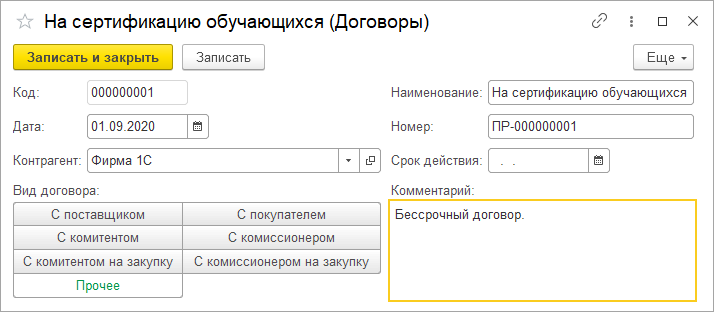
Если НЕ ЗначениеЗаполнено(Объект.ВебСайт) Тогда ПоказатьПредупреждение(,"Укажите веб-сайт!",,);

Иначе

Оповещение = Новый ОписаниеОповещения(); НачатьЗапускПриложения(Оповещение,Объект.ВебСайт,,);

КонецЕсли;

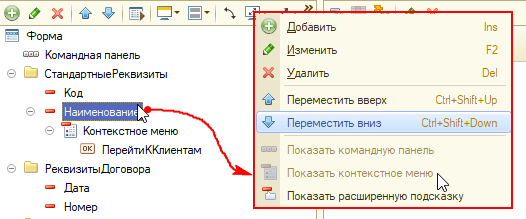
КонецПроцедуры

1. Создайте форму элемента справочника Договоры:

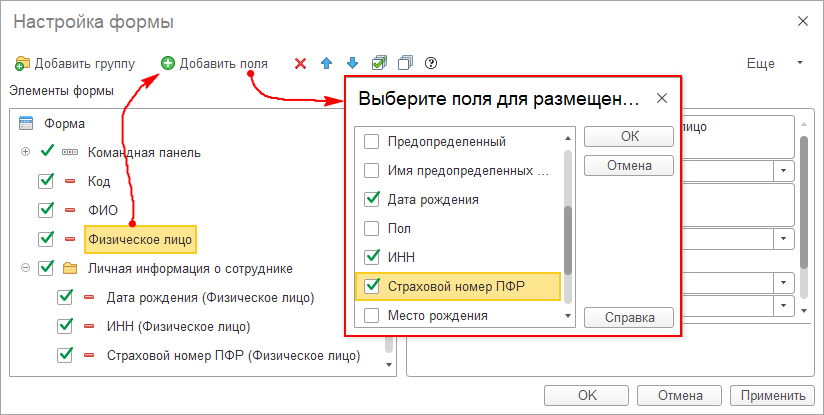
**Рис. 41 – Вид формы элемента справочника Договоры**

1. Создайте и настройте по своему усмотрению экранные формы разных типов (элемента, списка, выбора) для остальных справочников,

1 Об отличиях между синхронными и асинхронными методами [https://1c-](https://1c-programmer-blog.ru/programmirovanie/sinhronnye-i-asinhronnye-vyzovy-v-1s.html) [programmer-blog.ru/programmirovanie/sinhronnye-i-asinhronnye-vyzovy-v-](https://1c-programmer-blog.ru/programmirovanie/sinhronnye-i-asinhronnye-vyzovy-v-1s.html) [1s.html](https://1c-programmer-blog.ru/programmirovanie/sinhronnye-i-asinhronnye-vyzovy-v-1s.html)

созданных в первой лабораторной работе. В том числе нужно преду- смотреть наличие контекстного меню с произвольным составом ко- манд для полей на формах справочников (поля и формы на ваше умот- рение). В примере на рисунке для элемента Наименование отображено меню со стандартной командой по открытию справочника клиентов.

**Рис. 42 – Настройка команд контекстного меню в редакторе формы**

1. Изучите возможности по пользовательской настройке управ- ляемых форм в режиме 1С:Предприятие (через командную панель формы справочника > Еще > Изменить форму…). Для полей ссылочного типа (на примере любого справочника) добавьте на форму связанные поля, объедините их в группу (без использования редактора формы Конфигуратора).

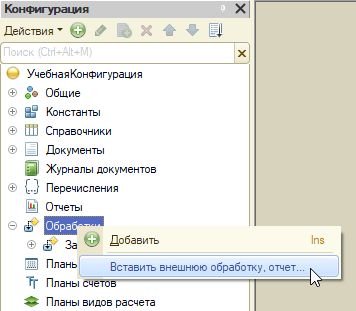
**Рис. 43 – Пример настройки управляемой формы в режиме 1С:Предприятие**

1. Освойте штатный механизм выгрузки списка элементов спра- вочника в табличный/текстовый документ (командная панель формы списка справочника > Еще > Вывести список…).

Задания Upper-Intermediate (макс. оценка за выполнение - «хорошо»)

1. Создайте и настройте интерфейс (с главным меню, панелями инструментов и контекстным меню) для доступа к информационной базе в режиме обычного приложения. Настройте обычные формы для отображения элементов двух произвольных справочников.
2. Усовершенствуйте программный код из п.6 по переходу на веб-сайт контрагента или запуску почтового клиента. Необходимо предусмотреть информирование пользователя о вводе некорректного адреса (без домена верхнего уровня, без символа @ или др.). Задание можно реализовать с использованием регулярных выражений2 или через объект встроенного языка HTTPЗапрос.

Задания Advanced (макс. оценка за выполнение - «отлично»)

1. Настройте загрузку данных в любой справочник из файла xls с помощью готовой обработки ЗагрузкаДанныхИзТабличногоДокумента.epf. Выбирать следует справочник, содержащий реквизиты ссылочного типа. Алгоритм использования универсальной обработки описан в ста- тье [https://programmist1s.ru/zagruzka-nomenklaturyi-v-1s-8-3-iz-excel-ili-](https://programmist1s.ru/zagruzka-nomenklaturyi-v-1s-8-3-iz-excel-ili-dbf/) [dbf/.](https://programmist1s.ru/zagruzka-nomenklaturyi-v-1s-8-3-iz-excel-ili-dbf/) Использовать обработку в своей конфигурации можно двумя спо- собами: 1) открыть файл обработки в режиме 1С:Предприятие через меню Сервис и настройки > Файл > Открыть… ; 2) добавить файл обра- ботки в состав конфигурации через ветку Обработки дерева конфигу- рации (см. рис.). При использовании 2-го способа доступен для изуче- ния и изменения программный код обработки.

**Рис. 44 – Добавление файла обработки в состав конфигурации**

2 Использование регулярных выражений (RegExp) в 1С8.х [Электронный ресурс]. – Режим доступа: <https://infostart.ru/public/64222/>

# Контрольные вопросы

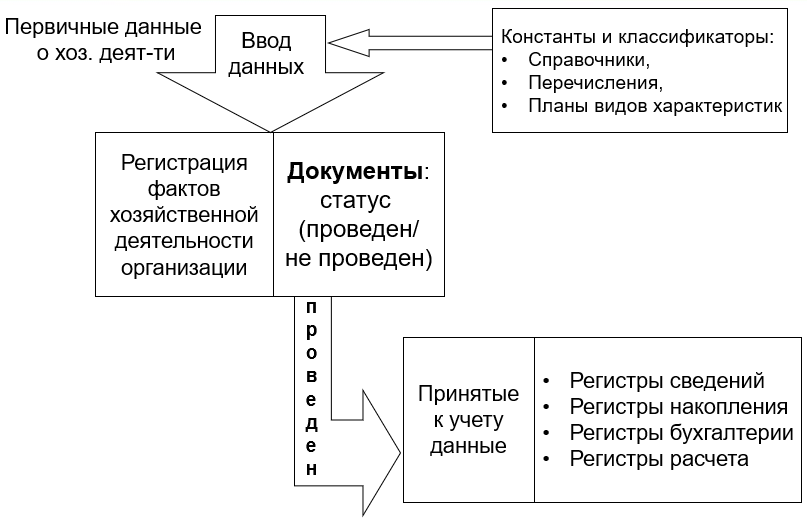
1. В чем отличие управляемого приложения и обычного приложе- ния?
2. Что такое командный интерфейс?
3. Каково назначение подсистем в управляемом и обычном при- ложении?
4. Что такое Роли?
5. Для чего используется список пользователей и как происходит его конфигурирование?
6. Управляемый интерфейс Такси: стандартные панели и их настройка.
7. Что такое форма?
8. В чем отличие управляемых и обычных форм?
9. Каков принцип разработки управляемой формы?
10. Какие элементы управления можно разместить на управляемой форме?
11. Какие возможности предоставляет редактор управляемой фор- мы?
12. Классификация команд в системе 1С:Предприятие 8.
13. Что такое событие объекта системы 1С:Предприятия?
14. Объясните принцип работы управляемых форм.
15. Каким образом можно запустить внешнее приложение на ис- полнение из 1С:Предприятия?
16. Для чего предназначены объекты, располагающиеся на ветви конфигурации «Общие»?
17. Для чего предназначен механизм привязок в обычных формах?
18. Какие механизмы интеграции заложены в платформу 1С:Предприятие?

# Лабораторная работа №3.

**КОНФИГУРИРОВАНИЕ ПРИКЛАДНОГО РЕШЕНИЯ В СИСТЕМЕ 1С:ПРЕДПРИЯТИЕ В СООТВЕТСТВИИ С ПРИНЦИПОМ УЧЕТА «ОТ ДОКУМЕНТА»**

*Цель работы:* освоение основных приемов работы с документами и регистрами в системе 1С:Предприятие, изучение правил работы с об- щими модулями и макетами. Знакомство с объектной моделью доступа к данным системы 1С:Предприятие.

# Теоретические сведения

Концепцию системы учета в системе 1С:Предприятие можно изоб- разить в следующем виде:

**Рис. 45 – Принцип учета «от документа»**

Суть принципа учета «от документа»:

* деятельность организации разбивается на хозяйственные операции;
* под каждую операцию создается тип документа, фиксирующего пер- вичную информацию о хозяйственной операции;
* большинство реквизитов документов связывается с помощью ссы- лочных типов данных с классификаторами, т.е. справочниками и другими объектами, предназначенными для хранения нормативно- справочной информации списочного характера;
* вводится различие между *фактом регистрации* хозяйственной опера- ции через документ и *фактом принятия данных к учету*. Принятые к

учету данные записываются в учетные объекты (регистры) и влияют на финансовое состояние организации.

На основе данных регистров, т.е. на основе данных, принятых к учету, строятся отчеты. Технически отчеты можно строить и по дан- ным документов, но с точки зрения учета это не является корректным действием. Данные документов некоторого типа могут использоваться для построения отчетов только в том случае, если в конфигурации не предусмотрено проведение этого типа документов по регистрам.

## Документы в системе 1С:Предприятие

Документ – одно из ключевых понятий системы 1С:Предприятие (как и любой системы учета). При помощи документов организуется ввод в систему информации о совершаемых хозяйственных операциях. Как правило, документы, которые создаются в процессе конфигу- рования, являются электронными аналогами стандартных бумажных

документов предприятия некоторого профиля.

Документ, как и большинство объектов конфигурации, характери- зуется набором стандартных реквизитов (Ссылка, Номер, Дата, Прове- ден, ПометкаУдаления), который можно расширить за счет добавления новых реквизитов и табличных частей.

Ссылка. Служебное поле в виде уникального (в пределах всей ИБ) набора символов. Ссылка присваивается при первичной записи объек- та (документа) и однозначно его идентифицирует.

Номер, Дата. Каждый документ характеризуется номером, датой и временем создания. Таким образом, все документы привязываются к оси времени и выстраиваются в хронологическом порядке, что необ- ходимо для правильного отражения хозяйственных операций. Номер и Дата присваиваются документу в момент записи.

Документ в системе 1С:Предприятие может специальным способом (этот способ называется проведение) влиять на другие объекты (на регистры). Состояние документа (проведен/не проведен) фиксируется через стандартный реквизит Проведен (тип Булево).

Механизм проведения отличает документы от остальных объектов конфигурации. В системе 1С:Предприятие принято следующее прави- ло: если документ проведен, то он повлиял на состояние данных (со- стояние учета) в системе, иначе, когда документ только записан, он считается «черновиком». Поэтому запись документа с незаполненны- ми обязательными реквизитами возможна, проведение – запрещено.

При проведении документ обычно отражает зафиксированное им событие (хозяйственную операцию) в регистрах, т.е. формирует дви- жения.

Движения документа – это записи в регистрах, которые создаются в процессе проведения документа и отражают изменения, производимые документом. Таким образом, документ является центральным элемен- том 1С:Предприятия: с одной стороны, он заполняется на основании справочников и перечислений, с другой стороны, он выполняет прове- дение по регистрам, на основании чего формируется вся необходимая на предприятии отчетность.

Под термином Документ на этапе конфигурирования подразумева- ется не конкретный экземпляр документа определенного типа (Накладная №435 от 12.04.04), а шаблон объекта, содержащий опреде- ленный набор реквизитов и механизмов управления и проведения. В режиме 1С:Предприятие создается необходимое количество экземпля- ров данного объекта, имеющих общую предопределенную структуру, но различное содержание реквизитов.

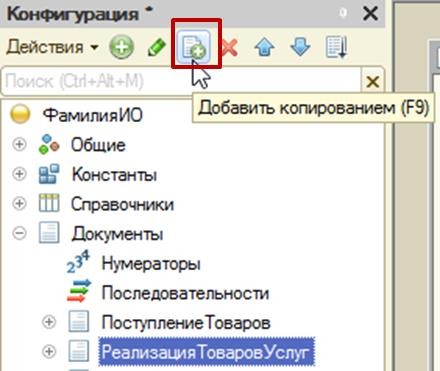
Все документы создаются и редактируются в ветке Документы дере- ва конфигурации. Помимо документов, в данной ветке также находят- ся служебные объекты – Нумераторы и Последовательности.

Для настройки алгоритма формирования Номера документа предна- значена вкладка Нумерация в мастере документа со следующими свой- ствами:

* флаг Автонумерация – при установке данного режима номер для ново- го документа будет устанавливаться автоматически путем увеличе- ния на «1».
* Нумератор – вспомогательный объект, позволяющий установить сквозную нумерацию для документов нескольких видов (например, для кадровых документов). Нумераторы предварительно создаются в ветке Документы > Нумераторы дерева конфигурации. Все документы, которым будет назначен конкретный нумератор, будут нумероваться по единым правилам, которые в нем прописаны.
* Длина номера – количество разрядов (знакомест) в номере. Длина номера в версии 8 увеличена до 50.
* флаг Контроль уникальности аналогичен справочникам – при попытке записать документ с номером, который уже есть в системе, платфор- ма выдаст ошибку, препятствующую записи в базу.
* Периодичность – свойство устанавливает пределы контроля уникаль- ности номеров документов и период повторяемости номеров. Если установлен Контроль уникальности, то в свойстве Периодичность ука- зывается, в каких пределах будет осуществляться этот контроль. При установленном флаге Автонумерация система 1С:Предприятие будет присваивать очередной порядковый номер каждому новому доку-

менту. После завершения периода, установленного в свойстве Перио- дичность, нумерация документов начнется заново.

## Механизмы создания документов

Создание новых типов документов в конфигураторе не всегда удобно выполнять с нуля. Если новый тип документа схож по составу реквизитов, табличных частей и поведению с одним из существующих в системе объектов ветви Документы, целесообразно воспользоваться механизмом копирования.

**Рис. 46 – Механизм добавления объекта копированием (в конфигураторе)**

Механизм создания копированием применим как для объектов од- ного типа, так и разного. В худшем случае копируется только наиме- нование и к нему добавляется префикс. Если объекты одного типа (ре- комендуемый вариант), то в объект-копию копируются: реквизиты и табличные части; формы и модули форм с процедурами- обработчиками событий; содержимое модуля объекта и модуля мене- джера.

**Важно!** Принадлежность подсистемам и права доступа при добавле- нии новых объектов копированием существующих игнорируется.

Механизм добавления копированием применим как на уровне объ- екта конфигурации, так и на уровне подчиненных объектов (форм, табличных частей, макетов и др.).

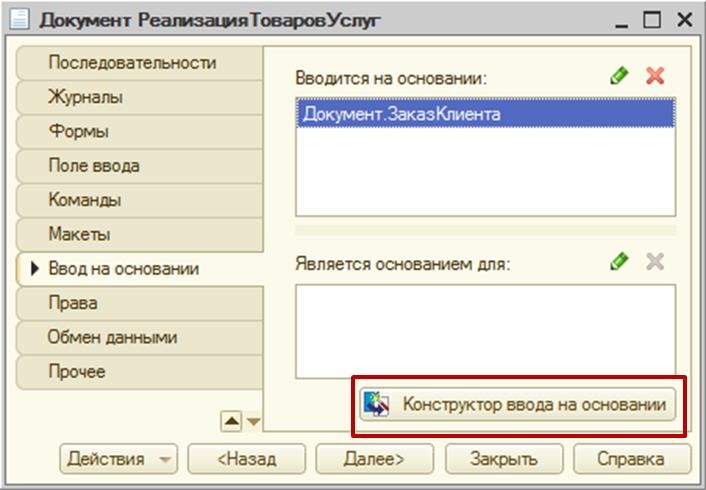
Добавление копированием работает и в режиме 1С:Предприятие. На уровне платформы заложен и другой механизм – Ввод на основа-

нии (предусмотрен для Справочников, Документов, Планов видов харак- теристик, Планов счетов и Планов видов расчета). Ввод на основании настраивается на одноименной вкладке мастера документа (справоч- ника или др.) в конфигураторе. После в режиме 1С:Предприятие этот механизм позволяет создать экземпляр документа (элемент справоч-

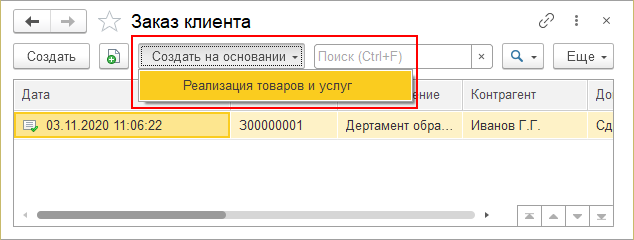
ника) одного типа с уже заполненными реквизитами (или частью рек- визитов); значения заполнения берутся из экземпляра документа дру- гого типа, который является основанием для создания нового.

Примеры использования ввода на основании: создание нового до- кумента типа Реализация товаров и услуг на основании документа Заказ клиента 000000001 от 13.10.2020; создание нового документа типа Выплата зарплаты на основании документа Начисление зарплаты 000000001 от 30.10.2020.

Для настройки ввода на основании в Конфигураторе предусмотрен

конструктор ввода на основании.

**Рис. 47 – Настройка ввода на основании в Конфигураторе**

В конструкторе нужно указать соответствие между реквизитами объекта-основания и объекта, который вводится на основании. Кон- структор сформирует типовой программный код в модуле объекта (вкладка Прочее > Модуль объекта) в виде процедуры ОбработкаЗапол- нения; в интерфейсе автоматически появится соответствующая коман- да (см. рисунок ниже).

**Рис. 48 – Кнопка «Создать на основании»**

**в командной панели формы документа-основания**

Конструктор ввода на основании – один из множества конструкто- ров в системе 1С:Предприятие, который позволяет визуальными сред- ствами расширить функциональные возможности прикладного реше- ния.

## Вспомогательные для документов объекты конфигурации: последовательности

Последовательности служат для контроля логической целостности хронологических цепочек документов, для поддержания актуальности итогов. Например, при внесении изменений задним числом все движе- ния последующих документов могут стать некорректными. Чтобы вос- становить корректность, нужно перепровести все нужные документы с момента внесения изменений и внести в них при необходимости опре- деленные изменения. Для решения таких задач и предназначены по- следовательности.

Системе необходимо знать, с какой даты (с какого документа) нуж- но восстановить последовательность, когда были внесены изменения

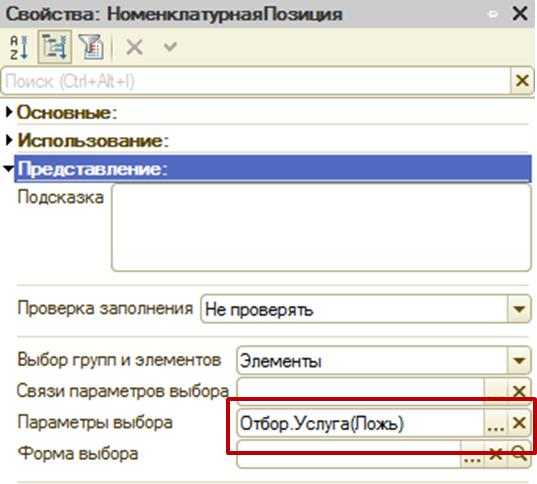
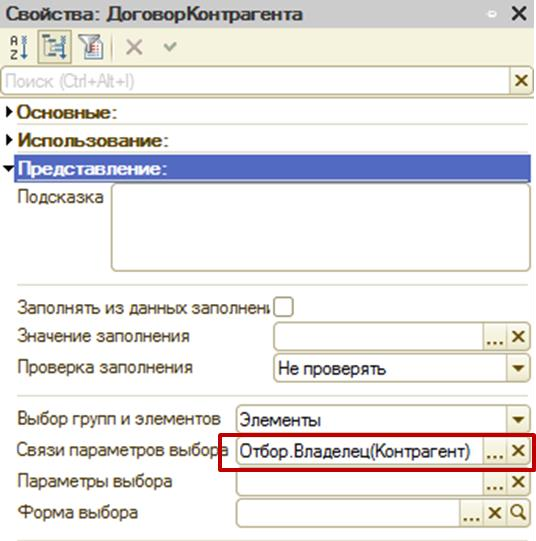
«задним числом». Для этого введено понятие Граница последовательно- сти – это маркер, устанавливаемый на позицию последнего проведен- ного документа. Если какие-то движения изменяются задним числом (при перепроведении документа с новым набором данных, удалении документа или отмене проведения), то Граница последовательности устанавливается на этот документ (или на предыдущий в случае уда- ления документа). Для каждой введенной в конфигурации последова- тельности документов система будет автоматически поддерживать границу последовательности. Получаем возможность видеть, с какой позиции, с какой даты у нас могут быть неактуальные итоги, неверные данные. Все документы последовательности, проведенные на основе ставших неактуальными итогов, будут находиться за Границей после- довательности.

Последовательности могут иметь подчиненные объекты, называе- мые измерениями, задаваемые на вкладке Данные. Если для последова- тельности не создано ни одного измерения, то при восстановлении данной последовательности будут перепроводиться все входящие в последовательность документы. Если требуется, чтобы данная после- довательность учитывала не все, а вполне определенные ситуации, то в последовательность включают измерение. В этом случае перепрово- дить нужно будет только те документы, которые изменяют состояние регистра с учетом свойств измерения. Восстановление последователь- ностей производится штатным средством Проведение документов в ре- жиме 1С:Предприятие.

## Примеры настройки формы документа

При заполнении реквизитов ссылочного типа документов (или дру- гих объектов) часто требуется организовать вызов форм выбора с установленным отбором. Рассмотрим две такие ситуации.

**Ситуация 1**. На форме документа размещены реквизиты Контрагент (тип СправочникСсылка.Контрагенты) и ДоговорКонтрагента (тип Справоч- никСсылка.Договоры); справочник Контрагенты является владельцем справочника Договоры. Пользователь при заполнении документа вы- бирает контрагента; далее, при выборе договора контрагента, пользо- вателю должна быть показана форма выбора, содержащая договоры, принадлежащие указанному на форме контрагенту, а не все содержи- мое справочника Договоры. Другими словами, необходимо открыть форму выбора элемента справочника Договоры с установленным в ней отбором по владельцу. Самый простой способ – использование свой- ства Связи параметров выбора реквизита ДоговорКонтрагента документа. **Ситуация 2**. На форме документа размещена табличная часть с реквизитом НоменклатурнаяПозиция типа СправочникСсылка.Номенк- латура. При этом для каждого элемента справочника Номенклатура установлен/сброшен признак Услуга (реквизит типа Булево). При выбо- ре номенклатурной позиции пользователю нужно предлагать только элементы, для которых Услуга = Ложь. В данном случае разумно ис- пользовать свойство Параметры выбора реквизита НоменклатурнаяПози-

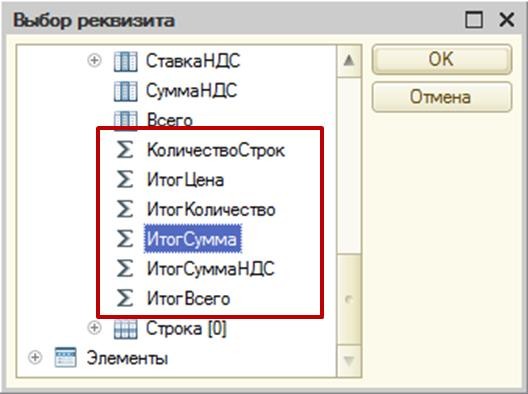
ция табличной части документа.

**Рис. 49 – Связи параметров выбора (слева) и Параметры выбора (справа)**

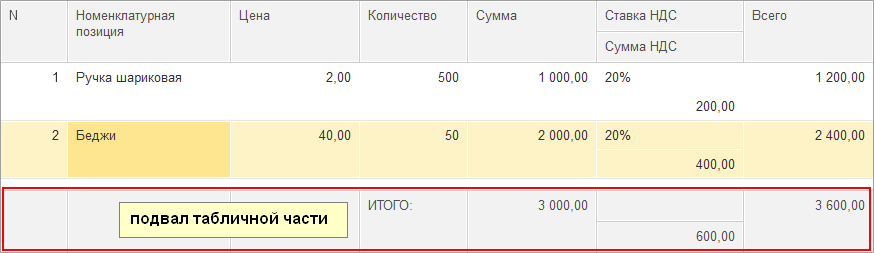
Документ, как и справочник, может иметь произвольное количе- ство табличных частей. Несколько табличных частей требуются в тех случаях, когда одним документом необходимо зарегистрировать раз-

ные по сути, но связанные события. Например, в документе Поступле- ние товаров в конфигурации Управление торговлей (УТ) есть 3 таблич- ные части: Товары, Услуги и Тара.

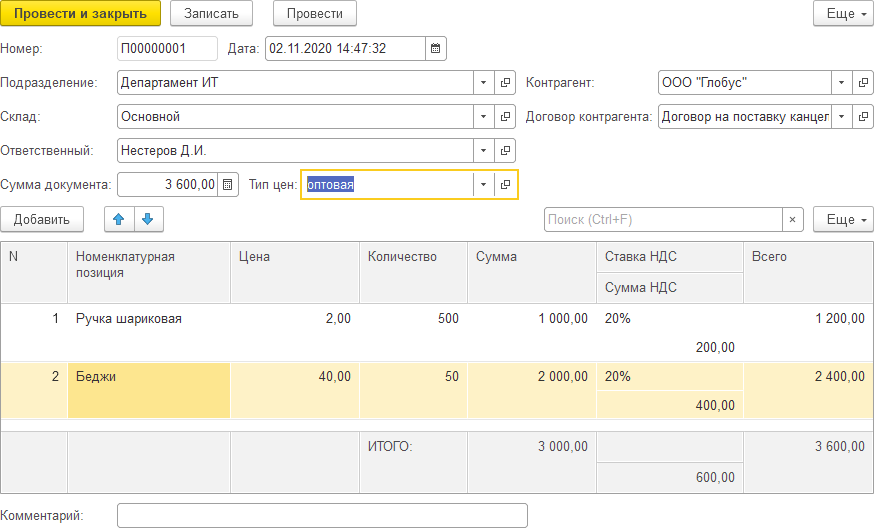
Любая табличная часть состоит из нескольких областей: шапка, строки табличной части и подвал. По умолчанию отображение подвала отключено. Для управления отображением подвала предназначен флаг Подвал в свойствах элемента управления формы, связанного с таблич- ной частью объекта.

В подвале удобно размещать итоговые значения по числовым ко- лонкам табличной части. Для этого нужно выбрать поле ввода на фор- ме, соответствующее числовому реквизиту табличной части. Одно из его свойств – ПутьКДаннымПодвала – позволяет указать данные, с кото- рыми связан подвал. Для выбора предлагается несколько агрегатных функций:

**Рис. 50 – Заполнение свойства ПутьКДаннымПодвала**

Таким образом, путем настройки свойств подвала табличной части можно отобразить итоговые значения ее числовых колонок без при- влечения средств программирования.

**Рис. 51 – Форма документа с табличной частью, в подвале которой отображены итоговые значения по колонкам Сумма, СуммаНДС и Всего**



**Рис. 52 – Форма документа ПоступлениеТоваров с табличной частью**

Программный доступ из формы документа (или элемента справоч- ника, или др. объектов) к данным активной строки табличной части имеет особенности: обращение к реквизитам документа выполняется через основной реквизит формы, *обращение к реквизитам табличной части – через элементы управления*:

&НаКлиенте

Процедура НоменклатурнаяПозицияПриИзменении(Элемент)

//обращаемся к активной строке табл. части Номенклатура ТекущиеДанные =

ЭтаФорма.Элементы.Номенклатура.ТекущиеДанные;

//обращаемся к реквизиту Цена табличной части ТекущиеДанные.Цена = 0;

КонецПроцедуры

## Журналы документов

Журнал документов – объект конфигурации, предназначенный для группировки связанных по смыслу документов разного вида. Журналы документов позволяют просматривать в одном списке документы раз- ных видов. Фактически журналы не хранят данных, а содержат набор ссылок на группируемые документы.

Свойство Регистрируемые документы определяет, документы каких типов будут отображаться в журнале. Каждый тип документа может быть показан в нескольких журналах.

Для журналов документов предопределен набор стандартных рек- визитов (Ссылка, Дата, Номер, Тип, Проведен, ПометкаУдаления). Рекви- зит Тип содержит тип регистрируемого в журнале документа.

Набор стандартных реквизитов можно расширить путем создания граф. При создании графы из каждого регистрируемого документа нужно выбрать по реквизиту (только по одному, но можно разных ти- пов), который будет в ней отображаться.

## Многообразие программных модулей: модуль объекта и модуль менеджера, общие модули

С понятием программного модуля связано понятие контекст модуля. Владелец модуля объекта – объект информационной базы (модуль до- ступен в мастере объекта конфигурации на вкладке Прочее); владелец модуля менеджера – объект конфигурации (модуль также доступен на вкладе Прочее); владелец общих модулей – прикладное решение (мо- дули располагаются на ветви Общие дерева конфигурации). Таким об- разом, каждый программный модуль связан с какой-то частью конфи- гурации. Такая привязка и называется контекстом модуля. Контекст мо- дуля определяет набор объектов, свойств и методов (процедур и функ- ций), доступных для написания и исполнения алгоритма.

Частично данный контекст можно увидеть при редактировании мо- дуля при помощи контекстного меню: через Ctrl + пробел, через точку от указанного объекта, через знак =.

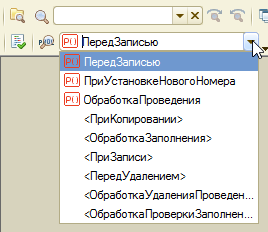
Таким образом, контекст определяет возможность исполнения той или иной строки кода в определенном модуле или его части.

Модуль объекта. На этапе исполнения модуль объекта существует в контексте конкретного экземпляра объекта БД (например, объекта элемента справочника Номенклатура или документа Поступление това- ров №000000001). Из модуля имеется доступ к реквизитам и табличным частям объекта, а также его методам и событиям.

**Важно!** Поскольку владельцем модуля объекта является объект ин- формационной базы (который не может существовать на стороне кли- ента), поэтому контекст (набор свойств, методов) модуля объекта не доступен на клиенте.

Если тип прикладного объекта конфигурации не предполагает мо- дификацию хранимых в БД данных на пользовательском уровне (пере- числения), либо не связан с хранением данных (журналы документов), то для таких объектов не будет модулей объектов.

У объекта, как и у формы, есть набор предопределенных событий, процедуры-обработчики этих событий располагаются в модуле объек- та (ПриЗаписи, ПриКопировании, ПередУдалением, и др.). Имена этих

процедур предопределены на уровне платформы и не подлежат изме- нению. Заполняя тело процедуры-обработчика предопределенного события своим кодом, мы фактически задаем правила поведения си- стемы в отношении объектов ИБ при записи, при копировании и про- чих операциях. Перечень предопределенных процедур-обработчиков доступен в выпадающем списке Процедуры и функции [Ctrl+Alt+P], кото- рый отображается в панели инструментов конфигуратора при актив- ном окне модуля объекта.

**Рис. 53 – Процедуры-обработчики предопределенных событий объекта**

Кроме процедур-обработчиков предопределенных событий кон- текст модуля объекта может включать пользовательские процедуры и функции, которыми разработчик расширяет функциональность объек- та.

В модуле объекта контекст доступен через конструкцию ЭтотОбъ- ект. Конструкция не является обязательной, зато поможет ориентиро- ваться в контексте модуля.

Следующий листинг содержит пример заполнения реквизита доку- мента СуммаДокумента при событии объекта ПередЗаписью. Для запол- нения используется стандартный метод табличной части Итог с пара- метром – имя реквизита табличной части.

Процедура ПередЗаписью(Отказ)

// в реквизит документа СуммаДокумента записывается

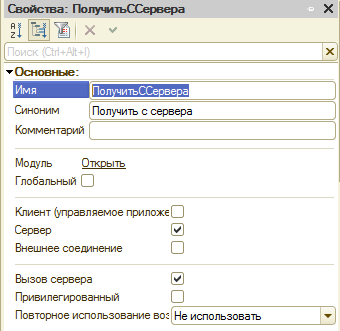
// итог по колонке Всего табличной части Номенклатура ЭтотОбъект.СуммаДокумента =

ЭтотОбъект.Номенклатура.Итог("Всего");

КонецПроцедуры

Модуль менеджера существует в контексте набора объектов некого типа (например, документов типа ПоступлениеТоваров). С помощью модулей менеджера можно расширить набор методов по работе с ти- пами объектов ИБ, т.е. с множеством значений. Поэтому наличие мо-

дуля менеджера зависит от того, позволяет ли указанный тип объекта конфигурации работать с наборами значений. Например, константа может содержать только одно значение, поэтому для данного объекта нет модуля менеджера.

Общие модули предназначены для размещения в них текстов функ- ций и процедур, которые могут вызываться из любого другого модуля конфигурации (требуется использовать ключевое слово Экспорт при описании таких подпрограмм).

**Рис. 54 – Пример настроек общего модуля**

Свойство Глобальный общего модуля. Если это свойство установле- но в значение Истина, то экспортируемые процедуры и функции обще- го модуля являются частью глобального контекста и их имена в раз- личных глобальных общих модулях должны быть уникальными. При вызове таких процедур и функций из других модулей нет необходимо- сти использовать имя глобального модуля.

Свойства Клиент, Сервер, Внешнее соединение определяют место ис- полнения процедур и функций общего модуля.

Если свойство Вызов сервера установлено в значение Истина, то экспортные процедуры и функции модуля могут быть использованы на клиенте.

В следующем листинге приведена фунция ПолучитьЗначениеНДС, определенная в общем модуле ПолучитьССервера (его настройки на рис. 54). Функция используется для получения численного значения ставки НДС, в качестве формального параметра – ссылка на одно из значений перечисления СтавкаНДС. Также в программном коде обозна- чен способ обращения к значениям перечисления.

//функция определена в общем модуле

Функция ПолучитьЗначениеНДС(Ставка) Экспорт Если Ставка = Перечисления.СтавкиНДС.НДС20

Тогда Возврат 0.20;

ИначеЕсли Ставка = Перечисления.СтавкиНДС.НДС10 Тогда Возврат 0.10;

// обрабатываем аналогично остальные не нулевые ставки Иначе

Возврат 0; КонецЕсли; КонецФункции

Вызов описанной функции возможен из других общих модулей и, например, из модулей форм объектов (документов, справочников). Поскольку функция определена в неглобальном общем модуле, при ее вызове нужно использовать имя самого модуля.

## Макеты и создание печатных форм

Объект конфигурации Макет предназначен для хранения различных форм представления данных, которые могут потребоваться каким-либо объектам конфигурации или всему прикладному решению в целом. Макеты бывают общие (доступны из любого модуля) и подчиненные прикладным объектам (макеты справочников, документов, журналов документов, отчетов, обработок, регистров и др.).

В системе 1С:Предприятие предусмотрено 10 типов макетов: таб- личный документ, текстовый документ, двоичные данные, HTML- документ, Active Document и др. Одно из предназначений макета, под- чиненного объекту конфигурации и содержащего табличный документ

– создание печатной формы этого объекта. Создание печатной формы заключается в конструировании ее составных частей – именованных областей, из которых затем собирается готовая печатная форма. Поря- док заполнения областей данными и порядок вывода их в итоговую форму описывается при помощи встроенного языка.

Печатная форма может включать в себя различные графические объекты: картинки, OLE-объекты, диаграммы и т.д.

Макет типа ТабличныйДокумент состоит из ячеек, которые могут быть объединены в именованные области. Каждая ячейка имеет имя и может быть заполнена текстом, через параметр или по шаблону (спо- соб заполнения ячейки указывается через ее свойство Заполнение):

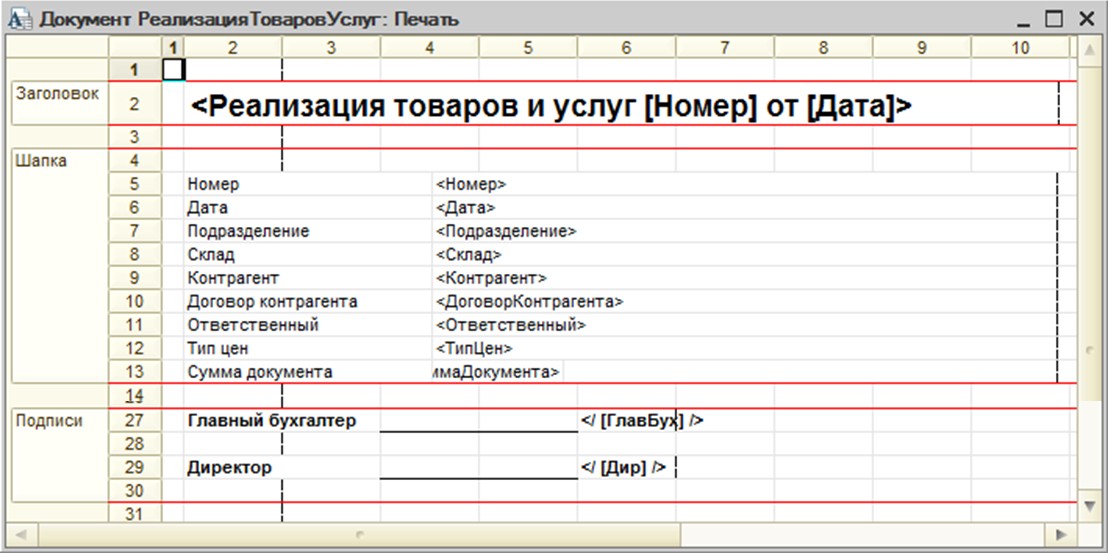
* Текст, содержащийся в ячейке, будет показан на экране;
* Параметр будет заменен некоторым значением, которое может быть присвоено ему средствами встроенного языка. Текст, содержащийся в ячейке, является именем этого параметра;
* Шаблон представляет собой текстовую строку, в определенные места которой будут вставлены значения параметров.

В макете на рис. 55 присутствуют ячейки с разными вариантами

Заполнения:

* ячейка R2C2 (2-я сторока, 2-й столбец): шаблон с параметрами Номер

и Дата;

* ячейка R5C2: текст;
* объединенная ячейка R5C3:R5C4: параметр Номер;
* ячейка R27C5: шаблон с параметром ГлавБух.

**Рис. 55 – Макет типа ТабличныйДокумент**

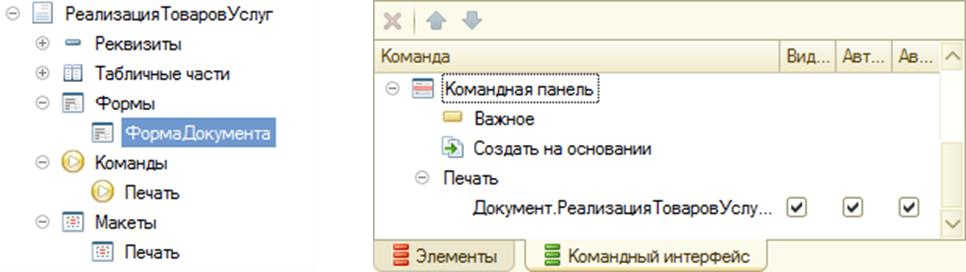
**с именованными областями Заголовок, Шапка, Подписи**

Особенностью макета типа ТабличныйДокумент является то, что возможно задать разный формат (ширину) колонок для разных строк.

В системе 1С:Предприятие возможно создание макетов как вруч- ную, так и с помощью конструктора печати, который доступен на вклад- ке Макеты мастера объекта либо из его контекстного меню.

На первом шаге в конструкторе печати задается имя процедуры, в которую будет помещен программный код по заполнению печатной формы. На втором, третьем и последующих шагах выбираются рекви- зиты объекта, которые нужно печать в стандартных областях макета: шапке, табличных частях и подвале. На последнем шаге указывается группа команд, которая обозначает, в какое место в командном интер- фейсе будет помещена команда формирования печатной формы.

В результате работы конструктора печати происходят следующие изменения в конфигурации:

* создается макет объекта (документа, элемента справочника или др.);
* создается команда Печать и связанная с ней кнопка в командном ин- терфейсе;
* создается процедура Печать (с программным кодом по заполнению макета данными объекта) в модуле менеджера объекта.

**Рис. 56 – Подчиненные документу команда и макет (слева) и элемент командного интерфейса (справа), сформированные автоматические конструктором печати**

В следующем листинге приводится типовой код, генерируемый конструктором печати, а также некоторые пояснения.

//процедуры ОбработкаКоманды и Печать определены в модуле

//команды Печать документа РеализацияТоваровУслуг &НаКлиенте

Процедура ОбработкаКоманды(ПараметрКоманды,

ПараметрыВыполненияКоманды)

//{{\_КОНСТРУКТОР\_ПЕЧАТИ(Печать)

//в оперативной памяти создается объект ТабДок ТабДок = Новый ТабличныйДокумент; Печать(ТабДок, ПараметрКоманды);

ТабДок.ОтображатьСетку = Ложь; ТабДок.Защита = Ложь; ТабДок.ТолькоПросмотр = Ложь; ТабДок.ОтображатьЗаголовки = Ложь;

//Заполненный данными ТабДок выводится пользователю ТабДок.Показать();

//}}

КонецПроцедуры

&НаСервере

Процедура Печать(ТабДок, ПараметрКоманды)

//вызывается метод Печать документа

//РеализацияТоваровУслуг;

//метод определен в модуле менеджера этого документа Документы.РеализацияТоваровУслуг.Печать(ТабДок,

ПараметрКоманды);

КонецПроцедуры

Основной программный код по заполнению областей и параметров макета Печать данными вынесен в модуль менеджера. Общий алго- ритм:

* подключается макет командой ПолучитьМакет;
* выполняется запрос на выборку из БД данных, которые следует по- местить в печатную форму;
* из макета выбирается именованная область;
* в параметры именованной области записываются значения;
* именованная область с заполненными параметрами включается в табличный документ.

Следующий листинг содержит пример заполнения параметра

ГлавБух области Подписи макета с рис. 55.

// получаем именованную область Подписи ОбластьПодписи = Макет.ПолучитьОбласть("Подписи");

// находим сотрудника с нужной должностью ГлавБух =

Справочники.Сотрудники.НайтиПоРеквизиту

("Должность",Справочники.Должности.ГлавБух);

// конструкция Справочники.Должности.ГлавБух отработает

// в случае, если этот элемент предопределенный СтруктураПодписи = Новый Структура; СтруктураПодписи.Вставить("ГлавБух",ГлавБух.Наименование); ОбластьПодписи.Параметры.Заполнить(СтруктураПодписи); ТабДок.Вывести(ОбластьПодписи);

## Регистры как учетные объекты в системе 1С:Предприятие

В системе 1С:Предприятие определено 4 типа регистров: регистры сведений, регистры накопления, регистры расчета и регистры бухгалтерии. Формальное отличие между ними – набор стандартных методов и свойств.

Регистр сведений универсален, остальные типы регистров решают специфические задачи (регистр накопления – накопление остатков и оборотов по различным показателям, бухгалтерии – организация бух- галтерского учета, расчета – выполнение периодических и не периоди- ческих расчетов: зарплата и т.п.).

Информация в регистрах хранится в виде записей. Регистры, в от- личие от справочников/перечислений/документов, не создают новый тип (не являются типообразующими объектами). Поэтому нельзя заве- сти ссылку на запись регистра.

**Напоминание**. При добавлении нового справочника (документа, пере- числения) в системе появляется новый тип данных СправочникСсыл- ка.<Имя справочника> (ДокументСсылка.<Имя документа>, Перечисле- ниеСсылка.<Имя перечисления>). В случае с регистрами такого не про- исходит.

На физическом уровне регистр – таблица, где колонки могут быть нескольких типов: измерения, ресурсы, реквизиты. Измерение – еди-

ница детализации хранимой информации (в каких разрезах хранятся данные), ресурс – единица хранения информации, реквизит – некая сопроводительная информация для записей регистра.

Получаем, что регистр представляет собой многомерный массив данных: индексы массива – измерения, элементы массива – ресурсы. Доступ к ресурсам может осуществляться как по полному набору из- мерений, так и по их подмножеству, и тогда результатом будет не одна запись регистра, а множество:

w: (x, y, z)  wxyz

W: (x,y)  W = {wz1, wz2 … wzn}

**Важно!** В отличие от справочников и документов информация в реги- стре находится не по ссылке (ссылок на записи регистра не существу- ет, нет стандартного реквизита Ссылка), а по адресу, т.е. по набору из- мерений.

Основная цель использования регистров – это накопление инфор- мации о хозяйственных операциях, которая вводится с использованием документов или при помощи расчетов. В дальнейшем эта информация извлекается и анализируется с предоставлением пользователю отчет- ных форм. C одной стороны, сохранение части данных документов в регистрах (при проведении документов) приводит к хранению избы- точной информации. Но с другой стороны, только регистры поставля- ют методы, позволяющие быстро анализировать данные (а именно: получать актуальные значения реквизитов, вычислять остатки, оборо- ты и итоги по некоторым показателям), которые аккумулируются в регистрах. Этим и обосновывается наличие регистров в системе 1С:Предприятие.

## Регистры сведений и регистры накопления

Основная задача регистра сведений – хранить существенную для прикладной задачи информацию, состав которой развернут по опреде- ленной комбинации значений и при необходимости развернут во вре- мени. Примеры использования регистра сведений: Курсы валют (хра- нение курсов в разрезе валют), Цены номенклатуры (хранение цен в разрезе товаров, поставщиков, типов цен). *Как правило, в один регистр сведений объединяются ресурсы, значения которых обычно изменяются одновременно.*

На тип ресурсов и измерений не накладывается ограничений, но рекомендуется для измерений использовать ссылочные типы.

Регистры сведений позволяют хранить историю изменения рекви- зитов и констант и получать их на заданную точку на временной оси. Важно отметить, что, если не нужно разворачивать информацию по

какому-либо набору измерений, кроме времени, то вообще не нужно создавать измерения. Штатный механизм для хранения истории изме- нения информации – свойство Периодичность регистра сведений. Если выбран какой-то вариант периодичности (в пределах секунды, дня, месяца, квартала, года), то время добавляется как дополнительное из- мерение (активируется стандартный реквизит Период). Информация в регистре не может изменяться чаще, чем указано в свойстве Периодич- ность. Если выбирается значение Непериодический, то вновь введенное значение ресурса с конкретным набором измерений затрет старое. Ес- ли же задана периодичность и период, за который было сделано изме- нение, не меньше установленного, то старое значение автоматически сохранится. В этом случае при добавлении записи в регистр происхо- дит округление даты до начала периода.

Свойство Режим записи регистра сведений определяет способ изме- нения его записей:

* Независимый режим, при котором изменения в регистр вносятся вручную;
* Режим Подчинение регистратору, при котором изменения в регистр вносит документ (т.е. регистратор). Такой способ воздействия на ре- гистр называется проведением, в результате которого по нашему за- данному алгоритму данные из реквизитов документа дублируются в регистре. Если регистр подчинен регистратору, то в командном ин- терфейсе видимость регистра по умолчанию снята (даже если ре- гистр включен в подсистемы), т.к. в этом случае регистр считается служебным.

В режиме записи Подчинение регистратору запись регистра жестко подчинена документу-регистратору (и может быть удалена при удале- нии документа). Кроме того, появляется дополнительный вариант пе- риодичности – По позиции регистратора, при котором хронологически запись регистра будет характеризоваться парой Дата+Ссылка на доку- мент-регистратор. Это обеспечивает максимальную детализацию, с точностью больше секунд.

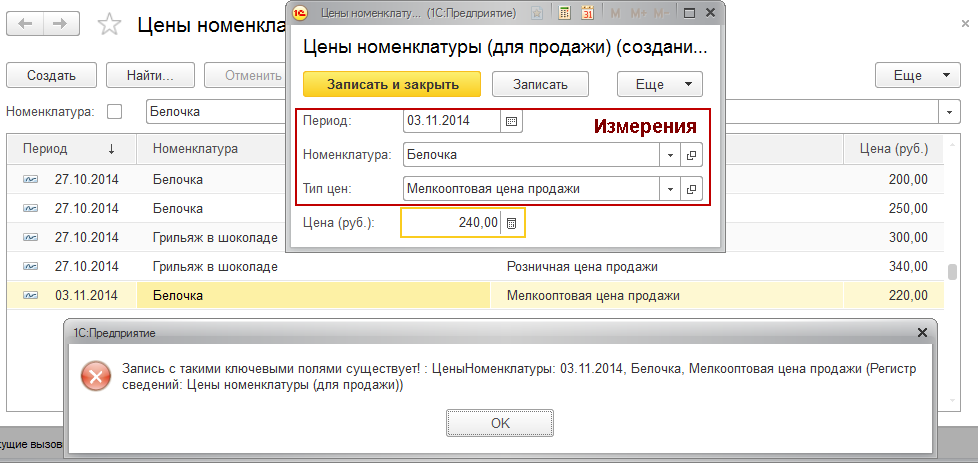
В режиме записи Подчинение регистратору задействуются стандарт- ные реквизиты Регистратор (ссылка на документ-регистратор), Номер строки (с одним документом может быть связано несколько записей регистра).

При независимом режиме записи регистра подчиняются так назы- ваемым ведущим измерениям.

Свойство измерения Ведущее устанавливает явную связь между объектом и ссылками на него в записях регистра сведений. Благодаря

этой связи при удалении объекта могут быть удалены записи из реги- стра сведений, ссылающиеся на удаляемый объект.

Свойство Основной отбор устанавливает измерение как одно из ос- новных для оптимизации запросов. При включении свойства Запрет незаполненных значений нельзя будет сохранить значение в регистр с незаполненным значением данного измерения. Свойство Индексировать доступно для редактирования, если измерение не является ведущим. Для таких измерений создается отдельный индекс, что увеличивает производительность при работе с регистром. Но чрезмерное индекси- рование может замедлить работу системы.

В регистре сведений не может существовать две записи с одинако- вым набором значений измерений и периодом (для периодических регистров). Это принципиально отличает регистр сведений от регистра накоплений, в котором такая ситуация является вполне нормальной (например, два прихода одного товара на один и тот же склад). Если мы запишем какое-либо значение в ресурс с набором измерений, кото- рый уже был введен в регистре, то возникнет ошибка (в общем слу- чае):

**Рис. 57 – Ошибка, вызванная нарушением уникальности запси в регистре сведений**

Все регистры имеют нечто общее: записи регистров не имеют объ- ектной природы (на записи регистров нельзя завести ссылки); структу- ра регистров определяется набором измерений, ресурсов, реквизитов; для каждого типа регистров на уровне платформы определены «спе- цифические умения».

Говоря о регистрах сведений, среди таких «специфических уме- ний» выделяют возможность быстро получать срезы первых (послед- них) значений ресурсов.

Регистры накопления используются для накопления информации о наличии и движении средств (товарных, денежных и других) и позво- ляют быстро получать информацию об остатках, оборотах, остатках с оборотами, итогах на интересующие даты, за указанные периоды. Т.е. регистр накопления оптимизирован для решения задач получения остатков и оборотов показателей – то «специфическое» умение (мето- ды объекта), за которое регистр накопления выделен в отдельный тип объектов.

Регистр накопления не может существовать без ресурсов. Тип ре- сурсов - Число (т.к. при получении остатков, оборотов, итогов регистра имеют место арифметические операции).

Изменение состояния регистров накопления выполняется при про- ведении документа. Процедура проведения документа содержится в модуле документа. Такая процедура содержит алгоритм формирования сведений об изменениях в регистрах, которые необходимо выполнить при проведении документа. Эти сведения называются движениями регистра и представляют собой записи в регистрах, которые создаются в процессе проведения документа и отражают изменения, производи- мые документом.

Движения регистров содержат только приращения (со знаком «+» или «-») значений ресурсов регистра, а не итоговые величины значе- ний.

Изменения в регистрах вносятся движениями регистра. Движения регистров оказывают влияние на его итоги. Итоги – это сводная ин- формация регистров накопления, которая получается путем суммиро- вания значений, вносимых движениями регистров.

Специалист, выполняющий конфигурирование системы, имеет возможность предоставить конечному пользователю средства про- смотра движений регистров. Конфигуратор позволяет создавать экранные и печатные формы для просмотра и анализа движений реги- стров.

В отличие от движений регистра, нет возможности непосредствен- но просматривать итоги регистра накопления. Для обращения к итогам в конфигурации может быть создано необходимое число отчетов, ко- торые будут обращаться к итогам и выдавать их в виде товарных отче- тов, складских карточек и ведомостей и т.д.

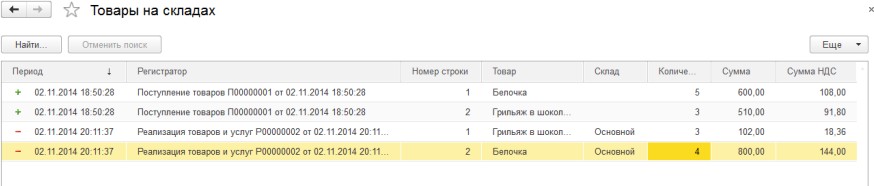
В зависимости от того, показатели какого вида должны храниться в регистре, определяется вид регистра (регистр накопления с видом остатки или регистр накопления с видом обороты).

Показатели вида Остатки: остаток товара на складе, деньги в кассе или на расчетном счете, взаиморасчеты с контрагентами (увеличе-

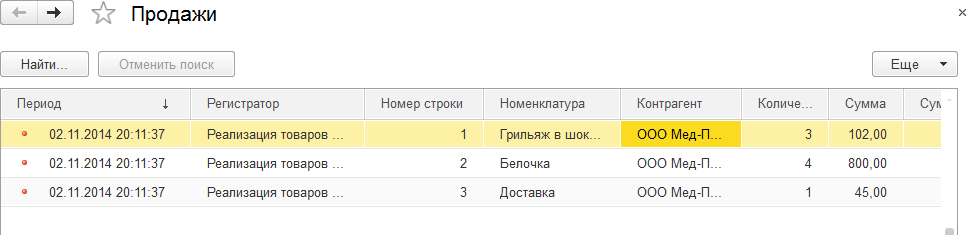
ние/уменьшение долга). Значение такого показателя может увеличи- ваться и уменьшаться. Для регистра накопления вида Остатки преду- смотрен стандартный реквизит ВидДвижения = Приход/Расход, кото- рый хранит знак приращения значений ресурсов.

Показатели вида Обороты: как правило, изменяется только в одну сторону (увеличивается). Примеры: количество проданного товара, прибыль (стоит на месте или увеличивается).

Для регистра остатков методы встроенного языка позволяют полу- чить остатки регистра накопления на заданный момент времени. Есть возможность фильтрации по значениям измерений, а также получения остатков в разрезе других измерений.

Физически регистр накопления представляется в виде двух таблиц БД – таблица движений (хранит исходные данные, т.е. приращения) и таблица итогов (формируется в БД системой автоматически). Итоги в регистре накопления хранятся с периодичностью месяц. При получе- нии итогов на момент, отличный от начала месяца, система будет ана- лизировать движения от ближайшего рассчитанного месяца.

**Рис. 58 – Регистр накопления ТоварыНаСкладах (регистр вида Остатки)**



**Рис. 59 – Регистр накопления Продажи (регистр вида Обороты)**

Регистры используются для хранения информации, к которой тре- буется получать оперативный доступ. Также регистры позволяют по- лучать наиболее достоверную информацию о состоянии средств. Так как процессы сохранения документа и записи изменений в регистрах разделены (допускается сохранение документа без его проведения), может возникать расхождение между данными документов и инфор- мацией в регистрах. Но регистр, в отличие от документа, является хра- нилищем итоговой информации, поэтому именно запись изменений в регистры служит подтверждением того, что хозяйственная операция совершена.

Проведение документа может осуществляться одновременно по не- скольким регистрам, если этого требует учетная политика предприя- тия. Для осуществления такой возможности, необходимо выбрать не- обходимые регистры, по которым будут осуществляться движения, на вкладке Движения требуемого документа. Затем можно воспользовать- ся конструктором движений для формирования алгоритма внесения из- менений в регистры (т.е. сформировать необходимые движения по выбранным регистрам).

## Проведение документов по регистрам

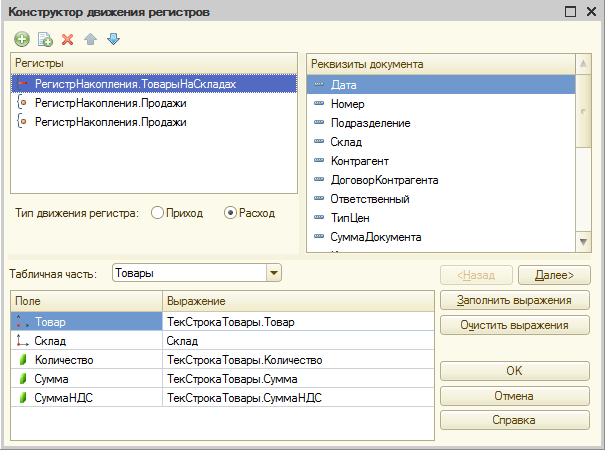
Проведение документа – это создание движений по различным ре- гистрам системы, то есть это и есть регистрация движений документа в системе. Обычно движения основаны на данных, введенных в рекви- зиты документа. Потом на основании этих движений строятся различ- ные отчеты, определяются остатки и обороты чего-либо. Конечно, от- четы можно сформировать на основании самих реквизитов докумен- тов, без формирования движений, но это неэффективный и нерекомен- дуемый подход.

Свойство Проведение на вкладке Движения мастера документа уста- навливает, разрешено ли проведение документа при записи. Если вы- брано Разрешить, то документ осуществляет движения регистров (из- меняет их состояние).

Свойство Оперативное проведение устанавливает, разрешено ли оперативное проведение документа при записи. Для документов с раз- решенным оперативным проведением при выборе не текущей даты, проведение осуществляется в неоперативном режиме, так как учиты- вается уже свершившийся факт, который не требует контроля, осу- ществляемого в оперативном режиме (например, проверка остатка указанного в расходной накладной товара).

Свойство Удаление движений позволяет выбрать между автоматиче- ским и неавтоматическим удалением движений регистров сведений, накопления, бухгалтерии и расчета. Движения могут удаляться при перепроведении, удалении и отмене проведения документа. Если про- ведение документа запрещено, то данное свойство недоступно. Стан- дартным считается режим автоматического удаления движений. Если его отменить, то ответственность за состояния движений ложится на плечи разработчиков - необходимо средствами встроенного языка кон- тролировать и удалять движения. При помощи отключения автоудале- ния можно, например, сделать так, чтобы при введении новой строки в табличную часть проведенного документа совершались движения только по этой строке, а по другим – игнорировались.

Ниже находится окно списка регистров. В нем нужно отметить только те регистры, по которым возможны движения данного доку- мента. Не стоит отмечать все регистры, если у вас их большое количе- ство – это может плохо сказаться на производительности системы в целом. Будьте аккуратны со снятием пометки регистра – после ре- структуризации базы движения по этому регистру будут безвозвратно уничтожены, установка пометки обратно их не вернет. Потребуется перепроводить все документы данного типа.

Если выбрать хотя бы один регистр, то станет активной кнопка Конструктор движений. Конструктор движений позволяет настроить в визуальном режиме алгоритм проведения документа по нужным реги- страм и автоматически сформировать типовую процедуру проведения.

**Рис. 60 – Конструктор движений для документа РеализацияТоваровУслуг**

Рассмотрим ситуацию, когда в конфигурации имеется документ Ре- ализацияТоваровУслуг с двумя табличными частями (Товары и Услуги). Документ отражает продажу товаров и оказание услуг клиентам. Эта хозяйственная операция влияет на остатки единиц товаров на складах, а также на сумму прибыли от продаж. Поэтому при проведении доку- мент изменяет состояние двух регистров накопления: ТоварыНаСкла- дах (вида Остатки) и Продажи (вида Обороты).

Основная задача при работе с конструктором движений – указать, по какому правилу (выражению) будут заполняться поля регистра. В выражении, как правило, участвуют реквизиты документа. Соответ-

ствие между полями (измерениями, ресурсами) регистра и реквизита- ми документа устанавливается в первую очередь по типу. Если есть совпадения по типам и именам, то срабатывает кнопка Заполнить выра- жения. Корректность выражения конструктором не проверяется.

При необходимости проводить документ по нескольким регистрам (как на рис. 60), в области Регистры по кнопке Добавить создается нуж- ное количество движений.

Маркеры  и  обозначают, что при проведении документа по соответствующему регистру в нем формируется столько записей, сколько строк присутствует в табличной части документа. Для обеспечения такого поведения в одноименном поле в конструкторе движений следует выбрать табличную часть документа.

При создании движений по регистру накопления также в конструкторе следует указать Тип движения регистра. Поскольку отражение в учете документа РеализацияТоваровУслуг связано с уменьшением единиц товаров на складе, то в данному случае тип движения – Расход.

После завершения работы конструктора в модуле объекта докумен- та формируется процедура ОбработкаПроведения. Процедура является транзакцией, т.е. если предполагается проведение документа по не- скольким регистрам, но в процессе выполнения произошел сбой, то данные не запишутся ни в один регистр (действие будет полностью отменено).

## Объектная техника доступа к данным регистров

Для работы с регистром сведений средствами встроенного языка применяются следующие объекты:

// РегистрСведенийМенеджер.<Имя регистра сведений>

// объект этого типа предназначен для управления данными кон- кретного регистра сведений. Позволяет осуществлять поиск, выби- рать и создавать записи регистра сведений

// РегистрСведенийМенеджерЗаписи.<Имя регистра сведений>

//объект этого типа позволяет читать, записывать и удалять

//отдельную запись регистра сведений ЦенаТовара =

РегистрыСведений.ЦеныНоменклатуры.СоздатьМенеджерЗаписи(); ЦенаТовара.Период = ТекущаяДата();

// заполняем остальные измерения и ресурсы в соответствии со

// структурой регистра ЦенаТовара.Записать(Истина);

// метод Записать(<Замещать>), параметр <Замещать> позволяет установить или отключить режим замещения существующей записи с тем же набором значений измерений регистра и с тем же периодом,

если Замещать = Истина, то происходит замещение, иначе – та же ошибка, что и при интерактивном добавлении одинаковых записей в регистр сведений

В первой строке листинга создается экземпляр объекта РегистрСведе- нийМенеджерЗаписи, который связывается с переменной ЦенаТовара. Далее заполняются ресурсы и измерения регистра. Запись будет произ- водиться для всех ресурсов. Если какому-то ресурсу не будет присвоено значение, то запишется пустое значение. При помощи метода Записать изменения заносятся в БД.

Для периодических регистров сведений предусмотрены механизмы извлечения наиболее ранней/поздней информации с использованием следующих методов встроенного языка:

ПолучитьПервое(<НачалоПериода>, <Отбор>)

// Метод получает значения ресурсов наиболее ранней записи ре- гистра, соответствующей указанным периоду и значениям измерений регистра.

ПолучитьПоследнее(<КонецПериода>, <Отбор>) СрезПервых(<НачалоПериода>, <Отбор>)

// Получает наиболее ранние записи регистра, соответствующие установленным в параметрах метода значениям ключевых полей. За- писи подбираются для каждого сочетания из всех имеющихся значе- ний измерений регистра.

СрезПоследних(<КонецПериода>, <Отбор>)

Подробнее о перечисленных методах – в справке 1С:Предприятия по запросу РегистрСведенийМенеджер.<Имя регистра сведений>.

Для обращения к итогам регистра накопления также предусмотре- ны объекты и методы встроенного языка:

РегистрНакопленияМенеджер.<Имя регистра накопления>

// предназначен для управления данными конкретного регистра накопления. Позволяет получать остатки и обороты по данным ре- гистра накопления, формировать выборки, создавать формы и маке- ты

Обороты(<НачалоПериода>, <КонецПериода>, <Отбор>, <Измерения>,

<Ресурсы>)

Остатки(<МоментВремени>, <Отбор>, <Измерения>, <Ресурсы>)

Подробнее о перечисленных методах – в справке 1С:Предприятия по запросу РегистрНакопленияМенеджер.<Имя регистра накопления>.

# Содержание работы

Задания Intermediate (макс. оценка за выполнение - «удовлетворит.»)

1. Создайте документ ПоступлениеТоваров, отражающий хозяй- ственную операцию по поступлению товаров от поставщика на склад организации.

Таблица 18. Свойства документа ПоступлениеТоваров

|  |  |
| --- | --- |
| **Свойство** | **Значение** |
| Подсистемы | ОтделЗакупок > Документы (подчиненную подсистему Документы нужно предварительно создать) |
| Состав реквизитов | Подразделение, тип СправочникСсылка.Подразделения; Склад, тип СправочникСсылка.Склады;  Контрагент, тип СправочникСсылка.Контрагенты; ДоговорКонтрагента, тип СправочникСсылка.Договоры; Ответственный, тип СправочникСсылка.Сотрудники; ТипЦен, тип СправочникСсылка.ТипыЦенНоменклатуры (справочник нужно предварительно создать); СуммаДокумента, тип Число:10:2, неотрицательное;  Комментарий, тип Строка:100:переменная |
| Состав табличных частей | Одна табличная часть Номенклатура с реквизитами: НоменклатурнаяПозиция, тип СправочникСсылка. Номенклатура;  Цена, тип Число:10:2, неотрицательное; Количество, тип Число:10:0, неотрицательное; Сумма, тип Число:10:2, неотрицательное; СтавкаНДС, тип ПеречислениеСсылка.СтавкиНДС;  СуммаНДС, тип Число:10:2, неотрицательное; Всего, тип Число:10:2, неотрицательное |
| Нумерация | Периодичность = В пределах года |

1. Создайте форму документа ПоступлениеТоваров. Вид формы приведен на рис. 51, 52. Обратите внимание на наличие подвала таб- личной части с итогами по колонкам Сумма, СуммаНДС и Всего.

Также необходимо предусмотреть, чтобы при заполнении документа пользователю предлагались только договоры, принадлежащие вы- бранному на форме контрагенту.

Кроме того, в соответствии с логикой документа в качестве номенкла- турной позиции (в табличой части) можно выбирать только элементы справочника Номенклатура со значением Услуга = Ложь.

1. Предусмотрите автоматические подстановки и расчеты в таб- личной части документа ПоступлениеТоваров в соответствии со следу- ющей логикой:

* изменение поля СтавкаНДС активной строки табличной части иници- ирует перерасчет полей СуммаНДС и Всего;
* изменение поля Цена инициирует перерасчет полей Сумма, СуммаНДС, Всего;
* при изменении поля Количество выполняется перерасчет аналогично случаю с изменением поля Цена;
* при изменении номенклатурной позиции в соответствующее поле подставляется ставка НДС, поле Цена обнуляется, остальные поля перерасчитываются исходя из изменения цены.

Необходимые процедуры и функции расположите в модуле формы документа ПоступлениеТоваров.

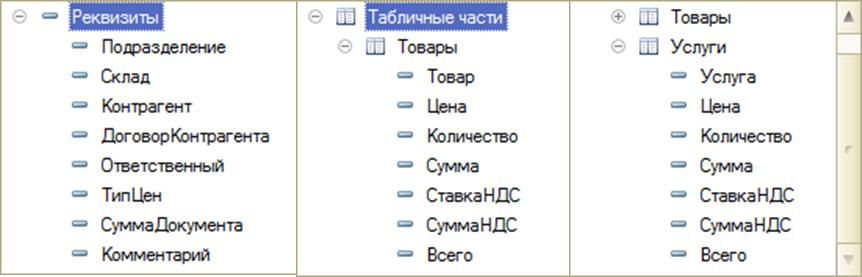
Для числовых реквизитов табличной части настройте Формат (по умолчанию поля с нулевыми значениями отображаются как пустые). Свойство можно настроить как со стороны элементов формы, так и со стороны реквизитов табличной части документа.

1. Организуйте заполнение реквизита СуммаДокумента перед за- писью документа ПоступлениеТоваров. Реквизит должен содержать итог по колонке Всего табличной части.

**Совет**. Алгоритм заполнения реквизита поместите в модуль объекта документа (используйте процедуру-обработчик ПередЗаписью).

1. Измените алгоритм формирования номера для документов ПоступлениеТоваров. Вместо номеров 000000001, 000000002 и т.п. до- кументам должны присваиваться номера с префиксом П: П00000001, П00000002 и т.п.

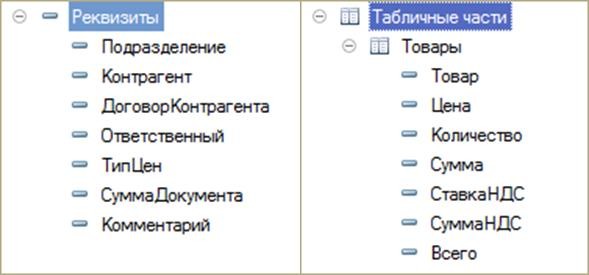
**Совет**. Алгоритм формирования номера поместите в модуль объекта документа (используйте процедуру-обработчик ПриУстановкеНового- Номера).

1. Создайте документ РеализацияТоваровУслуг. Документ отража- ет хозяйственную операцию по продаже товаров и оказанию услуг клиентам. Документ входит в состав подсистемы ОтделПродаж > Документы (дочернюю подсистему Документы заранее добавить).

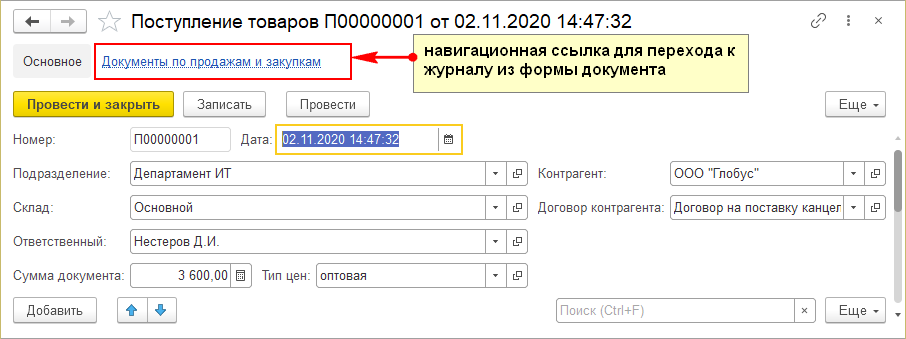
**Рис. 61 – Состав реквизитов и табличых частей документа РеализацияТоваровУслуг**

**Совет.** Структура документа РеализацияТоваровУслуг схожа со струк- турой документа ПоступлениеТоваров, поэтому для выполнения п.6. целесообразно использовать механизм создания копированием.

Для документа РеализацияТоваровУслуг выполнить п. 2-5 с учетом наличия двух табличных частей. Префикс номера для документа Р.

1. Создайте документ ЗаказКлиента (отражает хозяйственную операцию по предварительному заказу товаров клиентом) копировани- ем документа РеализацияТоваровУслуг. Удалите лишние реквизиты и табличные части, префикс номера измените на З.

**Рис. 62 – Состав реквизитов и табличых частей документа ЗаказКлиента**

1. Создайте журнал документов ДокументыПоЗакупкамПродажам. В нем регистрируются документы, созданные в предыдущих заданиях. Журнал содержит три графы: Контрагент, ТипЦен, СуммаОперации.
2. На ветви Общие дерева конфигурации создайте группы ко- манд: ЖурналыДокументов (категория Панель навигации формы) и Печать (категория Командная панель формы).
3. Разместите на форме документов всех видов навигационную команду по переходу к журналу документов, используйте ранее со- зданную группу команд ЖурналыДокументов.

**Рис. 63 – Навигационная ссылка для перехода к журналу документов из формы одного из документов**

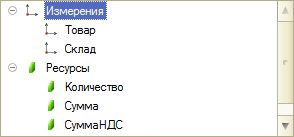
**Совет**. п.10 выполняется через командный интерфейс формы докумен- та с использованием глобальных команд.

1. Создайте макет для вывода документа РеализацияТоварУслуг на печать (с помощью конструктора печати). В конструкторе укажите, что команда печати должна располагаться в группе Командная панель формы.Печать, созданной в п. 9.
2. Добавьте в состав конфигурации регистр сведений

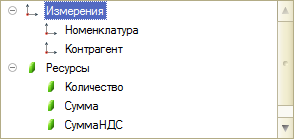
ЦеныНоменклатуры.

Таблица 19. Свойства регистра сведений ЦеныНоменклатуры

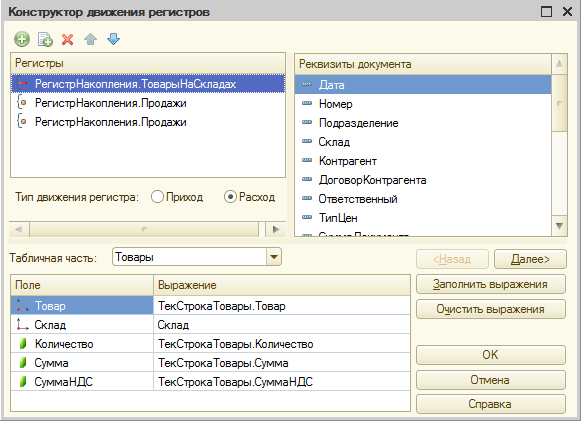
|  |  |
| --- | --- |
| **Свойство** | **Значение** |
| Синоним | Цены номенклатуры (для продажи) |
| Периодичность | В пределах дня |
| Режим записи | Независимый |
| Подсистемы | ОтделПродаж > Регистры (дочернюю подсистему Реги- стры предварительно добавить) |
| Состав измерений и ресурсов | *Ведущие измерения:*  Номенклатура, тип СправочникСсылка.Номенклатура; ТипЦен, тип СправочникСсылка.ТипыЦенНоменклатуры; *Ресурсы:*  Цена, синоним Цена (руб.), тип Число:10:2, неотрица- тельное |

1. Добавьте в состав конфигурации регистр накопления ТоварыНаСкладах с видом регистра Остатки, изменения в регистр вно- сятся документами-регистраторами ПоступлениеТоваров и Реализация- ТоваровУслуг.

**Рис. 64 – Структура регистра ТоварыНаСкладах**

1. Добавьте в состав конфигурации регистр накопления Продажи с видом регистра Обороты, изменения в регистр вносятся документом- регистратором РеализацияТоваровУслуг.

**Рис. 65 – Структура регистра Продажи**

1. Настройте видидимость созданных регистров в пользователь- ском интерфейсе (через командный интерфейс подсистем, к которым отнесены регистры).
2. С помощью конструктора движений настройте проведение до- кументов ПоступлениеТоваров (по регистру ТоварыНаСкладах) и доку- мента РеализацияТоваровУслуг (по регистрам ТоварыНаСкладах и Продажи).

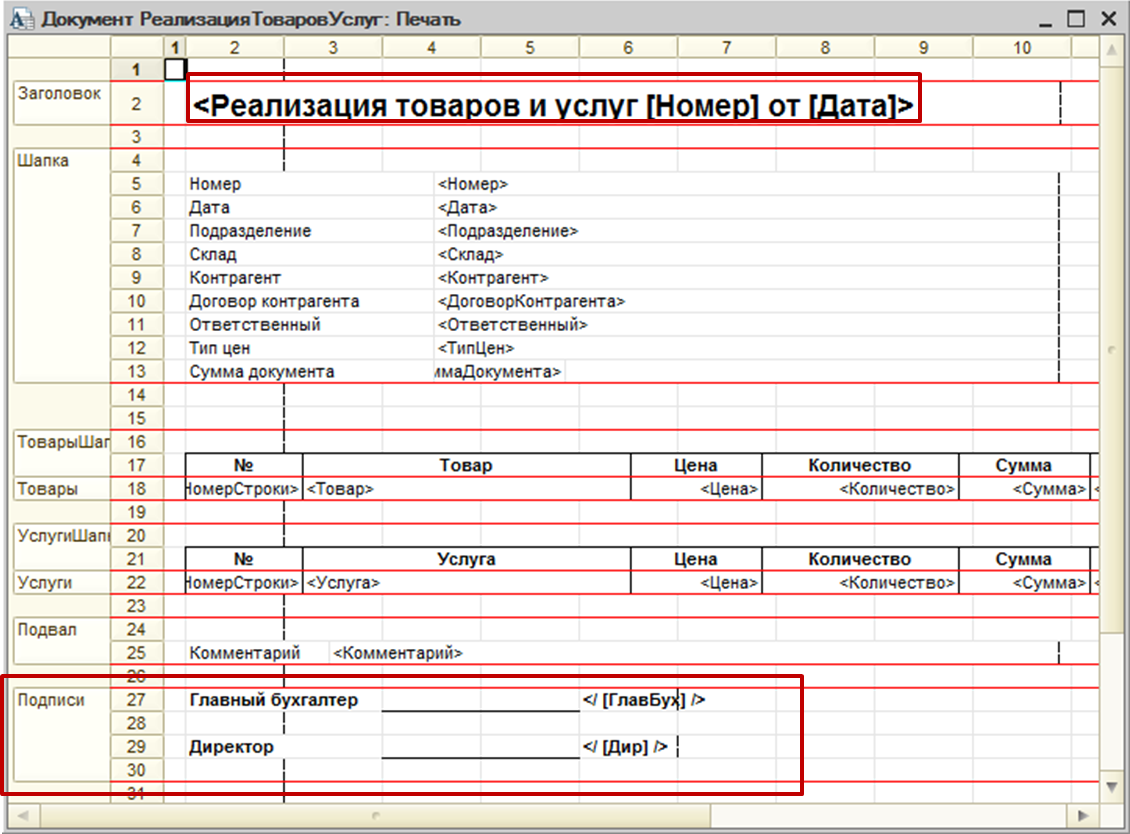
**Рис. 66 – Настройка проведения документа РеализацияТоваровУслуг в конструкторе**

1. В пользовательском режиме создайте по 4-5 экземпляров до- кументов каждого вида, заполните их и проведите по регистрам.

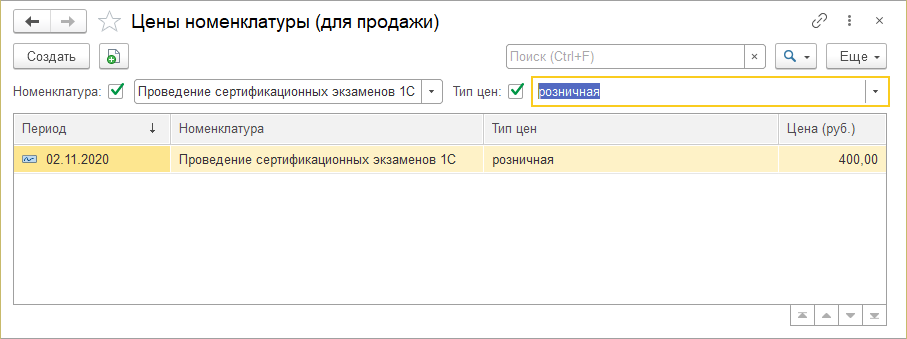
Задания Upper-Intermediate (макс. оценка за выполнение - «хорошо»)

1. Реализуйте возможность создания документа РеализацияТова- ровУслуг на основании документа ЗаказКлиента. Используйте меха- низм ввода на основании.
2. Модифицируйте макет Печать и одноименную процедуру из модуля менеджера, созданные при выполнении п. 11.

В процедуре Печать, во-первых, нужно предусмотреть заполнение па- раметров Номер, Дата из области Заголовок и параметров ГлавБух и Дир из области Подписи. Во-вторых, если документ содержит пустые таб- личные части, не нужно выводить на печать пустые шапки этих таб- лиц.



**Рис. 67 – Модифицированная печатная форма**

1. Настройте форму списка регистра сведений ЦеныНоменклату- ры: добавьте возможность отбора по номенклатуре и типу цен (задание без программирования).

**Рис. 68 – Форма списка регистра сведений**

1. Ознакомьтесь, как созданные документы и регистры реализо- ваны в типовой конфигурации 1С:Бухгалтерия предприятия.

Задания Advanced (макс. оценка за выполнение - «отлично»)

1. Через Конфигуратор с помощью инструмента Отладка > Замер производительности зафиксируйте время исполнения процедур и функ- ций (отвечающих за вычисления в табличных частях документов) мо-

дулей формы документов ПоступлениеТоваров, ЗаказКлиента, Реализация ТоваровУслуг.

1. Перенесите алгоритмы расчета/перерасчета/подстановки зна- чений табличных частей документов из модулей форм в общие моду- ли.
2. Выполнив п. 22 еще раз, проанализируйте, как изменилось время выполнения алгоритмов расчета и перерасчета.
3. Реализуйте подстановку цены продажи номенклатуры из реги- стра сведений в табличную часть документа РеализацияТоваровУслуг. Подстановка цены выполняется при изменении значения поля Товар/Услуга. Нужная цена выбирается из регистра по периоду, номен- клатуре и типу цены.

**Совет**. Используйте метод СрезПоследних периодического регистра сведений.

Факультативные задания (на оценку за выполнение не влияет)

1. При записи документа РеализацияТоваровУслуг организуйте проверку на соответствие цены продажи документа данным самой поздней записи регистра. Если цены не совпадают или в регистре нет нужных записей, программно следует обновить данные регистра по содержимому документа.

# Контрольные вопросы

1. Что такое регистр сведений и для чего он используется?
2. Для чего используется свойство Периодичность регистра сведений?
3. Что такое Режим записи для регистра сведений?
4. Что такое измерения и ресурсы для регистра сведений?
5. Какие объекты применяются для работы с регистром сведений средствами встроенного языка?
6. Что такое документ в системе 1С:Предприятие 8?
7. Какие существуют стандартные реквизиты для документов?
8. Каковы принципы нумерации документов, что такое Нумератор?
9. Что такое понятие «движения» и «проведение»?
10. Какие виды проведения документов существуют?
11. Что такое Последовательности, каковы принципы работы с последо- вательностями?
12. Для чего в системе 1С:Предприятие версии 8 применяются реги- стры?
13. Что такое регистр накопления?
14. Как можно интерпретировать измерения и ресурсы регистра?
15. Какие типы регистра накопления используются в системе 1С:Предприятие 8?
16. Как осуществляется проведение документа по нескольким реги- страм?
17. Что такое Макет в системе 1С:Предприятие 8?
18. Каким образом можно создать макет?

# Лабораторная работа №4.

**МЕХАНИЗМ ЗАПРОСОВ В СИСТЕМЕ 1С:ПРЕДПРИЯТИЕ. ОБРАБОТКИ И ОТЧЕТЫ**

*Цель работы:* освоение табличной техники доступа к учетным данным посредством языка запросов 1С:Предприятие; получение навыков разработки отчетов и обработок.

# Теоретические сведения

Табличная техника доступа к учетным данным подразумевает ис- пользование механизма запросов для чтения данных из базы.

Информация хранится в БД в виде таблиц: каждому объекту кон- фигурации сопоставляется одна или несколько таблиц БД (в зависимо- сти от наличия табличных частей в структуре объекта и других усло- вий). Поэтому при использовании механизма запросов взаимодействие с БД осуществляется на уровне записей таблиц.

## Декларативный язык запросов системы 1С:Предприятие

Запрос – инструкция системе относительно того, какие данные необходимо извлечь из БД. Такая инструкция описывается на встроен- ном языке запросов, который основывается на классическом языке SQL, но имеет ряд существенных расширений. Расширения направле- ны на поддержку работы с объектами и эффективное решение эконо- мических задач. Язык запросов оптимизирован для чтения больших массивов информации.

Запросы формируются на основе таблиц, результатом запроса так- же является таблица (частный случай: результат, состоящий из одной колонки/строки). В простейшем запросе задается два основных огра- ничения на чтение данных: 1) из какого источники следует получить данные, т.е. из каких таблиц БД; 2) что именно следует извлечь, т.е. значения каких полей таблиц БД.

*В качестве источника данных для запроса могут выступать*:

* реальные таблицы, данные которых физически хранятся в БД;
* виртуальные (расчетные) таблицы, данные которых вычисляются системой 1С:Предприятие в момент обращения к этому источнику;
* временные таблицы, которые хранятся в оперативной памяти ком- пьютера и позволяют определить в качестве источников в табличном виде абсолютно любую информацию (из незаписанного документа, из внешней СУБД, результат предшествующего запроса).

Виртуальные таблицы существуют для всех регистров. Например, обратившись к виртуальной таблице регистра накопления, можно по- лучить остатки и обороты некоторых показателей (ресурсов регистра) в разрезе измерений, определенных в структуре регистра.

Структуру запроса можно описать следующим синтаксическим правилом:

# ВЫБРАТЬ [РАЗРЕШЕННЫЕ] [РАЗЛИЧНЫЕ] [ПЕРВЫЕ <Коли-

чество>**]** <Список полей выборки **| \***> **[ИЗ** <Список таблиц источников>**]**

**[ГДЕ** <Список условий отбора>**] [ПОДОБНО]**

**[СГРУППИРОВАТЬ ПО** <Список полей группировки>**] [ИМЕЮЩИЕ** <Список условий>]

**[ДЛЯ ИЗМЕНЕНИЯ** <Список блокируемых данных>**]**

**[УПОРЯДОЧИТЬ ПО** <Список полей упорядочивания>**] [ИЕРАР- ХИЯ] | [АВТОУПОРЯДОЧИВАНИЕ]**

**[ИТОГИ** <Агрегатные функции> **ПО** <Список полей>**]**

Текст запроса состоит из нескольких секций, каждая из которых начинается с ключевого слова. Секции, заключенные в квадратные скобки, являются необязательными. Вертикальной чертой разделены взаимоисключающие выражения. Назначение секций будет рассмот- рено позднее.

*Применение запросов* в системе 1С:Предприятие:

* построение отчетов;
* получение необходимых данных в алгоритмах;
* получение данных при отладке (служебные запросы для анализа данных, получаемых на разных этапах выполнения алгоритма, и спо- собов их обработки).

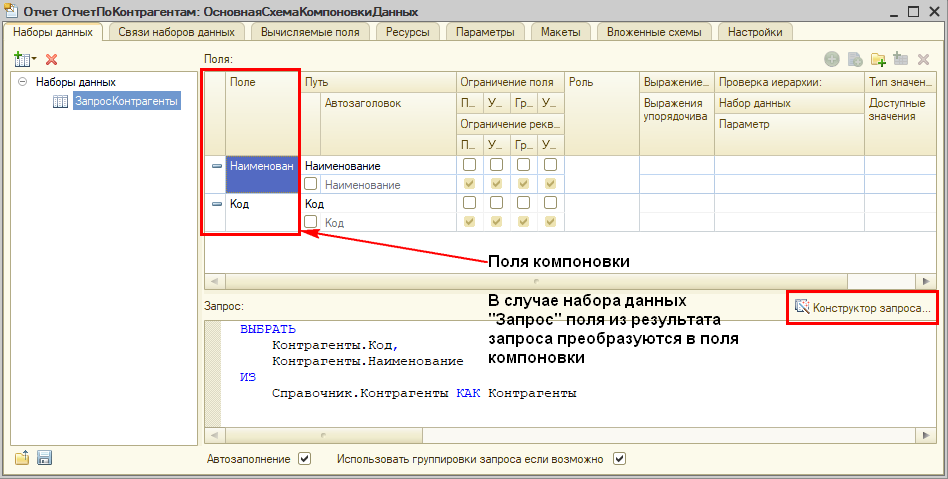
Вспомогательным инструментом для создания текстов запросов яв- ляется *конструктор запроса*, который позволяет в визуальном режиме добавить в запрос необходимые секции и заполнить их, тем самым формируя инструкцию по извлечению данных из БД, понятную плат- форме.

## Принципы и механизмы формирования отчетов

Обязательным требованием к системам автоматизации учета явля- ется наличие средств обработки накопленной в системе информации и получения сводных данных в удобном для просмотра и анализа виде.

Для получения сводной выходной информации в системе 1С:Предприятие используются объекты конфигурации Отчеты. Для интерактивной настройки вывода информации (т.е. для создания отче- тов без программирования) на уровне платформы (с версии 8.2) зало- жен механизм – *Система компоновки данных*, или сокращенно *СКД*. Данный механизм интегрирован в объект конфигурации Отчет (свой- ство Основная схема компоновки данных). СКД возникла из необходи- мости предоставить пользователю возможность самостоятельно пред- ставлять результаты одного и того же запроса в отчете в различных вариантах.

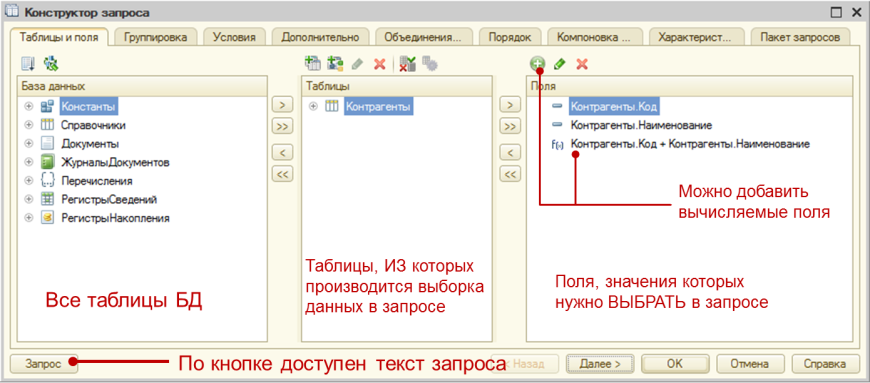
**Важно!** Основная особенность отчетов на основе СКД: результаты запроса, заложенного в отчет разработчиком, могут быть представле- ны пользователем в различных вариантах с помощью отборов, сорти- ровки, условного оформления, использования параметров отчета и др.

После создания объекта конфигурации Отчет и отнесения его к су- ществующим подсистемам необходимо вызвать конструктор макета, нажав на кнопку Открыть схему компоновки данных (вкладка Основные мастера отчета). В названном конструкторе создается макет отчета типа Схема компоновки данных. Дальнейшая работа по интерактивному созданию отчета производится в конструкторе схемы компоновки дан- ных, который запускается после создания макета (или при редактиро- вании отчета).

**Рис. 69 – Вид конструктора схемы компоновки данных**

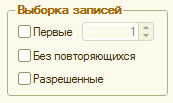
Конструктор СКД позволяет описать исходные данные, которые будет использовать отчет: наборы данных (источники данных), связи между ними, вычисляемые поля, параметры, ресурсы, настройки и др. Результат работы конструктора – схема компоновки данных, полно- стью описывающая отчет. С полями компоновки данных в дальней- шем работает и разработчик, и пользователь.

В качестве набора данных, который используется в отчете, может выступать Запрос, Объект или Объединение. Если определен набор данных Запрос, на вкладке Наборы данных конструктора СКД доступен конструктор запроса.

На вкладке Таблицы и поля конструктора запроса описывается спи- сок таблиц-источников данных и список полей для выборки из БД в запросе. Данная вкладка отвечает за формирование секций ВЫБРАТЬ и ИЗ текста запроса. Список полей выборки может содержать: поля реальных и виртуальных таблиц; выражения (вычисляемые, логиче- ские, строковые), записанные на языке схемы компоновки данных; вложенные таблицы.

**Рис. 70 – Вкладка «Таблицы и поля» конструктора запросов**

При описании полей выборки можно присвоить им псевдонимы на вкладке Объединения/Псевдонимы конструктора запроса (колонка Имя поля), что соответствует секции КАК текста запроса.

На вкладке Дополнительно конструктора запроса можно указать ряд настроек для выборки записей:

**Рис. 71 – Дополнительные параметры выборки записей в запросе**

Флаг Первые соответствует ключевому слову ПЕРВЫЕ в тексте за- проса. Конструкция позволяет задать предельное количество строк в результате запроса. Будут отобраны самые первые (в соответствии с правилами упорядочивания) строки.

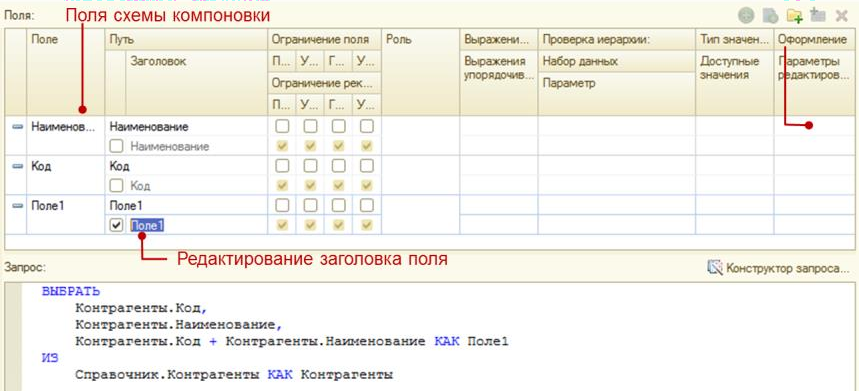
Флаг Без повторяющихся соответствует ключевому слову РАЗЛИЧ- НЫЕ в тексте запроса. Конструкция убирает повторяющиеся записи из результата запроса.

Флаг Разрешенные соответствует ключевому слову РАЗРЕШЕН- НЫЕ в тексте запроса. Конструкция означает, что запрос выберет только те записи, на которые у текущего пользователя есть права. Если запрос возвращает записи, на которые у пользователя нет прав, гене- рируется сообщение «У пользователя недостаточно прав на исполне- ние операции над БД».

В ряде случаев необходимо блокировать считываемые в запросе данные для последующего изменения вне запроса. Такая возможность реализуется добавлением секции ДЛЯ ИЗМЕНЕНИЯ в текст запроса, что в конструкторе запроса соответствует флагу Блокировать получае- мые данные… на вкладке Дополнительно. Считанные данные становятся недоступными для чтения в других сессиях. Блокировка снимается после завершения транзакции.

Для сортировки записей результата запроса по алфавиту или чис- ловому полю используются настройки вкладки Порядок конструктора запроса, которые соответствуют секции УПОРЯДОЧИТЬ ПО текста запроса. Предусмотрено несколько вариантов сортировки: возрастание (по умолчанию), убывание, иерархия, иерархия по убыванию. Дей- ствует сортировка по нескольким полям. Настройки Иерархия / Иерар- хия по убыванию имеют смысл для иерархических таблиц и добавляют в результат запроса группы элементов. При этом сортировка работает как для групп, так и для элементов внутри этих групп. При использо- вании опции АВТОУПОРЯДОЧИВАНИЕ поля для сортировки резуль- тата запроса будут формироваться автоматически по правилам, опре- деленным на уровне платформы.

Для создания простейшего запроса обязательным условием являет- ся заполнение вкладки Таблицы и поля конструктора запроса. Как толь- ко запрос сформирован, поля запроса автоматически преобразуются в поля схемы компоновки отчета, с которыми может работать и разра- ботчик, и пользователь.



**Рис. 72 – Поля схемы компоновки данных**

В области Поля (конструктор СКД > вкладка Наборы данных) до- ступно множество опций, например:

* Заголовок полей, выводимых в отчет; опция используется для быст- рого редактирования автозаголовка;
* Ограничение поля используется, если необходимо сделать поле недо- ступным для упорядочивания, группировки и др.;
* Оформление позволяет установить формат поля: цвет текста, шрифт, стиль границы, формат представления чисел/дат, ориентация текста и т.п.

Для вывода в отчет сформированного набора данных необходимо настроить вариант отображения данных (вкладка Настройки конструк- тора схемы компоновки данных), т.е. указать платформе, как именно будет представлена информация пользователю. По умолчанию для отчета опреден один вариант представления данных, его необходимо настроить с помощью Конструктора настроек компоновки данных.

**Рис. 73 – Настройки варианта отчета в конструкторе СКД**

*1* – кнопка для добавления варианта представления отчета;

*2* – конструктор настройки текущего варианта отчета;

*3* – меню для ручной настройки варианта отчета

На первом шаге в конструкторе настроек следует выбрать тип отче- та (список, таблица или диаграмма). Далее нужно выбрать поля, кото- рые будут отображаться в отчете. Для отчета типа Список или Таблица также можно указать поля, по которым будет производиться группи- ровка и упорядочивание данных. Настройка отчета с диаграммой бу- дет рассмотрена позднее.

Представление отчета, созданного на основе СКД, может быть мо- дифицировано в пользовательском режиме и сохранено как новый ва- риант отчета. Для этого используются команды Изменить вариант… и Сохранить вариант… из меню Еще формы отчета.

## Табличная модель получения данных из справочников и документов

В качестве примера простого отчета на основе СКД рассмотрим от- чет ПоставщикиНоменклатуры, который должен выводить номенклатур- ные позиции, сгруппированные по основному поставщику. Перед со- зданием отчета внесем необходимые изменения в конфигурацию:

* добавим новый реквизит ОсновнойПоставщик в справочник Номенкла- тура;
* создадим форму элемента справочника Номенклатура;
* для элемента формы, соответствующего реквизиту ОсновнойПостав- щик, определим обработчик события НачалоВыбора. В процедуре- обработчике обеспечим, чтобы выбор основного поставщика произ- водился только из группы Поставщики справочника Контрагенты. Для этой задачи можно использовать несколько вариантов реализации:

Процедура ОсновнойПоставщикНачалоВыбора

(Элемент, ДанныеВыбора, СтандартнаяОбработка)

//1й вариант - без открытия формы выбора,

//с выбором из выпадающего списка

//СтандартнаяОбработка = Ложь;

//СформироватьДанныеВыбора(ДанныеВыбора);

//2й вариант - открыть форму выбора с установленными

//параметрами выбора Отбор.Родитель = Поставщики

//при этом не отказываемся от стандартной обработки открытия

//формы

НовыйПараметр = Новый ПараметрВыбора("Отбор.Родитель",

ПолучитьСсылкуПоставщиков());

НовыйМассив = Новый Массив; НовыйМассив.Добавить(НовыйПараметр);

НовыеПараметры = Новый ФиксированныйМассив(НовыйМассив); Элементы.ОсновнойПоставщик.ПараметрыВыбора = НовыеПараметры;

//3й вариант - аналогичное можно сдедать без программирования,

//настроив Параметры выбора для реквизита ОсновнойПоставщик

//но тогда необходима предопределенная группа Поставщики КонецПроцедуры

&НаСервереБезКонтекста

Функция ПолучитьСсылкуПоставщиков()

//вернуть ссылку на группу Поставщики справочника Номенклатура КонецФункции // ПолучитьСсылкуПоставщиков()

&НаСервереБезКонтекста

Процедура СформироватьДанныеВыбора(ДанныеВыбора) Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Контрагенты.Ссылка

|ИЗ

| Справочник.Контрагенты КАК Контрагенты

|ГДЕ

| Контрагенты.Родитель.Наименование = ""Поставщики"""; РезультатЗапроса = Запрос.Выполнить().Выгрузить(); МассивЗначений = РезультатЗапроса.ВыгрузитьКолонку("Ссылка"); ДанныеВыбора = Новый СписокЗначений; ДанныеВыбора.ЗагрузитьЗначения(МассивЗначений);

КонецПроцедуры //СформироватьДанныеВыбора

**p.s.** С листингом требуется разобраться самостоятельно.

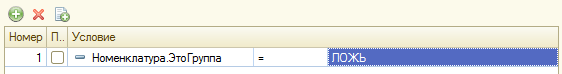
Перейдем к созданию отчета на основе СКД. Набор данных - за- прос к таблице Контрагенты, поля выборки – Основной поставщик и Ссылка (стандартный реквизит, уникально идентифицирующий любой объект). Добавление поля Ссылка в список полей выборки имеет в го- товом отчете следующий эффект: в пользовательском режиме из стро- ки отчета возможно открыть форму объекта, который идентифициру- ется по ссылке.

На вкладке Объединения/Псевдонимы конструктора запроса опреде- лим для поля Ссылка более понятное имя – Номенклатура.

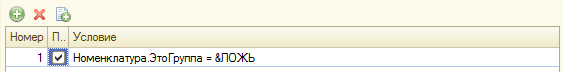
Наложим несколько условий на результат запроса (вкладка Условия конструктора запроса). Заданные условия отобразятся в секции ГДЕ текста запроса. Механизм следующий: в результат запроса попадают только те записи, для которых условия отбора оказываются истинны- ми. Условия отбора начинают действовать после того, как из таблицы- источника выбраны все записи. Поля-условия можно не указывать в полях выборки, а задействовать только в секции с условиями. В усло- виях отбора можно использовать логические операторы НЕ, И, ИЛИ, а также формировать произвольные условия на *языке выражений системы компоновки данных* (с помощью конструктора произвольных выраже- ний, который осуществляет синтаксический контроль выражений). Подробнее про язык выражений – справка 1С:Предприятия > Система компоновки данных > Язык выражений системы компоновки данных.

Условимся, что в отчет не должны попадать группы номенклатуры.

Добавляем на вкладку Условия конструктора запроса первую запись:

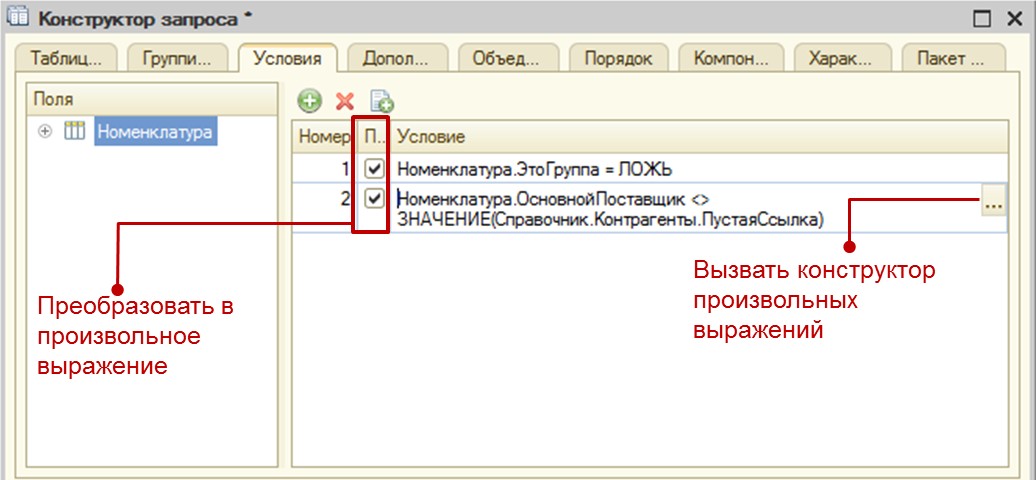


**Рис. 74 – Условие по отбору в запросе элементов справочника**

Если оставить условие в таком виде, то платформа воспримет ЛОЖЬ как параметр запроса, а не как значение типа Булево. Поэтому необходимо преобразовать первое условие в произвольное выражение, установив флаг П.. в конструкторе.

**Рис. 75 – Условие в виде произвольного выражения**

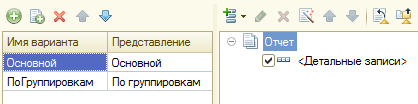
Параметры в тексте запроса задаются через &. В данном случае следует удалить & из выражения.

Также условимся, что в отчет не должны попадать номенклатурные позиции, для которых не задан основной поставщик. Зададим второе произвольное условие, в котором будем сравнивать значение поля Ос- новнойПоставщик со значением пустой ссылки (с помощью функции ЗНАЧЕНИЕ языка выражений компоновки данных).

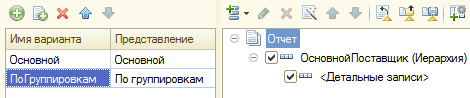
**Рис. 76 – Условия, сформированные в конструкторе произвольных выражений**

На этом работа в конструкторе запроса завершена. Переходим в конструктор СКД на вкладку Настройки для формирования варианта отображения отчета. Вызвав конструктор настроек схемы компоновки данных, выбираем Список в качестве типа отчета. Далее укажем, что в отчет нужно выводить поля Номенклатура и ОсновнойПоставщик.

Создадим второй вариант отчета – ПоГруппировкам. В конструк- торе настроек схемы компоновки данных проделаем аналогичные дей- ствия. Дополнительно укажем поле ОсновнойПоставщик в качестве по-

ля, по которому будет производиться группировка номенклатурных позиций (тип группировки Иерархия).

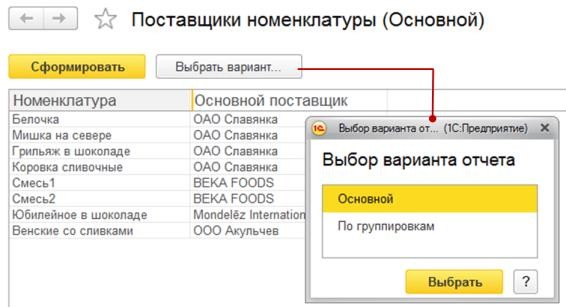
(а)



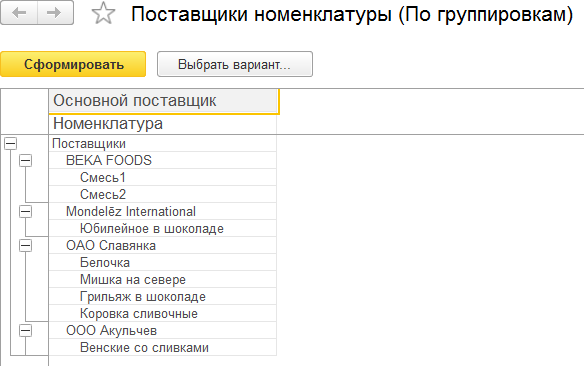
(б)

**Рис. 77 – Структура вариантов отчета Поставщики номенклатуры**

(а) – основной вариант отчета; (б) – вариант отчета с группировкой по поставщику

Между созданными вариантами отчета можно переключаться в пользовательском режиме:

**Рис. 78 – Выбор варианта отчета в пользовательском режиме**



**Рис. 79 – Второй вариант отчета Поставщики номенклатуры (с группировкой)**

В отчете Поставщики номенклатуры (вариант По группировкам) на первом уровне отображается группа Поставщики справочника Контр- агенты, на втором уровне – элементы этой группы, на третьем уровне – номенклатурные позиции.

В предыдущей работе рассматривался принцип учета «от докумен- та» в системе 1С:Предприятие. В соответствии с этим принципом дан- ные о хозяйственной деятельности обрабатываются по схеме Доку- менты > Регистры > Отчеты. Поэтому для системы учета, реализован- ной на платформе 1С:Предприятие, корректным подходом является построение отчетов по данным регистров. Но в реальных учетных за- дачах и типовых конфигурациях есть примеры построения отчетов по документам – это корректно в том случае, если документ не делает движений по регистрам. Поэтому в лабораторной работе будут приве- дены примеры и задания по построению отчетов в том числе на основе данных из документов.

Выполняя запросы к документам и их табличным частям (ТЧ), нужно помнить о том, что документ с ТЧ представлен в базе несколь- кими таблицами (каждая табличная часть хранится в виде отдельной таблицы). Общая рекомендация такова: если необходимо получить данные и из документа, и из его табличной части, то в качестве источ- ника данных в запросе следует выбирать таблицу нижнего уровня (т.е. таблицу, соответствующую табличной части документа). В такой таб- лице всегда присутствует поле Ссылка (на документ, которому при- надлежит табличная часть). Путем *разыменования* этого поля можно обратиться к реквизитам верхнего уровня (т.е. к реквизитам самого документа) и добавить их в список полей выборки. В ряде случаев разыменование позволяет избежать использования в запросе несколь- ких таблиц. При наличии нескольких таблиц-источников данных воз- никает необходимость установки связей между ними (в конструкторе запросе появляется вкладка Связи).

Рассмотрим ряд возможностей конструктора запроса и конструкто- ра схемы компоновки данных, которые часто применяются при подго- товке отчетов.

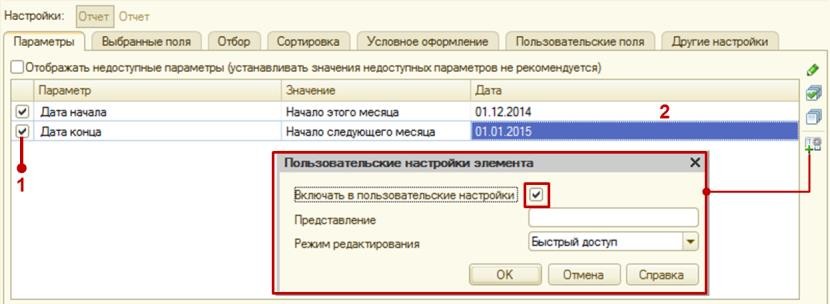
Например, на вкладке Условия конструктора запроса определено ограничение на дату документа РеализацияТоваровУслуг (обращаемся к дате документа путем разыменования поля Ссылка табличной части Товары этого документа):

РеализацияТоваровУслугТовары.Ссылка.Дата | МЕЖДУ | ДатаНачала | ДатаКонца

В данном случае ДатаНачала и ДатаКонца являются параметрами, которые определил разработчик. В тексте запроса ограничение будет выглядеть следующим образом:

ГДЕ

РеализацияТоваровУслугТовары.Ссылка.Дата МЕЖДУ &ДатаНачала И &ДатаКонца

Как только в конструкторе запроса определы параметры, они авто- матически появляются и в схеме компоновки данных (вкладка Пара- метры конструктора схемы компоновки). На вкладке Параметры можно изменить тип параметра, снять Ограничение доступности (если параметр должен быть доступен пользователю), сделать параметр обязательным для заполнения (флаг Запрещать незаполненные значения). Если пред- полагается, что пользователь должен работать с параметрами в форме отчета, то их необходимо включить в пользовательские настройки. Это действие выполняется на вкладке Настройки конструктора схемы ком- поновки в одноименном разделе:

**Рис. 80 – Включение параметров схемы компоновки в пользовательские настройки**

1 – флажки, отвечающие за активность полей с параметрами; 2 – поля для установки значений по умолчанию для параметров

При использовании ограничений по периоду нужно помнить, что отчет формируется по начало дня, указанного в параметре (штатное поведение платформы). Обычно же пользователи ожидают, что отчет формируется по указанную дату включительно. Тогда параметр, соот- ветствующий концу периода, за который формируется отчет, можно принудительно привести к концу дня (на вкладке Параметры конструк- тора схемы компоновки, поле Выражение). Как упоминалось ранее, выражение составляется на *языке выражений системы компоновки дан- ных*. Следующее выражение позволяет привести дату, переданную в запрос через параметр КонецПериода, к концу дня:

КонецПериода(&КонецПериода, "День")

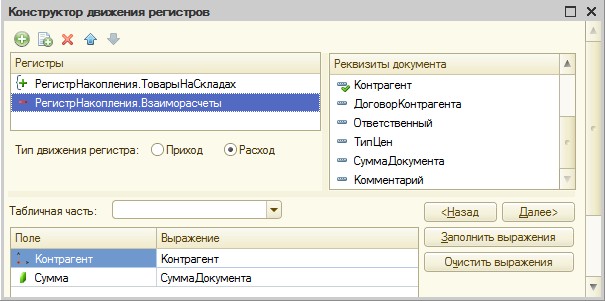
На вкладке Ресурсы конструктора схемы компоновки данных мож- но перевести числовые поля схемы компоновки в ресурсы. Это означа- ет, что система компоновки данных сможет выполнять арифметиче- ские операции (сумма, среднее, максимум, минимум, количество) над этими полями и автоматически формировать итоги по указанным по- лям.

## Получение данных из регистров для построения отчетов

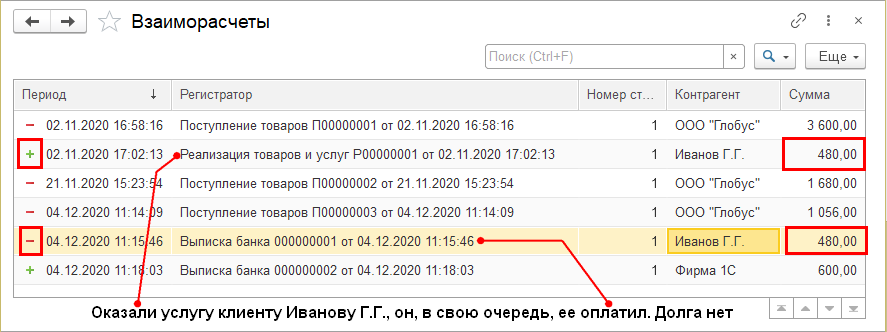
Ранее отмечалось, что на практике источником данных для постро- ения отчетов являются регистры. Например, преимущество регистров накопления в том, что они хранят просуммированные данные, что об- легчает получение остатков и оборотов по каким-либо показателям.

Для любого регистра на уровне платформы заложен набор вирту- альных таблиц, которые можно использовать в запросах: СрезПервых и СрезПоследних для регистра сведений; Обороты для регистра накопле- ния вида Обороты; Обороты, Остатки и ОстаткиИОбороты для регистра накопления вида Остатки. Физическая таблица содержит данные обо всех движениях регистра. При построении отчета часто требуются остатки, обороты показателей за конкретный период, а не все движе- ния регистра. Остатки и обороты как раз можно получить из виртуаль- ных таблиц. Виртуальные таблицы регистров физически не хранятся в БД, а формируются автоматически (в зависимости от своих парамет- ров) в момент запроса к ним.

Рассмотрим задачу построения упрощенного отчета, показывающе- го сумму долга каждого клиента перед организацией и долгов органи- зации перед поставщиками на заданную дату. Предварительно потре- буется внести изменения в рабочую конфигурацию:

* создать документ ВыпискаБанка с двумя табличными частями - При- ходДенежныхСредств и РасходДенежныхСредств; реквизиты табличных частей – Контрагент и Сумма; строки первой табличной части будут отражать уменьшение задолженности некоторого клиента перед ор- ганизацией, а строки второй табличной части – уменьшение задол- женности организации перед поставщиками;
* создать регистр накопления Взаиморасчеты с видом Остатки, измере- нием Контрагент и ресурсом Сумма; регистраторы – документы По- ступлениеТоваров, РеализацияТоваровУслуг и ВыпискаБанка;
* настроить проведение документа ВыпискаБанка по регистру Взаимо- расчеты: тип движения по табличной части ПриходДенежныхСредств
  + Расход (т.е. если денежные средства поступили к нам, то сумму долга контрагента уменьшаем), по второй табличной части – Приход;
* настроить проведение документа ПоступлениеТоваров по регистру Взаиморасчеты: (т.е. если мы приняли товары на некоторую сумму, то сумму долга контрагента уменьшаем).

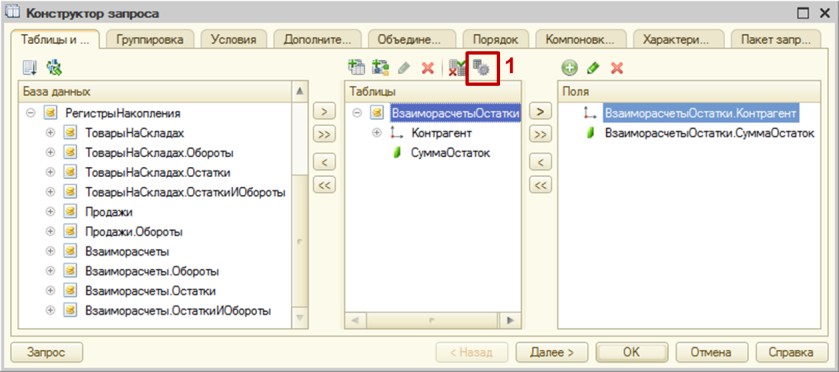
**Рис. 81 – Создание движений регистра Взаиморасчеты**

Аналогичную рисунку 81 настройку нужно выполнить и для по- следнего документа-регистратора РеализацияТоваровУслуг, но тип дви- жения регистра – Приход (если мы продали товары/оказали услуги на некоторую сумму некоторому контрагенту, то сумму долга этого контрагента увеличиваем на сумму реализованного товара/услуги).

**Рис. 82 – Пример записей регистра Взаиморасчеты**

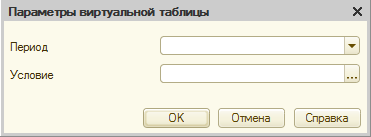
Добавим в состав конфигурации новый объект – отчет ОстаткиВзаи- морасчетов. Создадим для него схему компоновку данных и набор дан- ных Запрос. Дальнейшие действия будем выполнять в конструкторе запроса.

В конструкторе запроса (вкладка Таблицы и поля > область База данных) регистр накопления Взаиморасчеты представлен сразу четырь- мя таблицами: физическая таблица записей регистра и виртуальные таблицы. В соответствии с постановкой задачи выходной информаци- ей являются остатки долгов в разрезе контрагентов. Поэтому источни- ком данных для отчета будет выступать виртуальная таблица Взаимо-

расчеты.Остатки, в которой определено поле СуммаОстаток (постфикс Остаток означает, что по полю Сумма регистра подсчитываются остат- ки).

**Рис. 83 – Создание запроса для отчета Остатки взаиморасчетов**

1 – доступ к параметрам виртуальной таблицы регистра

Виртуальные таблицы регистров имеют предопределенные пара- метры. Для виртуальной таблицы остатков определены параметры Пе- риод и Условие:

**Рис. 84 – Параметры виртуальной таблицы остатков**

Далее зададим псевдоним Долг для поля СуммаОстаток (вкладка Объединения/Псевдонимы конструктора запроса). Сохраняем запрос и возвращаемся в конструктор схемы компоновки данных. Обеспечим вывод в отчет общих итогов: перенесем поле Долг в ресурсы (вкладка Ресурсы конструктора СКД).

Остается обеспечить в отчете ОстаткиВзаиморасчетов получение остатков на заданную пользователем дату. Без этого параметра остатки будут выдаваться на момент последней операции в системе. В пара- метрах виртуальной таблицы (рис. 84) присутствует параметр Период, его и будем использовать, чтобы пользователь мог выбирать период, на который нужно сформировать остатки. Данный параметр по умол- чанию отображается в списке на вкладке Параметры конструктора СКД. Оставим для параметра тип Дата, но изменим состав даты с Дата

и время на Дата, чтобы пользователю не пришлось задавать значение параметра с точностью до секунд.

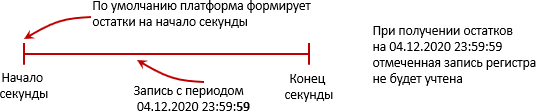
Получение остатков из регистра накопления на заданную дату име- ет несколько особенностей, которые определяются штатным поведе- нием платформы 1С:Предприятие.

Во-первых, остатки формируются на начало периода, т.е. на начало дня, указанного в параметре Период, что совсем не очевидно для поль- зователя. Поэтому принудительно будем приводить значение парамет- ра к концу дня:

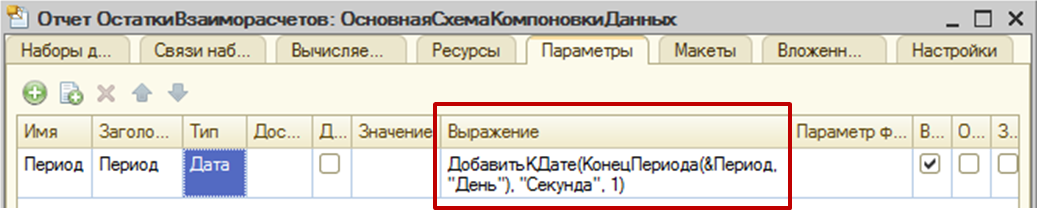
КонецПериода(&Период, "День")

Функция КонецПериода языка выражений системы компоновки да- ных с параметром День преобразует дату с 0:00:00 ко времени 23:59:59.

Во-вторых, все записи регистра накопления (вида Остатки), кото- рые имеют период с точностью до секунды, условно говоря, на оси времени находятся старше этой секунды (между текущей секундой и следующей). Поэтому, когда мы получаем остатки на конкретную се- кунду, система не включает сюда все данные, проведенные в рамках этой секунды.

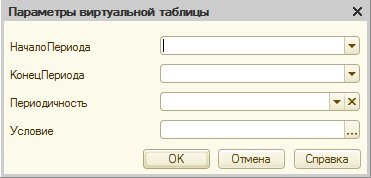


**Рис. 85 – Особенности формирования остатков регистра накопления**

Все записи, имеющие конкретное время t с точностью до секунд, старше этого времени, но младше следующей секунды, поэтому для получения корректного остатка необходимо получать данные на сле- дующую секунду. С учетом вышесказанного выражение для преобра- зования параметра Период отчета ОстаткиВзаиморасчетов выглядит сле- дующим образом:

**Рис. 86 – Выражение для преобразования параметра Период**

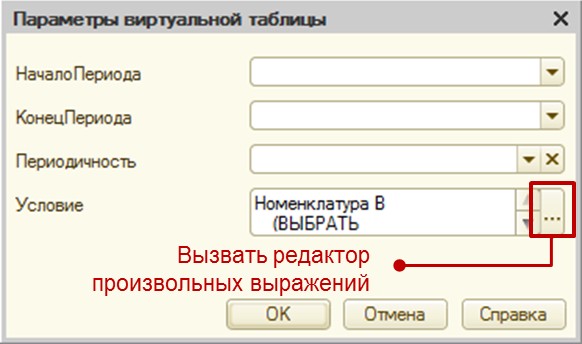
Остается настроить представление отчета (в виде обычного списка без группировок) и включить параметр Период в пользовательские настройки (на вкладке Настройки конструктора СКД). Окончательный вид отчета см. в п.4 содержания работы. В отчете присутствуют поло- жительные числа (дебиторская задолженность, т.е. долг клиентов пе- ред организацией) и отрицательные числа (кредиторская задолжен- ность, т.е. долг организации перед поставщиками).

Для оборотных регистров накопления предопределена одна вирту- альная таблица - Обороты, из которой можно получить обороты ресур- сов за определенный период. Виртуальная таблица Обороты получает данные по конец секунды (не нужно принудительно добавлять одну секунду как при получении данных из виртуальной таблицы Остатки регистра накопления). Виртуальная таблица (ВТ) Обороты имеет сле- дующие параметры:

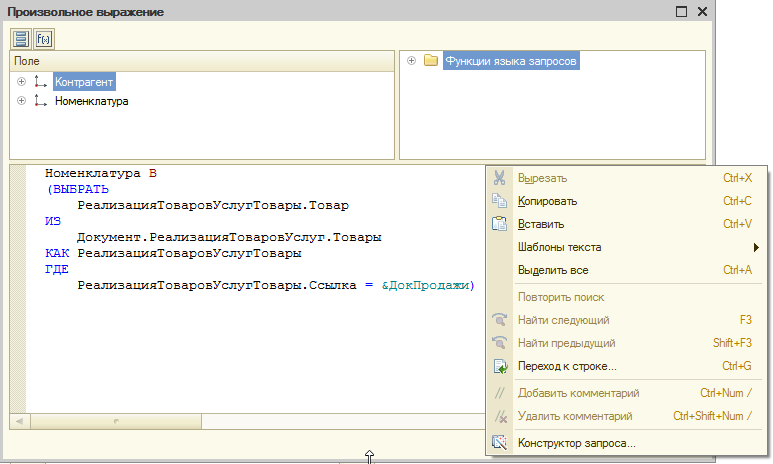
**Рис. 87 – Параметры ВТ Обороты оборотного регистра накопления**

В параметрах виртуальных таблиц допускается использование под- запросов (вложенных запросов), что позволяет организовать фильтра- цию выходных данных. Используем этот подход для решения услож- ненной задачи получения данных о продажах товара: в отчет должна попадать информация о продаже тех товаров, которые содержатся в табличной части документа РеализацияТоваровУслуг, указанного в каче- стве параметра в форме отчета.

Для решения поставленной задачи можно использовать параметр Условие виртуальной таблицы Продажи.Обороты регистра накопления Продажи (типа Обороты). Условие можно накладывать на измерения в редакторе произвольных выражений, который вызывается из поля па- раметра. Для решаемой задачи условие следующее: номенклатура, включаемая в отчет, должна входить в список товаров табличной части выбранного документа. Условие запишем на языке выражений систе- мы компоновки данных, используя оператор В (оператор проверки вхождения в список). Для получения списка товаров обратимся в под- запросе к табличной части документа. Для написания подзапроса удобнее всего воспользоваться конструктором запроса.



**Рис. 88 – Заполнение параметра Условие виртуальной таблицы регистра**



**Рис. 89 – Вложенный запрос в параметре виртуальной таблицы**

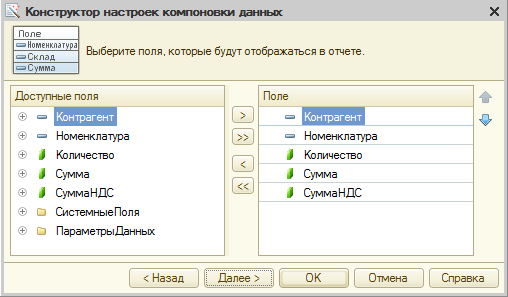
Для того чтобы параметр ДокПродажи вложенного запроса коррект- но распознавался, на вкладке Параметры конструктора СКД нужно назначить ему тип ДокументСсылка.РеализацияТоваровУслуг.

Ограничение по номенклатуре можно было бы задать и через вкладку Условия конструктора запроса, не используя вложенный за- прос. Но общая рекомендация такова: условия на измерения регистров нужно задавать в параметрах виртуальной таблицы. Это требование обусловлено вопросами производительности (скорости формирования отчета).

## Построение отчетов типа Таблица и Диаграмма

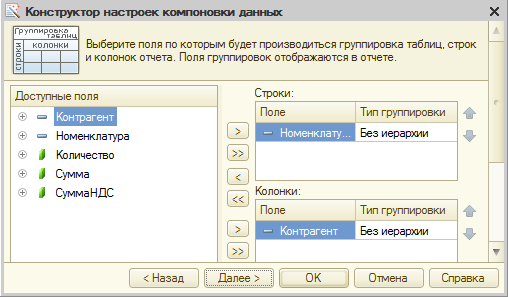
Система компоновки данных позволяет формировать отчеты, в ко- торых одна и та же информация может быть представлена в разных разрезах и разных формах представления (в виде линейных списков, списков с группировками, таблиц, диаграмм).

Наряду со списками еще одним способом вывода информации в отчет являются таблицы (кросс-таблицы). Например, в таких таблицах измерения регистра располагаются в строках и колонках, на их пере- сении выводятся значения ресурсов.

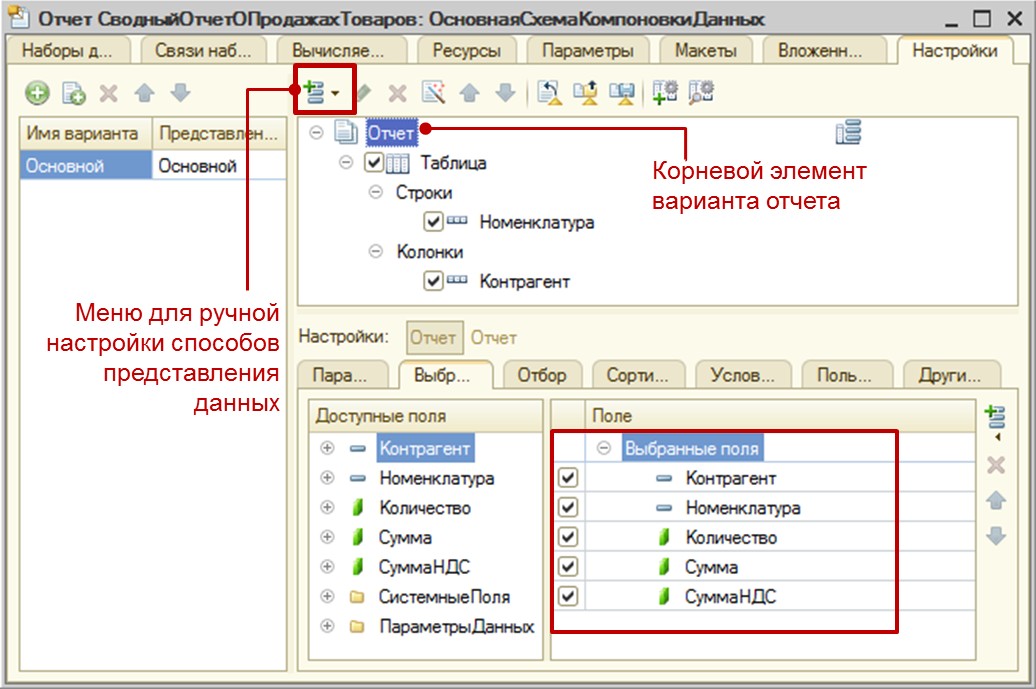
Условимся, что в конфигурации присутствует отчет, отражающий обороты продаж товаров (в количественном и суммовом выражении) за указанный период в разрезе контрагентов. Сформируем в конструк- торе СКД настройку варианта отчета в виде таблицы (шаг 1). Для это- го в схеме компоновки данных должны быть определены ресурсы, иначе на пересечении строк и колонок нечему будет выводиться.

**Рис. 90 – Шаг 2: выбор полей, которые будут отображаться в таблице**

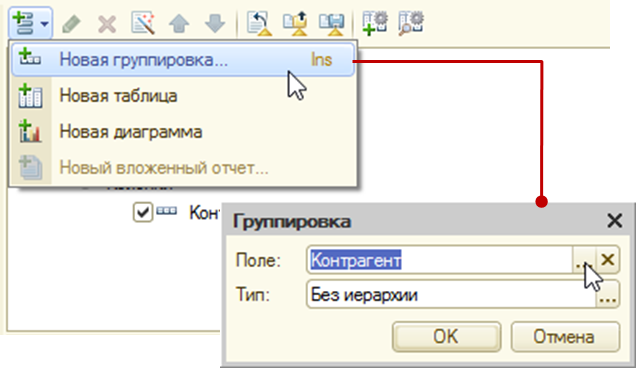
Отметим, что на третьем шаге ресурсы не выбираем, они автомати- чески будут выводиться на пересечении строк, колонок. Нажимаем OK. Усложняем задачу. В рамках этого же отчета нужно получить вто- рой вариант представления данных, не прибегая к созданию нового варианта отчета. Второй вариант представления выводных данных должен отражать продажи по клиентам, без разделения на номенкла- туру (в виде линейного списка). Конструктор СКД позволяет форми- ровать сколько угодно способов представления данных отчета в рам- ках одного варианта. В предыдущих примерах описывалось использо- вание конструктора настроек. Он позволяет сформировать первый

способ представления выходной информации, все последующие спо- собы формируются вручную путем добавления новых группировок, таблиц, диаграмм.

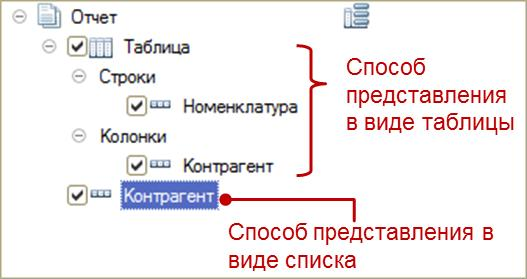
**Рис. 91 – Шаг 3: выбор полей (измерений регистра), которые будут отображаться в строках и колонках таблицы**

**Важно!** Поля, доступные для корневого элемента варианта отчета, до- ступны и для всех подчиненных элементов.

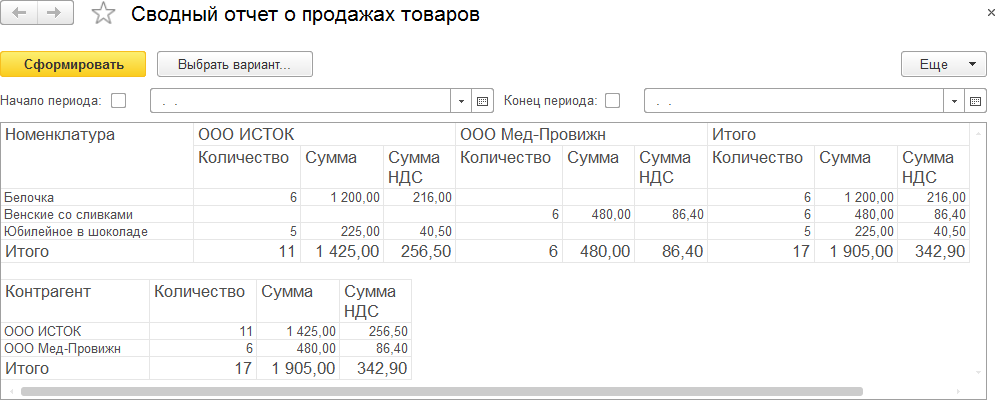
**Рис. 92 – Настройка способов представления выходной информации**

Используя меню для ручной настройки способов представления выходой информации, добавим новую группировку, подчиненную корневому элементу Отчет:

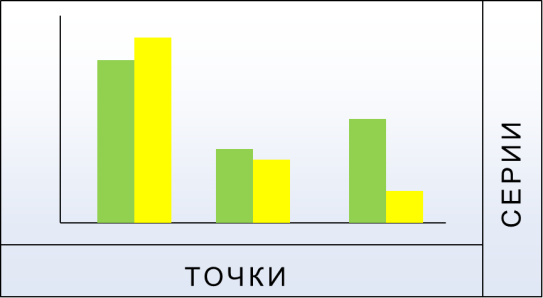
**Рис. 93 – Создание вручную второго способа представления выходных данных**

После всех манипуляций сводный отчет по продажам (с двумя спо- собами представления одного и того же набора данных) имеет вид:

**Рис. 94 – Структура сводного отчета**

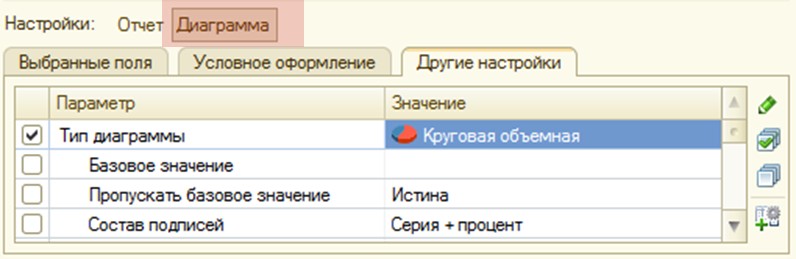


**Рис. 95 – Сводный отчет в пользовательском режиме**

Третьим способом представления данных в отчете является исполь- зование диаграмм. Диаграммы состоят из точек и серий. Для построе- ния диаграммы необходимо наличие в схеме компоновки данных хотя бы одного ресурса, поскольку в разрезе точек выводятся серии значе- ний (ресурсов).

**Рис. 96 – Структура диаграммы**

Для добавления диаграммы в констукторе настроек нужно выбрать в качестве типа отчета соответствующий вариант. На втором шаге ука- зываются поля выборки, которые будут отображаться в отчете (на этом этапе обязательно нужно выбрать ресурс в числе прочих полей). На третьем шаге указываются поля, по которым будет производиться группировка серий и точек диаграммы. Далее при необходимости ука- зываются поля упорядочивания. На последнем шаге выбирается тип диаграммы. Система 1С:Предприятие предлагает обширный набор типов диаграмм: от стандартных (график, гистограмма, круговая диа- грамма) до специфических (воронки продаж, биржевые диаграммы). Тип диаграммы по умолчанию – объемная гистограмма.

Если возникает необходимость изменить тип диаграммы, необхо- димо спозиционироваться на ней в структре отчета, щелкнуть по име- ни диаграммы в строке Настройки и перейти на вкладку Другие настрой- ки (параметр Тип диаграммы).

**Рис. 97 – Изменение типа диаграммы**

Параметры виртуальных таблиц регистров можно применять не только для фильтрации выходных данных, но и при построении диа- грамм. Продемонстрируем на примере построения диаграммы типа График. Такие диаграммы, как правило, строятся в разрезе оси време- ни.

Задача: сформировать график продаж по дням в разрезе клиентов. Обратимся к виртуальной таблице Продажи.Обороты. Полей Сум- маОборот и СуммаНДСОборот не достаточно, ведь нас интересует сум- ма продаж *на каждый день*. В параметрах виртуальной таблицы присут- ствует Периодичность. Выбрав подходящий вариант, мы обеспечим, что продажи будут собираться в разрезе этой периодичности. При Пе- риодичность = День в результат запроса будут попадать только те дни, за которые были продажи. С учетом сказанного запрос на выборку данных имеет вид:

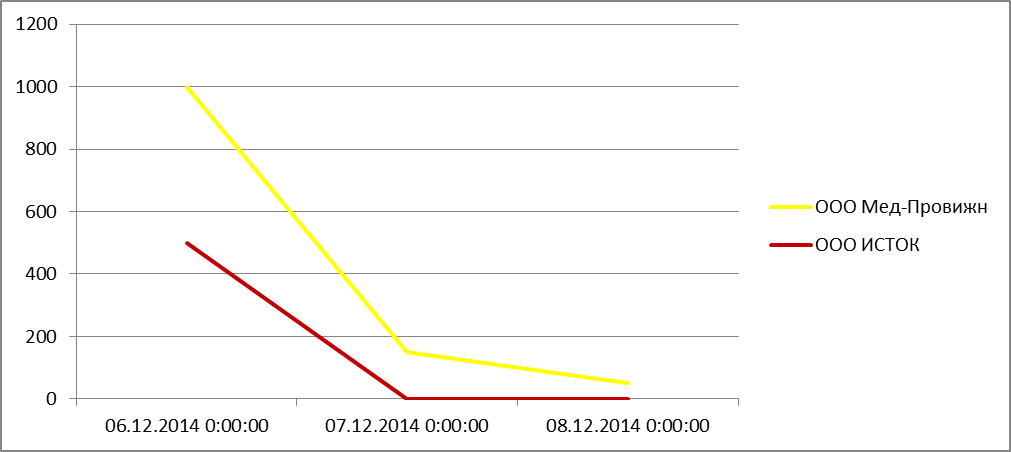
ВЫБРАТЬ

ПродажиОбороты.Контрагент, ПродажиОбороты.СуммаОборот + ПродажиОбороты.СуммаНДСОборот КАК Всего, ПродажиОбороты.Период

ИЗ

РегистрНакопления.Продажи.Обороты(, , День, )

КАК ПродажиОбороты

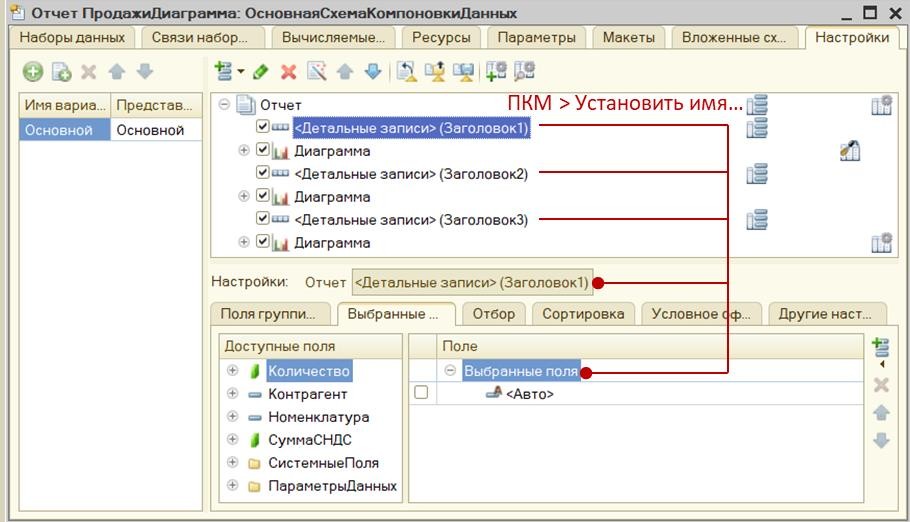
Настроенный отчет имеет следующий вид:

**Рис. 98 – Диаграмма типа «График с накоплением», отражающая суммы продаж**

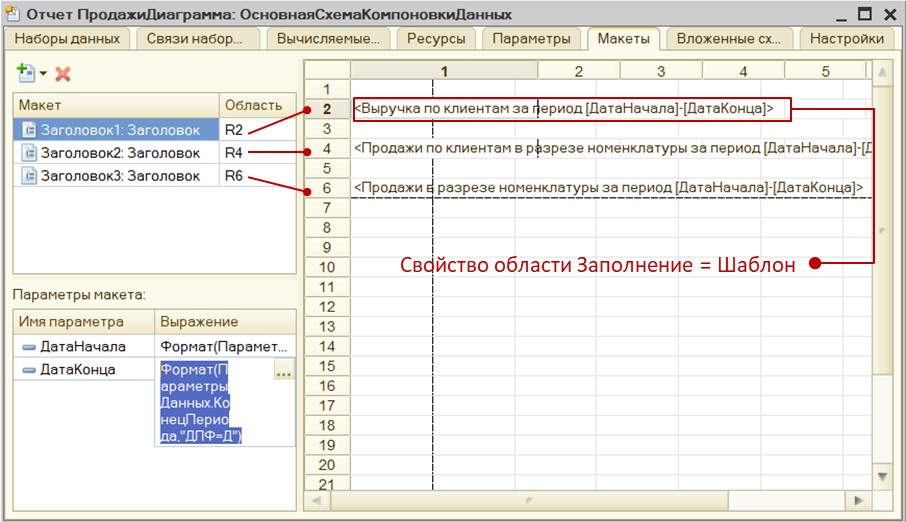
## Построение отчетов произвольной структуры

По умолчанию итоговая структура отчета определяется деревом элементов (группировки, таблицы, диаграммы, вложенные отчеты) и их параметрами с вкладки Настройки конструктора схемы компоновки

данных. Если же необходимо создать вариант отчета с нестрандарт- ным оформлением, пользуются Макетами в конструкторе СКД. Рас- смотрим алгоритм работы с макетами на примере задачи создания за- головков с параметрами для диаграмм.

Во-первых, в отчете необходимо определить новые группировки, установить для них произвольные имена, и для каждой группировки определить состав выбранных полей (см. рисунок 99).

**Рис. 99 – Настройка группировок для макета СКД**



**Рис. 100 – Макеты группировок с параметрами**

Далее в конструкторе СКД на вкладке Макеты создаются макеты группировки с типом макета Заголовок и задаются области для разме- щения заголовков. Области (ячейки) можно настраивать, например, изменять параметр Заполнение. Если в качестве варианта заполнения

выбрано Шаблон или Параметр, потребуется задать параметры макета в одноименной области. Так, на рисунке 100 каждый макет группировки имеет по два параметра – ДатаНачала и ДатаКонца. В качестве выраже- ния для заполнения параметров используются инструкции:

Формат(ПараметрыДанных.НачалоПериода,"ДЛФ=Д") Формат(ПараметрыДанных.КонецПериода,"ДЛФ=Д")

## Использование запросов в алгоритмах

Общая схема работы с запросом из встроенного языка:

Запрос = Новый Запрос;

// с помощью конструктора Новый создаем объект встроенного

// языка типа Запрос.

// тип данных Запрос предназначен для создания статических

// запросов, тип ПостроительОтчета – для динамических. Запрос.Текст = "ВЫБРАТЬ…";

// через свойство Текст указываем текст запроса РезультатЗапроса = Запрос.Выполнить();

// запрос выполняется с помощью метода Выполнить(),

// на выходе получаем тип данных РезультатЗапроса.

// РезультатЗапроса – таблица.

Если НЕ РезультатЗапроса.Пустой() Тогда

// проверяем, является ли результат пустой таблицей.

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

// с помощью метода Выбрать() курсор устанавливается на

// первую строку результат запроса.

// на выходе получаем тип данных ВыборкаИзРезультатаЗапроса.

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

// с помощью метода Следующий() последовательно обходим все

// строки результата в цикле Пока

Типовой код по обработке результата запроса можно сформиро- вать, вызвав из контекстного меню программного модуля конструктор запроса с обработкой результата.

Используем механизм запросов для того, чтобы контролировать достаточность товара на складе при реализации товаров. Потребуется модифицировать процедуру ОбработкаПроведения из модуля объекта документа РеализацияТовароУслуг. Схема решения:

* записать данные в регистры;
* выполнить запрос на поиск отрицательных остатков в регистре накопления ТоварыНаСкладах;
* если отрицательные остатки найдены, документ не должен быть про- веден; необходимо выдавать сообщение пользователю, какого товара и в каком количестве не хватает.

Процедура ОбработкаПроведения имеет два параметра: Отказ (ис- пользуя этот параметр, можем отказаться от проведения документа) и

Режим (параметр показывает, как документ проводится: оперативно – т.е. проведение осуществляется в реальном времени и дата документа устанавливается равной текущему времени, или неоперативно – зад- ним числом). Идеальная система учета – та, в которой все документы проводятся в оперативном режиме. Будем вести контроль остатков только для оперативно проводимых документов.

В штатном режиме движения попадают в регистр по окончанию исполнения обработчика ОбработкаПроведения (неявные штатные действия платформы). Для решения задачи контроля остатков товаров по предложенной схеме необходимо принудительно записать движе- ния до окончания исполнения обработчика.

Метод Записать() коллекции Движения вызовет принудительную за- пись тех наборов записей, для которых свойство Записывать установле- но в Истина.

Модифицированная процедура ОбработкаПроведения имеет следу- ющий вид:

Процедура ОбработкаПроведения(Отказ, Режим)

//{{ КОНСТРУКТОР\_ДВИЖЕНИЙ\_РЕГИСТРОВ

// формирование коллекции Движения для регистров

// ТоварыНаСкладах, Продажи, Взаиморасчеты

//}} КОНСТРУКТОР\_ДВИЖЕНИЙ\_РЕГИСТРОВ

// принудительная запись движений в регистры Движения.Записать();

// контролируем остатки, только если документ проводится

// в оперативном режиме

Если Режим = РежимПроведенияДокумента.Оперативный Тогда

// пишем текст запроса на поиск отрицательных остатков

// остатки будем получать на момент посл. операции в системе Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ … ";

// устанавливаем параметры запроса

// через стандартную конструкцию ЭтотОбъект обращаемся к

// реквизитам документа и его табличной части Товары Запрос.УстановитьПараметр("Склад", ЭтотОбъект.Склад); Запрос.УстановитьПараметр("МассивТоваров",

ЭтотОбъект.Товары.ВыгрузитьКолонку("Товар"));

РезультатЗапроса = Запрос.Выполнить(); Если НЕ РезультатЗапроса.Пустой() Тогда

Отказ = Истина; // отказываемся от проведения документа ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать(); Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

// выводим пользователю сообщение: сколько и каких

// товаров не хватает на складе Сообщить("Недостаточно товара …");

КонецЦикла;

КонецЕсли;

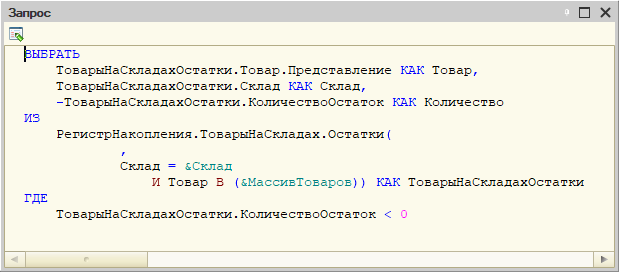
КонецЕсли;

КонецПроцедуры

Рекомендации по созданию запроса, находящего отрицательные остатки в регистре накопления ТоварыНаСкладах:

* используйте конструктор запроса с обработкой результата, тип обра- ботки – обход результата;
* источник данных – виртуальная таблица Остатки регистра накопления

ТоварыНаСкладах;

* достаточно трех полей выборки – Товар, Склад, КоличествоОстаток;
* используйте параметр Условие виртуальной таблицы, чтобы нало- жить ограничение на измерения регистра; через параметр Условие нужно получить список товаров табличной части проводимого доку- мента, чтобы проанализировать, остатки каких товаров и с какого склада следует проверить; если условие по товарам и складам разме- стить в секции ГДЕ, то сначала остатки будут рассчитаны по всем то- варам и складам, которые зафиксированы в регистре – это не опти- мально;
* для запроса нужно определить условие КоличествоОстаток < 0, т.к. нас интересуют отрицательные остатки.

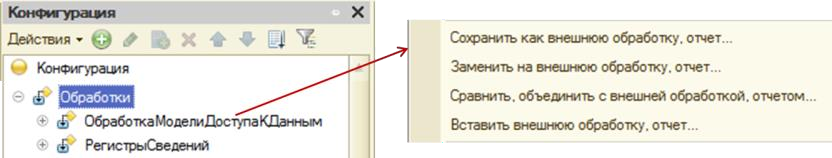
**Рис. 101 – Пример запроса на получение отрицательных остатков товаров**

## Обработки

Для выполнения различных действий над информацией в системе 1С:Предприятие используются объекты конфигурации, называемые Обработка. Концептуальным отличием отчетов и обработок является то, что в случае работы последних информация, хранящаяся в системе, претерпевает какие-либо изменения.

Обработка – прикладной объект конфигурации, предназначенный для реализации различных обработок информации и сервисных функ- ций.

Обработки содержат: алгоритмы обработки и получения информа- ции; формы для ввода параметров работы алгоритмов и представления результатов; макеты для вывода результатов выполнения алгоритма в табличный документ.

Обработки бывают внешние (\*.epf) и входящие в состав конфигура- ции. Любую обработку можно преобразовать во внешнюю, и добавить внешнюю обработку в состав конфигурации.

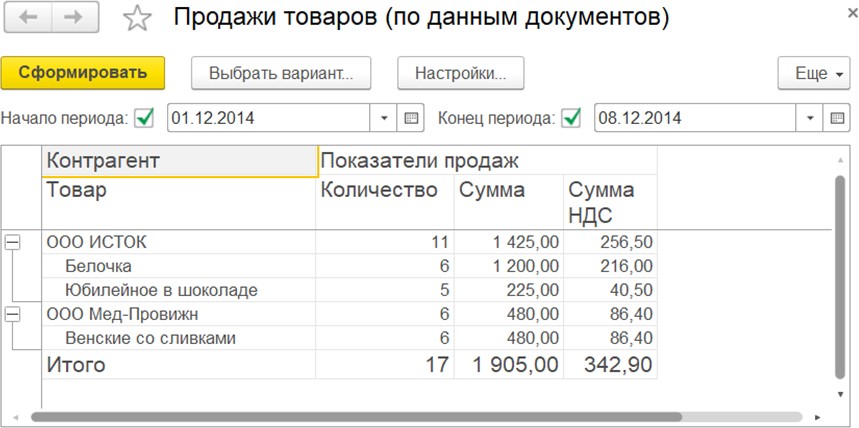
**Рис. 102 – Команды контекстного меню для преобразования обработок во внешние файлы и наоборот**

Для обеспечения целостности конфигурации внешние обработки рекомендуется использовать, в основном, в отладочных целях. После отладки алгоритма обработки необходимо включить в состав конфигу- рации.

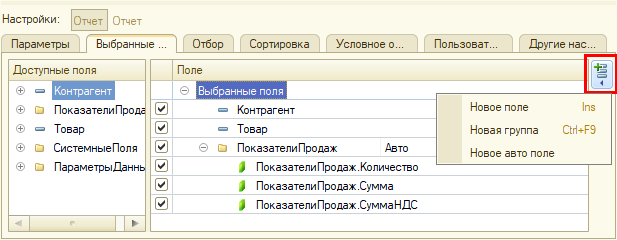
Примеры использования обработок:

* загрузка/выгрузка данных для обмена с другими приложениями;
* регламентные операции над данными БД (например, удалить поме- ченные объекты, обновить полнотекстовый индекс …);
* групповые действия над экземплярами объектов (например, перене- сти элементы справочника из одной группы в другую …).

# Содержание работы

1. Добавьте в состав конфигурации новую подсистему (Имя = Отчеты, Синоним = Отчеты по организации), которая будет предостав- лять пользователю доступ ко всем объектам конфигурации типа Отчет.
2. Выполните пример по созданию отчета ПоставщикиНоменкла- туры (с двумя вариантами представления данных).
3. Создайте отчет с именем ПродажиТоваровПоДокументам. Отчет отражает данные о продажах товаров (в количественном и суммовом выражении) в разрезе клиентов и номенклатурных позиций за задан- ный пользователем период. Обратите внимание, что в шапке отчета поля схемы компоновки Количество, Сумма и Сумма НДС объединены в группу Показатели продаж.

**Рис. 103 – Форма отчета Продажи товаров**



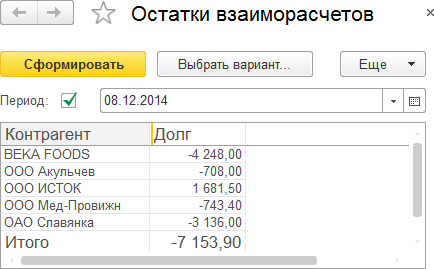
**Рис. 104 – Добавление группы полей в конструкторе СКД на вкладке Настройки**

Источник данных для построения отчета – табличные части докумен- тов РеализацияТоваровУслуг. Данные о продажах товаров из непрове- денных документов не должны попадать в отчет.

Границы периода, за который нужно сформировать отчет, передаются в запрос через параметры НачалоПериода и КонецПериода (тип Дата, состав Дата). Необходимо предусмотреть автоматическое приведение даты, заданной в поле КонецПериода, к концу дня.

Вариант отчета содержит группировки по клиенту и номенклатуре, а также итоги по каждому клиенту и в целом по всему отчету.

1. Выполните пример по созданию отчета ОстаткиВзаиморасчетов.

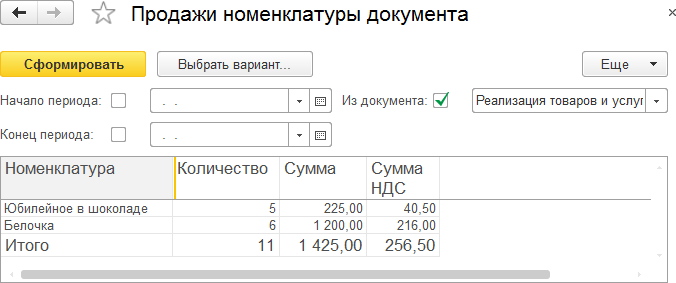


**Рис. 105 – Форма отчета Остатки взаиморасчетов**

1. Создайте отчет с именем ПродажиТоваров. В п.3 содержания работы для получения данных о продажах были использованы таблич- ные части документов. Реализуйте задание более корректным спосо- бом (на основе виртуальной таблицы соответствующего регистра накопления). Форма отчета аналогична рис. 103.

**Совет**. На результат запроса наложите условие (через параметры вир- туальной таблицы): в отчете не должны отражаться суммы за оказание услуг. Все числовые поля выборки переведите в ресурсы. Предусмот- рите возможность получения в отчете оборотов продаж за определен- ный период (используйте параметры НачалоПериода и КонецПериода виртуальной таблицы Обороты). Тип отчета – список, по полю Контр- агент производится группировка данных.

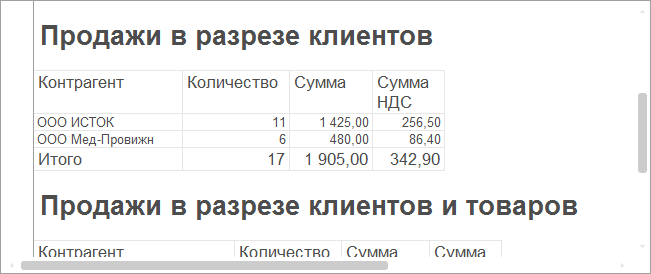
1. Создайте отчет с именем ПродажиНоменклатурыДокумента, реа- лизующий вывод информация о продажах (в количественном и суммо- вом выражении) тех товаров, которые содержатся в табличной части документа РеализацияТоваровУслуг, указываемого в качестве параметра в форме отчета. Используйте подзапрос в параметрах виртуальной таблицы (рис. 88, 89).



**Рис. 106 – Форма отчета Продажи номенклатуры документа**

1. Добавьте в состав конфигурации новый отчет копированием отчета, созданного по п.5. Имя нового объекта – СводныйОтчетОПрода- жахТовара. Отчет содержит один вариант с тремя способами представ- ления информации о продажах: 1) в виде таблицы, по строкам – това- ры, по колонкам – контрагенты; 2) в виде линейного списка, по клиен- там и без разделения на номенклатуру; 3) в виде списка, по клиенту и в разрезе номенклатуры.

**Рис. 107 – Форма отчета Сводный отчет о продажах товаров (часть 1)**

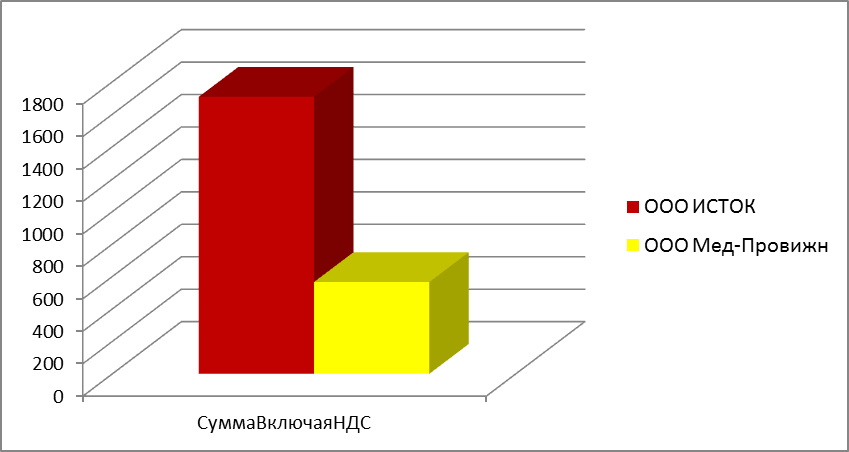


**Рис. 108 – Форма отчета Сводный отчет о продажах товаров (часть 2)**



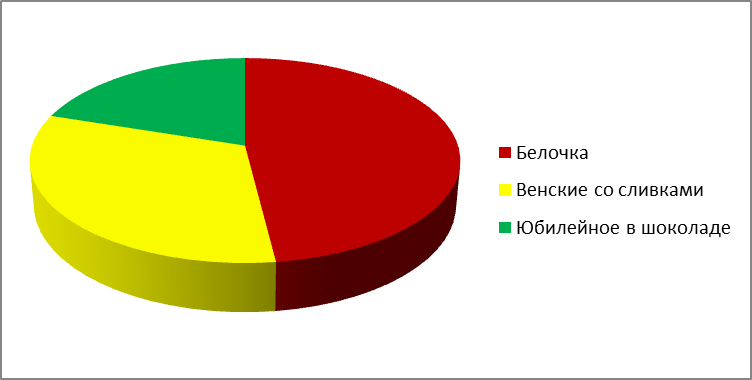
**Рис. 109 – Форма отчета Сводный отчет о продажах товаров (часть 3)**

Обратите внимание, что каждая часть отчета содержит свой заголовок. Заголовки можно задать в конструкторе схемы компоновки данных на вкладке Настройки (список параметров Другие настройки).

1. Создайте новый отчет ПродажиДиаграмма. Источник данных – виртуальная таблица оборотного регистра накопления Продажи. Поля выборки: Контрагент, Номенклатура, СуммаВключаяНДС (вычисляемое поле, получаемое как сумма полей СуммаОборот и СуммаНДСОборот). Отчет предполагает один вариант представления данных, включаю- щий три диаграммы, которые позволят анализировать данные о про- дажах в различных разрезах (см. рисунки 110-112). Каждая диаграмма сопровождается разными заголовками с параметрами: 1) Выручка по клиентам за период [НачалоПериода]-[КонецПериода]; 2) Продажи по клиентам в разрезе номенклатуры за период [НачалоПериода]- [КонецПериода]; 3) Продажи в разрезе номенклатуры за период [Нача- лоПериода]-[КонецПериода].

**Рис. 110 – Объемная тистограмма, показывающая сумму выручки по клиентам**



**Рис. 111 – Объемная тистограмма (продажи по клиентам в разрезе номенклатуры)**

**Рис. 112 – Круговая диаграмма (продажи в разрезе номенклатуры)**

1. Реализуйте отчет ГрафикПродаж по примеру (рис. 98). Отчет отражает график продаж по дням в разрезе клиентов.
2. Создайте отчет ОстаткиТоваровНаСкладах. Отчет отражает остатки товаров в количественном и суммовом (включая сумму НДС) выражении в разрезе складов. Пользователь должен иметь возмож- ность указать период, на который формируется отчет.
3. Решите задачу контроля остатков товаров: при проведении до- кумента РеализацияТоваровУслуг необходимо контролировать доста- точность товара на складе. Перепроведите все документы. Заново сформируйте отчет по остаткам товаров.
4. Расположите в командном интерфейсе созданные отчеты в по- рядке их следования в содержании работы.
5. Ознакомьтесь с инструментами пользовательского режима, которые позволяют настраивать отчеты (сортировка, отборы, условное оформление, изменение типа диаграммы). В отчет занести примеры настройки форм отчетов из пользовательского режима.
6. Для отчета о продажах товаров, созданного в п.5, добавьте в схему компоновки новый параметр – ЕдиницаИзмерения. С помощью условного оформления настройте выбор представления данных в столбцах Сумма и Сумма НДС (рубли, копейки; рубли; тыс. руб.). В ка- честве примера используйте статью [https://infostart.ru/public/880463/.](https://infostart.ru/public/880463/)

# Контрольные вопросы

1. Охарактеризуйте возможности табличной модели доступа к дан- ным, принятой в системе 1С:Предприятие 8.
2. Какова структура запроса?
3. Перечислите возможности конструктора схемы компоновки дан- ных.
4. Перечислите возможности конструктора запроса.
5. Перечислите возможности конструктора запроса с обработкой ре- зультата.
6. В чем отличие системы компоновки данных и схемы компоновки данных?
7. Какие таблицы могут выступать в качестве источника данных для запроса?
8. Для чего используется конструктор настроек компоновки данных?
9. Для чего используется конструктор произвольных выражений?
10. Из каких соображений в рамках одного отчета создаются несколько вариантов этого отчета?
11. В чем удобство использования регистров на этапе получения дан- ных для отчетов?
12. Какие виртуальные таблицы регистров можно использовать для построения отчетов? Приведите примеры.
13. В чем важность использования параметров виртуальных таблиц?
14. Какие способы вывода информации в отчет поддерживает СКД?
15. Опишите алгоритм использования Макетов в конструкторе СКД.
16. Опишите схему решения задачи по контролю остатков товаров на складе.
17. Приведите примеры использования обработок в системе 1С:Предприятие.

# Библиографический список

1. Информационно-аналитический центр по автоматизации учета и управления Infostart Journal [Электронный ресурс]. – Режим до- ступа: https://infostart.ru/ (свободный).
2. Профессиональные курсы для специалистов по внедрению 1С. Бесплатные материалы [Электронный ресурс]. – Режим доступа: https://курсы-по-1с.рф/free/ (свободный).
3. Радченко, М.Г. 1С:Предприятие 8.3. Практическое пособие разра- ботчика. Примеры и типовые приемы / М.Г. Радченко, Е.Ю. Хру- сталева. - М.: ООО «1С-Паблишинг», 2013. - 965 с.: ил.
4. Введение в конфигурирование в системе 1С:Предприятие 8. Ос- новные объекты. Методические материалы для слушателя серти- фицированного курса. – М.: ООО «1С-Учебный центр №3», 2012.

– 111 с.

1. Конфигурирование в системе 1С:Предприятие 8. Основные объ- екты. Методические материалы для преподавателя сертифициро- ванного курса – М.: ООО «1С-Учебный центр №3», 2010. – 35 с.