# Общая постановка задачи.

**Описание функционирования организации.**

Условная фирма оказывает услуги клиентам.

*Учетная политика*

Для учета затрат в фирме предлагается следующее:

Услуги детализированы по видам работ.

По каждому виду работ установлены нормативы расхода материалов, на основании которых планируется вести учет затрат по соответствующей статье прямых расходов.

По каждому виду работ установлены нормативы расхода трудовых ресурсов в разрезе должностей.

Утвержден перечень статей калькуляции.

*При обслуживании клиентов*

С клиентами заключаются договора на оказание услуг, на основании которых и производятся соответствующие работы. Также при заключении договора оговаривается, чьи материалы будут использоваться при оказании услуг (выполнении работ). Возможно использование как материалов фирмы, так и материалов заказчика. Соответствующие позиции прописываются в договоре. Выполнение работ закрывается актом, причем по одному договору может быть закрыто несколько актов. В акте указываются закрытые позиции договора, использованные материалы (если оговорено использование материалов исполнителя). Оплата производится клиентом на основании акта о выполнении работ). Акт и документ об оплате проводятся через бухгалтерию.

Бухгалтерский учет ведется на основании стандартного плана счетов.

Списание материалов производится на основании актов об оказании услуг в соответствии с расшифровкой по материалам. Списание отражается в бухгалтерском учете.

*Дополнительная характеристика*

Предприятие является самостоятельной юридической единицей и имеет собственный расчетный счет.

**Требования к информационной системе**

В среде 1С необходимо разработать информационную систему, поддерживающую следующие функции:

1. Ведение НСИ;
2. Ввод и накопление оперативных данных;
3. Обеспечение комфортного интерфейса пользователя при работе с документами: формы документов для ввода и редактирования данных, печать документов, автоматизированное проведение документов в соответствии с правилами бухгалтерского учета;
4. Формирование отчетов в соответствии с потребностями пользователей в анализе;
5. Расчет заработной платы сотрудников;
6. Интеграция данных с MS Excel в двух направлениях – передача данных в информационную систему, выгрузка данных.

**Информационные потребности пользователей**

*Руководител*ь: доступ ко всей отчетности, в том числе в среде MS Excel.

*Бухгалтер*: доступ к справочникам и документам для ввода и редактирования данных; проведение документов, доступ к бухгалтерским и аналитическим регистрам; формирование и доступ к отчетам; расчет заработной платы, получение соответствующих документов.

*Отдел закупок*: поддержка планирования закупок, доступ к отчетам по анализу расхода материалов.

*Краткая характеристика модели*

**2**

Фрагмент 1 модели данных представляет традиционную реляционную модель данных, все сущности нормализованы, связи один-ко-многим.

Фрагмент2. Представляет структуры данных:

* Обеспечивающих проведение хозяйственных операций (виды субконто, план счетов, бухгалтерский регистр). *Виды субконто* представляют конкретные информационные объекты (структуры данных), такие как Договор, Контрагент, Материалы, Услуги. Перечень видов субконто определяется учетной политикой организации. Связь устанавливается через ссылку на конкретные объекты (СсылкаТипДанных). Виды субконто обеспечивают аналитический разрез при проведении документов. Наполнение бухгалтерского регистра производится автоматически при проведении документов (не вручную) в соответствии с заданными алгоритмами.
* Структуры *Прайс* и *Учетные* *цены* имеют следующую специфику: привязка реквизита –основания (цены) не только к коду объекта (услуги, материалу), но и дате. По-другому говоря, требуется иметь доступ к определенной цене на ближайшую дату к заданной дате(не обязательно текущей). Кроме того, учетные цены не задаются пользователем, а рассчитываются по заданному алгоритмы в соответствии с учетной политикой организации.

В модели не представлены ряд структур данных, связанных с расчетом заработной платы. Их создание будет особо оговариваться при разработке системы.

# Программная реализация информационной системы в среде 1С:Предприятие версии 8

Предлагается следующая технология реализации:

1. Создание структур данных (справочников, документов), разработка форм документов, формирование простых процедур заполнения форм.
2. Создание регистров сведений, формирование процедур заполнения форм документов на основании данных регистров сведений. Разработка процедур заполнения регистров сведений.
3. Формирование регистров накопления, разработка простых отчетов.
4. Разработка обработки «Контроль расхода материалов», формирование отчета для вывода результатов обработки.
5. Создание дополнительных отчетов.
6. Разработка регистров расчета для начисления заработной платы.
7. Разработка механизма интеграции с MS Excel.
8. Администрирование информационной системы.

Создание информационной системы осуществляется в режиме Конфигуратора 1С:Предприятие. Конфигурация (в режиме конфигуратор) представляет описание тех структур данных, которые пользователь может использовать. По-другому говоря, данные о структурах данных, или *метаданные*. Конкретное описание структуры назовем объектом конфигурации. Каждый объект конфигурации включает доступ к настройке всех свойств, установленных для данного объекта. К свойствам могут относится: структура «представителя» объекта (конкретного справочника, документа, регистра, обработки и т.д.), форма, разные модули для формирования процедур обработки, макеты, правила движения документа и др. в зависимости от типа объекта конфигурации.

Дерево конфигурации представлено на Рис. 1

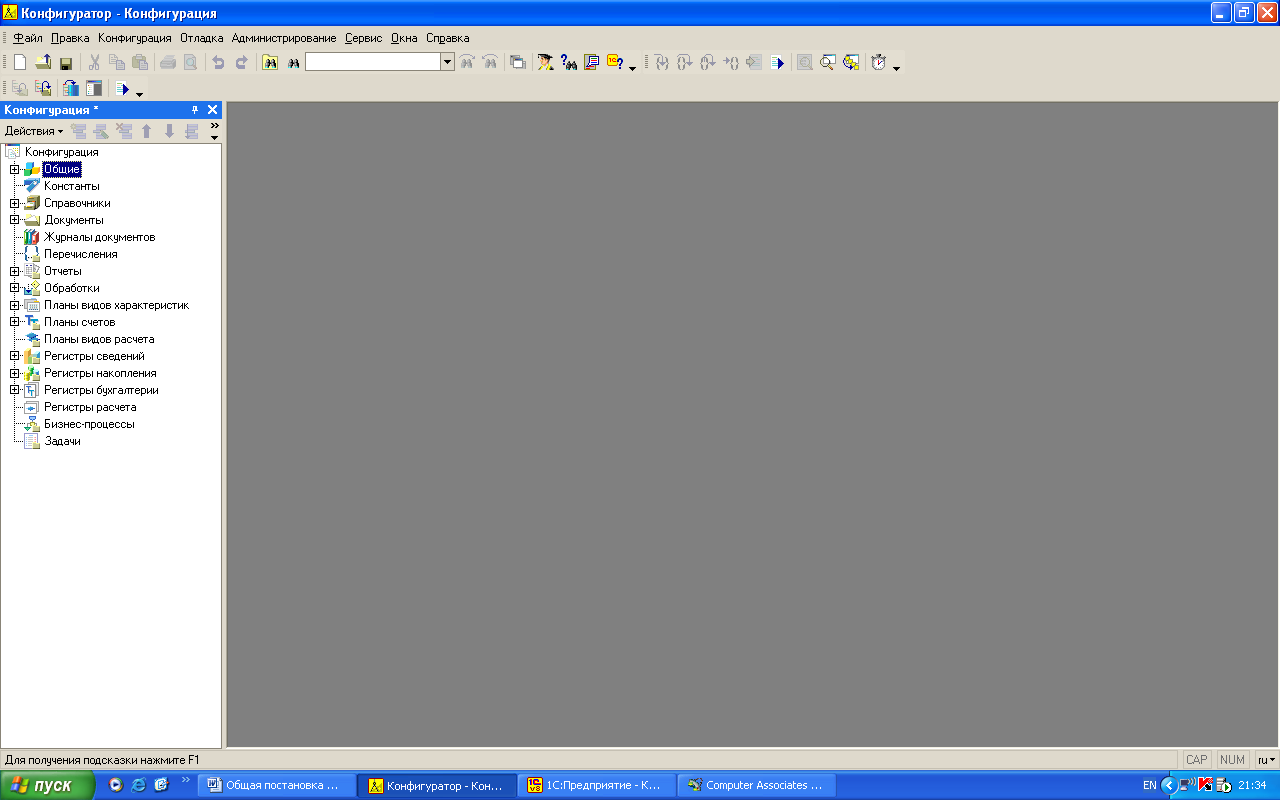


Рис.

Характеристика конкретных объектов будет даваться по мере разработки системы.

При первоначальном запуске программы создаем новую пустую конфигурацию, присваиваем ей соответствующее имя – *Услуги*.

## Создание справочников

Объект *Справочник* является инструментом для разработки тех сущностей (таблиц) модели данных, которые соответствуют справочникам в данной предметной области. Если для данной сущности в контексте предметной области определяется привязка значений (реквизитов-оснований) к дате (см. стр. 2), то лучше воспользоваться объектом *Регистр сведений*.

В соответствии с моделью данных к справочникам отнесем: *Контрагент*, *Материал*, *Нормативы*, *Услуга*.

**Справочник Контрагент**

Добавить новый справочник (соответствующая кнопка на панели инструментов дерева конфигурации или клавиша Insert).

Вкладка *Основные*. Имя – *Контрагент*.

Вкладка *Данные*. При создании справочника по умолчанию создаются два реквизита: код элемента, наименование. Свойства этих реквизитов можно увидеть и откорректировать на вкладке Данные. Изменять имена нельзя. Ключевым реквизитом в любом справочнике является реквизит *Код элемента*. В справочнике Контрагенты других полей (реквизитов) не определено, поэтому создание структуры справочника завершено.

Вкладка *Формы*. Проверяем: редактирование – в списке (не будет выводиться форма отдельного элемента).

**Справочник Материал**

Вкладка *Данные*. Добавляем реквизит *Единица* *Измерения*. В окне свойств: Тип данных – строка, длина 11.

Вкладка *Иерархия*. По специфике предметной области элементы данного справочника требуется группировать. В частности, это необходимо для связи с нормативами расхода материалов. Дело в том, что нормативы ссылаются не на конкретные материалы, необходимы для данной услуги, а на группу взаимозаменяемых материалов. На указанной вкладке активизируем свойство *Иерархический справочник*. Вид иерархии – иерархия элементов. В этом случае родителем будет выступать не папка (группа), а непосредственно один из элементов справочника. Такой вид иерархии упрощает процедуры при дальнейшем использовании справочника. Ограничим количество уровней иерархии – 2.

Вкладка *Формы*. Создать форму списка и форму выбора справочника. В разделе *Основные* *формы* в строке Списка нажать на «лупу» и запустить конструктор форм. На втором шаге в конструкторе справочника раскрыть иерархию *Список*, задать (проверить) вывод полей *Код*, *Наименование*, *Родитель*. Закончить создание формы, просмотреть результат. Аналогично создать форму выбора.

Справочники **Услуга, Сотрудники, Вид Работ** создать самостоятельно.

**Справочник НормативыМатериалов.**

Задать имя.

Вкладка *Данные*. Добавить реквизиты *ДатаФормирования*, тип – дата, *ВидРабот*. Для реквизита *ВидРабот* задается тип данных – *СправочникСсылкаВидРабот*. Таким образом в системе 1С:Предприятие устанавливаются связи между таблицами (объектами). Причем в представлении для пользователя ссылка идет не на конкретное поле справочника услуг (хотя связь устанавливается естественно по коду элемента услуга), а на все поля (реквизиты) выбранной услуги. Что позволяет в дальнейшем в процедурах без дополнительных обработок выходить на все значения реквизитов услуг.

На этой же вкладке создать новую табличную часть *СпецификацияНорматива*. Для спецификации задать реквизиты *Материал* (тип данных - *СправочникСсылка.Материал*), *количество*( тип данных – число). По умолчанию при создании табличной части справочника создается реквизит *НомерСтрокиТабличнойЧасти*. Связь с таблицей справочника Нормативы в этом случае устанавливается автоматически по полю *КодЭлемента* соответствующего справочника. Ключом для таблицы *СпецификацияНорматива* явлются: *НомерСтрокиТабличнойЧасти, КодЭлемента*.

Вкладка *Формы*. В окне Редактирование поставить переключатель Обоими способами. С помощью конструктора создать форму элемента.

После создания справочника запустить режим приложения (1С:Предприятие) – Кнопка *Начать отладку* на панели инструментов (F5). При запуска конфигурацию сохранить. Так как еще не решались задачи, связанные с разработкой интерфейсов пользователей (и с администрированием пользователей), в режиме приложения нет достойного меню. Поэтому просмотр всех созданных объектом возможен через меню *Операции/Справочники*. Открыть справочники, внести данные.

**Справочник НормативыТруда.**

Данный справочник предназначен для хранения информации об установленных нормативах затрат трудовых ресурсов (основных рабочих) при выполнении определенного вида работ.

Реквизиты основной части справочника: *ВидРабот* (тип данных — *СправочникСсылкаВидРабот*).

Имя табличной части: *Спецификация*. Реквизиты табличной части: Должность (тип данных - *СправочникСсылка.Должность), ЧеловекоЧасы (*тип данных *– число).*

Вкладка *Формы*. В окне Редактирование поставить переключатель Обоими способами. С помощью конструктора создать форму элемента.

**Справочник Статьи калькуляции**

## Создание документов

Объект Документ инструментом для разработки тех сущностей (таблиц) модели данных, которые соответствуют документам предметной области и содержат информацию о свершенных хозяйственных операциях или о событиях в жизни организации. Особенностью объекта Документ является то, что он обладает способностью проведения. Факт проведения означает, что событие, которое он отражает, повлияло на состояние учета. По-другому говоря, при проведении документа создается запись в одном или нескольких регистрах (бухгалтерских, регистрах накопления).

В соответствии с моделью данных к документам отнесем: *Поставка материалов, Договор на оказание услуг, Акт на оказание услуг*. Впоследствии будут добавлены ряд документов, связанных с расчетом заработной платы.

**Документ Поставка материалов.**

Добавить новый документ. Задать имя документа.

Вкладка *Данные*. При создании документа по умолчанию создаются два реквизита: номер документа, дата создания документа. Создать реквизиты: *Контрагент, ИтогоБезНДС, ИтогоПоДокументу*.

Создать табличную часть *СпецификацияПоставки*, включающую следующие реквизиты: *Материал, ЦенаПоставки, Количество, Сумма, НДС, СуммаСНДС*. По умолчанию при создании табличной части документа создается реквизит *НомерСтрокиТабличнойЧасти*. Связь с таблицей документа в этом случае устанавливается автоматически по полю *НомерДокумента* соответствующего документа. Ключом для таблицы *СпецификацияПоставки* явлются: *НомерСтрокиТабличнойЧасти, НомерДокумента*.

Вкладка *Движение*. Проверить следующие позиции: Проведение – разрешить, Оперативное проведение – разрешить. С оперативным проведением связано понятие оперативной отметки времени. Оперативная отметка (дата проведения) создается (указывается в соответствующем регистре) каждый раз при проведении документа. Если проходит повторное проведение документа (или проведение проходит позже, чем дата создания документа), то при оперативном проведении дата проведения устанавливается соответствующая текущей. Если проведение неоперативное, то дата проведения соответствует дате предыдущего проведения или дате документа. Движения документа будут созданы позднее.

Вкладка *Формы*. Создать форму документа. В строке Основные формы документа нажать на «лупу» и запустить конструктор форм. Просмотреть результат.

Для пользователя естественно, чтобы каждый раз при изменении значений в поле Количество или поле *ЦенаПоставки* автоматически рассчитывались значения полей *Сумма, НДС, СуммаСНДС*, а также итоговые поля документа. Для этого необходимо сформировать процедуру в модуле формы и привязать ее запуск к некоторому событию, связанному с изменением полей формы.

*Модуль* – место «хранения» процедур на встроенном языке. В системе 1С есть несколько типов модулей, привязанных к соответствующим объектам или конфигурации в целом. Для расчета полей формы воспользуемся модулем формы, где «собраны» процедуры, привязанные к различным объектам формы.

Запуск требуемой процедуры будет производиться в момент события изменения значения объекта формы Поле ввода *Количества*. Для соответствующего поля в форме вызвать окно свойств (двойной щелчок по полю ввода *Количество*), в списке свойств в разделе События найти событие *При изменении*, щелкнуть по лупе. В модуле формы появляется «заготовочка» процедуры обработки *СпецификацияПоставки КоличествоПриИзменении*. Сформируем текст процедуры:

|  |  |
| --- | --- |
| *Процедура СпецификацияПоставкиКоличествоПриИзменении(Элемент)* | Начало процедуры |
| *СТЧ=ЭлементыФормы.СпецификацияПоставки.ТекущиеДанные;* | Обращение к программному объекту *ЭлементыФормы*, который включает все элементы формы. Обращаемся к элементу формы *СпецификацияПоставки* (табличной части, всем строкам). Так как необходим расчет для текущей строки, вызываем свойство *ТекущиеДанные*. Ссылку на текущую строку присваиваем переменной *СТЧ*. |
| *СТЧ.Сумма=СТЧ.Количество\*СТЧ.ЦенаПоставки;* | Переменная *СТЧ* представляет собой строку, включающую все значения полей табличной части формы. Ссылка на конкретное значение осуществляется через обращение к имени поля. В представленных строках процедуры даны формулы расчета значений полей текущей строки. |
| *СТЧ.НДС=СТЧ.Сумма\*0.18;* |
| *СТЧ.СуммаСНдс=СТЧ.Сумма+ СТЧ.НДС;* |
| *ИтогоБезНДС=СпецификацияПоставки.Итог("Сумма");* | После расчета всех полей текущей строки необходимо выполнить пересчет итоговых полей. Здесь обращаемся к реквизитам шапки документа. Функция Итог суммирует значения всех строк в указанной колонке Спецификации поставки.. |
| *ИтогоПоДокументу=СпецификацияПоставки.Итог("СуммаСНдс");* |
| *КонецПроцедуры* |  |

Проверка синтаксиса процедуры: *Текст/Синтаксический контроль*.

Обратить внимание, что в свойствах поля *Количество* в событии При изменении появился текст процедуры.

Эту же процедуру поставим в поле При изменении (События) для поля ЦенаПоставки. (просто выбрать процедуру из списка.

Модуль документа и ряд других свойств документа сформируем позже.

**Документ Договор на оказание услуг**

Добавить новый документ. Задать имя документа.

Вкладка *Данные*. Создать реквизиты: *Контрагент, ИтогоБезНДС, ИтогоПоДокументу, Сотрудник*.

Создать табличную часть *СпецификацияДоговора*, включающую следующие реквизиты: *Услуга, Цена, Количество, Сумма, НДС, СуммаСНДС*.

Вкладка *Формы*. Создать форму документа. Поле Цена заполняется на основании прайса, цены на услуги не постоянны, меняются в зависимости от дат. Поэтому, процедура заполнения полей формы будет разработана позже, после знакомства с регистрами сведений.

**Документ Акт об оказании услуг**

Вкладка *Данные*. Реквизиты: *Договор* (тип данных ДокументСсылка.Договор НаОказаниеУслуг)*, Контрагент, ИтогоПоУслугам, ИтогоПоМатериалам, ИтогоБезНДС, ИтогоСНДС*.

Создать табличные части:

* *СпецификацияУслуг* (*Услуга, Цена, Количество, сумма, НДС, СуммаСНДС*)
* *СпецификацияПоМатериалам* *(Материал, Цена, Количество, Сумма, НДС, СуммаСНДС*)

Вкладка *Формы*. Создать форму документа. Поле *Цена* в спецификации акта по услугам будет заполняться на основании цены, определенной в договоре. Процедура будет сформирована позже.

На основании договора может быть закрыто несколько актов, поэтому при заполнении каждого акта желательно автоматическое формирование поля *Количеств*о в соответствии с «остатками» услуг по договору, также необходимо контролировать правильность заполнения этого поля (чтобы не превысить договорное количество). Соответствующие процедуры также будут сформированы несколько позже.

Наценка на цену продажи клиенту материалов при оказании услуг определяется в 25% от учетной цены. Так как еще не сформирован регистр учетных цен, то этот расчет сейчас сделать нельзя.

Запустить приложение 1С:Предприятие, просмотреть документы.

**Документ Наряд на выполненные работы**

Данным документом фиксируются выполненные сотрудниками работы, для которых установлена сдельная оплата труда. Учет ведется в разрезе договора, вида работ, исполнителя (сотрудника – основного рабочего).

Реквизиты основной части документа: Договор (тип данных *ДокументыСсылка.Договор*)

Имя табличной части – *Спецификация*. Реквизиты табличной части: ВидРабот (тип данных – *СправочникСсылкаВидРабот), Сотрудник (*тип данных *– СправочникСсылкаСотрудник), Отработано (*тип данных *– число), Расценка, Сумма (*число*).*

## Регистры сведений

Объект конфигурации регистр сведений предназначен для описания структур хранения данных в разрезе нескольких измерений. Каждая запись регистра сведений устанавливает новое значение ресурса (реквизита-основания в контексте реквизитов-признаков) Важной особенностью регистра сведений является его способность хранить сведения с привязкой ко времени. Таким образом, возможно, хранить не только текущие данные, но и «историю» их изменения во времени. С объектом регистр сведений связаны определенные методы управления и доступа к данным. Периодический регистр сведений (с привязкой к дате) всегда содержит поле период (дата). Ключом записи является совокупность значений измерений регистра и периода.

В разрабатываемой информационной системе предлагается два регистра сведений: *Прайс* (цены на услуги с привязкой к дате), заполняется пользователе и является основанием для установления цены услуги на конкретную дату; Учетные цены (цена учета материалов), заполняется автоматических по требованию пользователя.

**Регистр сведений Прайс**

Добавить новый регистр сведений. Имя – *ПрайсУслуги*. Периодичность – в пределах секунды (дата с точностью до секунды), режим записи – независимый.

Вкладка *Данные*. Измерение – *Услуга* (тип данных – ссылка на справочник Услуги). Ресурс – *ЦенаПродажи*. В регистре будут храниться сведения о ценах продаж в разрезе услуг на указанный период. Под указанным периодом для цены на услугу понимаем период от даты конкретной записи для услуги до даты следующей (поздней) записи для той же услуги.

**Регистр сведений УчетныеЦены**

Создать новый регистр сведений, присвоить имя. Задать периодичность и режим записи.

Вкладка *Данные*. Измерение – *Материал*, ресурс – *ЦенаУчетная*. В регистре будут храниться сведения об учетных ценах в разрезе материалов на указанный период.

В фирме принят метод ФИФО для оценки запасов и формирования учетных цен. В соответствии с методом ФИФО (первый на приход – первый на списание) учетная цена материалов будет определяться по последней поставке. Конечно, подобный расчет не будет выполняться каждый день. Для простоты расчетов, считаем, что бухгалтер сам определяет дату (период расчета учетной цены). Наша задача – разработать соответствующую процедуру формирования записей в регистре сведений *УчетныеЦены* в соответствии с политикой организации. Данная процедура будет представлена несколько позже после знакомства с механизмом использования запросов в системе 1С:Предприятие.

Запустить приложение 1С:Предприятие, просмотреть регистры сведений, заполнить сведения о ценах на услуги (регистр сведений *Прайс*).

**Регистр сведений Расценки**

Добавить новый регистр сведений. Имя – *Расценки*. Периодичность – в пределах секунды (дата с точностью до секунды), режим записи – независимый.

Вкладка *Данные*. Измерение – *ВидРабот* (тип данных – ссылка на справочник ВидРабот). Ресурс – *Стоимость*. В регистре будут храниться сведения о расценках в разрезе видов работ на указанный период.

Прикладная задача: использование регистра сведений при заполнении документов

Вернемся к документу *Договор на оказание услуг*. Выше отмечалось, что для пользователя необходима автоматизация следующих операций при работе с договором:

* при выборе услуги – автоматический вывод последней цены по *Прайсу* (последней по отношению к дате документа *Договор*);
* при изменении значения количества – расчет полей Сумма, НДС, Сумма СНДС, а также итоговых полей документа.

Запуск требуемой процедуры будет производиться в момент события изменения значения объекта формы Поля - ввода *Услуга*. Для соответствующего поля в форме вызвать окно свойств (двойной щелчок по полю ввода *Услуга*), в списке свойств в разделе События найти событие *При изменении*, щелкнуть по лупе. В модуле формы появляется «заготовочка» процедуры обработки *СпецификацияУслугУслугаПриИзменении*. Сформируем текст процедуры:

|  |  |
| --- | --- |
| *Процедура СпецификацияУслугУслугаПриИзменении(Элемент)* |  |
| *СТЧ=ЭлементыФормы.СпецификацияУслуг.ТекущиеДанные;* | Ссылку на текущую строку присваиваем переменной *СТЧ* |
| *Отбор=Новый Структура("Услуга", СТЧ.Услуга);* | Создаем новую (временную) структуру данных, которая состоит из заголовка поля («Услуга»), значения поля (СТЧ.Услуга). Данной структуре присваиваем имя Отбор. Отбор – предопределенное имя для объекта Отбор, по которому осуществляется текущая фильтрация записей при считывании или записи набора. Внимание: имя поля («Услуга») должно точно соответствовать имени соответствующего измерения в регистре сведений Прайс. |
| *Значение=РегистрыСведений.ПрайсУслуги.ПолучитьПоследнее(Дата,Отбор)* | Обращаемся к регистру сведений *ПрайсУслуги* и используем метод *ПолучитьПоследнее*(*Дата*,*Отбор*), который возвращает нам значения ресурсов наиболее поздней записи регистра, которая соответствует дате договора (параметр *Дата*) и значению передаваемого измерения (*Отбор*). Возвращается соответствующая запись регистра, которой присваиваем имя *Значение*. |
| *СТЧ.ЦенаПродажи=Значение.ЦенаПродажи;* | Присваиваем значение *ЦеныПродажи* найденной записи соответствующему полю документа. |
| *КонецПроцедуры* |  |

Запустить приложение, ввести договор, проверить работу процедуры.

Процедуру расчета полей Сумма, НДС, Сумма СНДС, а также итоговых полей документа сформировать самостоятельно (см. процедуру на стр. 6).

## Регистры Накопления

Объект конфигурации регистр накопления предназначен для описания структур хранения данных, поставляемых другими объектами базы, чаще всего документами. Данные документов будут храниться в регистре в виде отдельных записей одинаковой структуры. Структура регистра задается разработчиком при создании регистра и включает также измерения и ресурсы. Для формирования записей регистра при проведении документов создается процедура движения. Каждому движению присваивается вид движения: приход (+) или расход (-). Каждая запись регистра привязывается к порождающему ее документу (регистратору).

При работе с запросами Регистр накопления представляется как виртуальная таблица, позволяющая отобразить не только показатели прихода и расхода ресурсов, но и остатки. При этом разработчику не требуется прописывать дополнительные процедуры итогов, что очень удобно при анализе информации, разработке отчетов и др.

Для на шей информационной системы регистры накопления необходимы для:

* контроля выполнения договоров;
* контроля запасов на складе;
* ряда других контролирующих и аналитических функций, в том числе учете затрат.

### Создание регистров накопления

На данном этапе разработаем 2 регистра накопления:

*Выполнение договора*. Позволяет накапливать информацию о выполнении обязательств перед контрагентами в разрезе конкретных договоров, услуг. Регистраторами являются: *Договор на оказание услуг* (вид движения – приход), *Акт об оказании услуг* (вид движения – расход). На основании записей регистра можно отслеживать ход выполнения обязательств, анализировать взаимоотношения с клиентами, проводить мониторинг «востребованности» услуг и др.

*Движение материалов*. Позволяет накапливать информацию о движении материалов: поступление, расход, запасы. Регистраторы: *Поставка материалов* (вид движения – приход), *Акт об оказании услуг* (вид движения – расход). На основании записей данного регистра можно отслеживать оборот материальных запасов, определять потребность в закупках, контролировать запасы.

**Регистр накопления *ВыполнениеДоговоров*.**

Создать новый регистр, присвоить имя. Вид регистра – остатки (в представлениях регистра при конструировании запросов будут автоматически рассчитываться остатки в заданных пользователем измерениях).

Вкладка *Данные*. Измерения: *Контрагент, Договор, Услуга, ЦенаПоДоговору*. Типы данных – ссылки на соответствующие справочники и документы. Ресурсы: *Количество, Сумма*.

Вкладка *Регистраторы*. Активизировать соответствующие регистраторы (*Договор об оказании услуг, Акт об оказании услуг*).

**Регистр накопления *ДвижениеМатериалов*.**

Создать новый регистр, присвоить имя. Вид регистра – остатки

Вкладка *Данные*. Измерения: *Материал*. Типы данных – ссылки на соответствующие справочники и документы. Ресурсы: *Количество, Сумма*.

Вкладка *Регистраторы*. Активизировать соответствующие регистраторы (*ПоставкаМатериалов, Акт об оказании услуг*).

### Разработка движений документов

Движения документа – записи в регистре, которые создаются при проведении документа и отражают изменения, производимые документом.

По регистру *ВыполнениеДоговоров* необходимо сформировать движения для документов *Договор на оказание услуг, Акт об оказании услуг*.

**Движения документа Договор на оказание услуг**

Открыть документ Договор, перейти на вкладку *Движения*. Активизировать конструктор движений, добавить новое движение по регистру *ВыполнениеДоговоров*.

Установить тип движения регистра – приход. Выбрать табличную часть *СпецификацияДоговора*.

Для ускорения заполнения выражений (для полей регистра) нажать кнопку *ЗаполнитьВыражения*. Проверить заполнение. В качестве значения ресурса *Сумма* будем считать значение суммы без НДС[[1]](#footnote-1).

Для поля *Договор* устанавливаем выражение *Ссылка* (прописывается вручную).

Нажать ОК и просмотреть сформированную конструктором процедуру:

|  |  |
| --- | --- |
| *Для Каждого ТекСтрокаСпецификацияУслуг Из СпецификацияДоговора Цикл* | Цикл для перебора строк табличной части документа (СпецификацияДоговора) |
| *Движение = Движения.ВыполнениеДоговора.Добавить();* | По каждой строке добавляется запись в регистр накопления *ВыполнениеДоговоров* |
| *Движение.ВидДвижения = ВидДвиженияНакопления.Приход;* | Задается вид движения |
| *Движение.Период = Дата;* | Устанавливаются значения полей регистра накопления |
| *Движение.Контрагент = Контрагент;* |
| *Движение.Договор = Ссылка;* |
| *Движение.Услуга = СпецификацияДоговора.Услуга;* |
| *Движение.Количество = СпецификацияДоговора.Количество;* |
| *Движение.Сумма = ТекСтрока СпецификацияДоговора.Сумма;* |
| *КонецЦикла;* |  |
| *Движения.ВыполнениеДоговора.Записать();* | Сохранение записи в регистре |

Для проверки запустить приложение. Открыть любой созданный договор (создать), провести документ (кнопка ОК в форме документа). Через меню *Операции/Регистры* накопления открыть соответствующий регистр, просмотреть сформированные записи[[2]](#footnote-2).

**Движения документа Акт об оказании услуг**

При помощи конструктора движений сформировать соответствующую процедуру. Используемая табличная часть – *СпецификацияАктаПоУслугам.* Не забыть указать тип движения – расход.

На данный момент проверка движений в режиме приложения не возможна, так как еще не сформирована процедура заполнения спецификации акта по услугам. Проверим работу процедуры несколько позже.

По регистру *ДвижениеМатериалов* **самостоятельно** создать движения для документов *ПоставкаМатериалов*, *Акт об оказании услуг*. Проверить работу процедуры движения для документа акт также можно будет немного позже.

## Использование инструментария запросов

Как и любая СУБД, 1С:Предприятие (которая несомненно является таковой) поддерживает работу с хранимыми данными с использованием языка запросов (SQL). Запрос можно сформировать в любой процедуре (в любом модуле) как непосредственно на языке SQL, так и используя конструктор запросов[[3]](#footnote-3). Использование запросов позволяет осуществлять любые выборки, агрегирование данных, фильтры и т.д.

Для организации доступа к инструментарию запросов, необходимо определить запрос как объект, а потом уже формировать текст запроса. Для объекта Запрос встроенный язык 1С:Предприятие предлагает набор методов, некоторые из которых будут рассмотрены позже.

Прикладная задача: заполнение актов об оказании услуг на основании договоров с учетом выполненных работ

Возвращаемся к документу *Акт об оказании услуг*. Необходимо реализовать процедуру заполнения акта на основании договора (данная задача оговаривалась на стр. 7).

Постановка задачи:

* при заполнении акта, после выбора договора в шапке документа автоматически заполняются поля *Контрагент*, а также поля табличной части *Спецификация Акта по услугам*;
* при заполнении спецификации проверяется «остаток» по договору и в поле количество вносится не договорное, а «не выполненное» количество услуг;
* пользователь самостоятельно корректирует значение поля *Количество*. Если измененное пользователем значение превышает долг по договору, то выводится информационное сообщение.

Технология реализации заполнения спецификации акта по услугам:

Открыть форму документа *Акт об оказании услуг*. Вызвать свойства поля ввода *Договор*. Для события *При изменении* запустить создание процедуры.

Сформировать процедуру, включающую запрос. Текст процедуры с комментариями представлен ниже:

|  |  |
| --- | --- |
| *Процедура ДоговорПриИзменении(Элемент)* |  |
| *Контрагент=Договор.Контрагент;* | Заполняем поле Контрагент |
| *Запрос=Новый Запрос;* | Создаем объект *запрос*, присваиваем ему имя *Запрос* |
| *Запрос.Текст=* | Вызываем метод текст для формирования текста запроса |
| Для формирования текста запроса вызываем конструктор запроса. Вызов осуществляется через контекстное меню, команда *Конструктор запроса*. Формирование запроса в режиме конструктора описано в комментариях после текста процедуры. | |
| *"ВЫБРАТЬ*  *| ВыполнениеДоговораОстаткиИОбороты.Услуга.Ссылка КАК Услуга,*  *| ВыполнениеДоговораОстаткиИОбороты.КоличествоКонечныйОстаток КАК КоличествоОстаток,*  *| ВыполнениеДоговораОстаткиИОбороты.СуммаКонечныйОстаток КАК СуммаОстаток,*  *| ВыполнениеДоговораОстаткиИОбороты.СуммаКонечныйОстаток / ВыполнениеДоговораОстаткиИОбороты.КоличествоКонечныйОстаток КАК ЦенаПоДоговору*  *|ИЗ*  *| РегистрНакопления.ВыполнениеДоговора.ОстаткиИОбороты КАК ВыполнениеДоговораОстаткиИОбороты*  *|ГДЕ*  *| ВыполнениеДоговораОстаткиИОбороты.Договор.Ссылка = &Ссылка";* | |
| *Запрос.УстановитьПараметр("Ссылка",Договор);* | Устанавливаем значение договора |
| *ТаблЗнач=Запрос.Выполнить().Выгрузить();* | Выполняем запрос, выгружаем в таблицу значений *ТаблЗнач* (см. комментарий 2 после текста процедуры). |
| *Для каждого СтрокаТаблицыЗначений из ТаблЗнач Цикл* | Организуем цикл по перебору записей таблицы значений |
| *НоваяСтрока=СпецификацияАктаПоУслугам.Добавить();* | Добавить новую строку в спецификацию акта по услугам |
| *НоваяСтрока.Услуга= СтрокаТаблицыЗначений.Услуга;*  *НоваяСтрока.Цена= СтрокаТаблицыЗначений.ЦенаПоДоговору;*  *НоваяСтрока.Количество= СтрокаТаблицыЗначений.КоличествоОстаток;*  *НоваяСтрока.Сумма= СтрокаТаблицыЗначений.СуммаОстаток;*  *НоваяСтрока.НДС=НоваяСтрока.Сумма\*0.18;*  *НоваяСтрока.СуммаСНдс=НоваяСтрока.НДС+НоваяСтрока.Сумма;*  *КонецЦикла;* | Заполнить поля добавленной строки значениями соответствующих полей из обрабатываемой строки таблицы значений |
| ИтогоПоУслугам=СпецификацияАктаПоУслугам.Итог("СуммаСНдс");  Итого=ИтогоПоУслугам; | Пересчитать итоговые поля документа |
| КонецПроцедуры |  |
|  |  |

**Комментарии:**

1. Формирование запроса в режиме конструктора

Вид конструктора запрос представлен на Рис. 2.

1. Из окошка База данных в качестве источника выбираем регистр накопления *ВыполнениеДоговоровОстаткиИОбороты*. Это одно из представлений регистра накоплений *ВыполнениеДоговоров*, где автоматически рассчитываются показатели движения ресурсов (*КоличествоПриход, КоличествоРасход, СуммаПриход, СуммаРасход*) и их остатки ресурсов (*КоличествоКонечныйОстаток, СуммаКонечныйОстаток*).
2. Из регистра выбираем поля: *Услуга, КоличествоКонечныйОстаток, СуммаКонечныйОстаток*.
3. Для расчета цены по договору немного схитрим и рассчитаем ее как частное от деления суммы и количества (1).
4. На вкладке Условия добавить ограничения по договору (выборка идет по конкретному договору). См. Рис. 3.
5. На вкладке *Объединения/Псевдонимы* задать имена результирующих полей (см. текст запроса выше).
6. Нажать ОК, просмотреть текст запроса в модуле формы. Продолжить разработку процедуры.

1

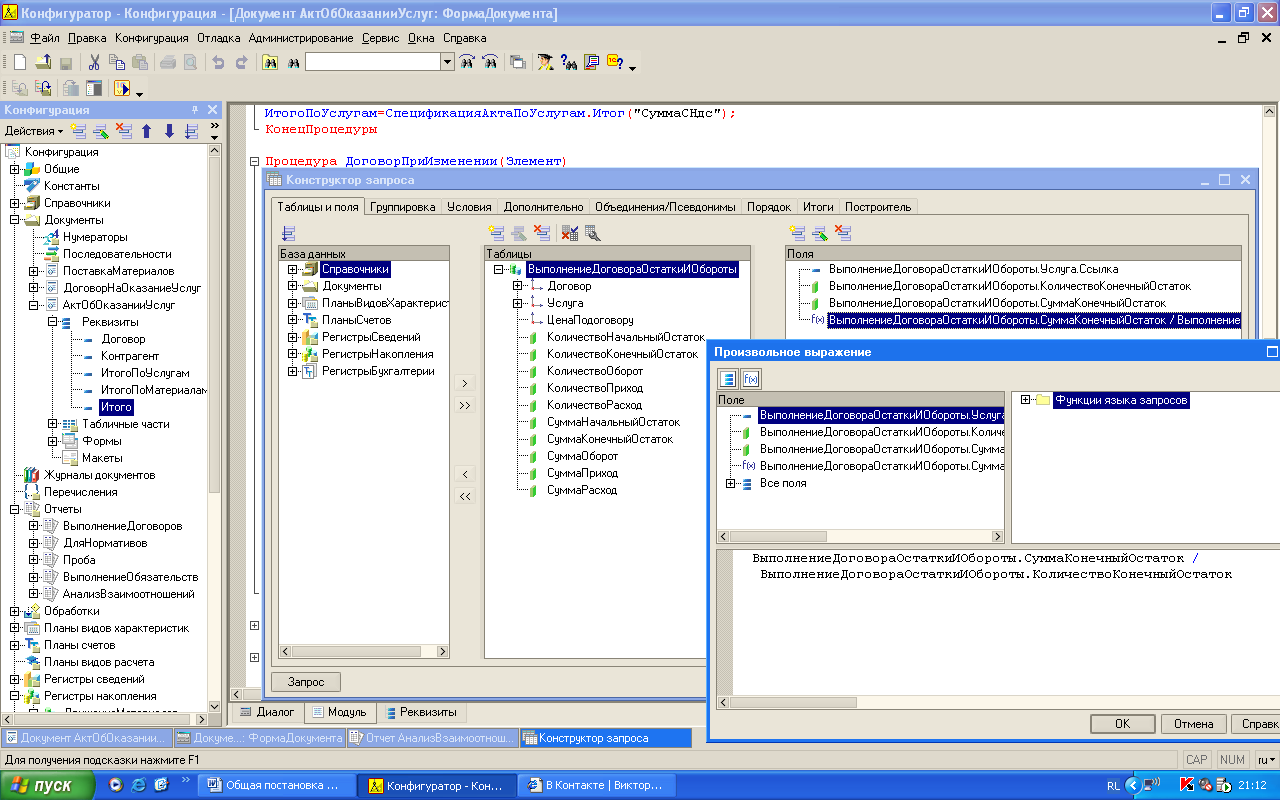


Рис. Конструктор запроса

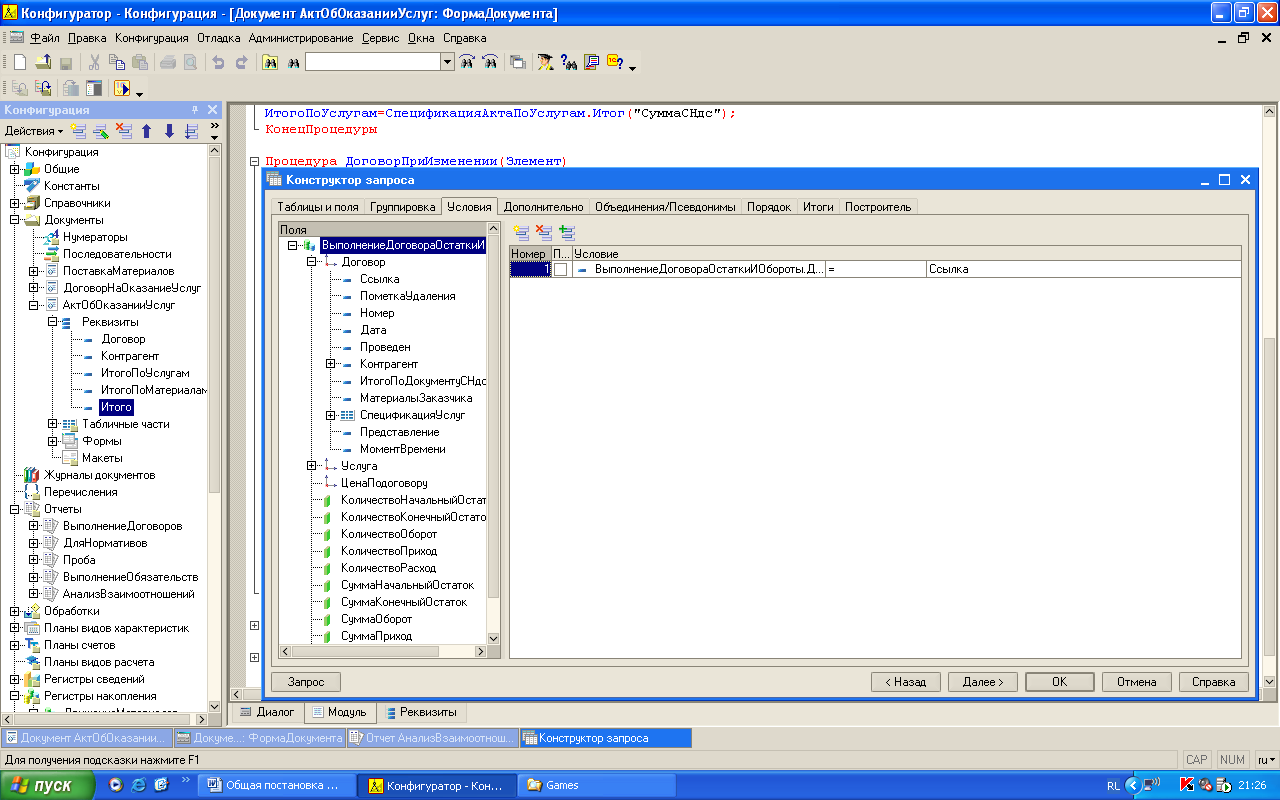


Рис. Добавление условий запроса

2. *Таблица значений* – виртуальная таблица, создаваемая на время выполнения процедуры и служащая для временного хранения данных. В рассматриваемом случае таблица значений включает 4 столбца, определенных в запросе. Записи таблицы соответствуют результату выборки (запросу). Имя таблицы задается в процедуре (*ТаблЗнач* – в рассматриваемом примере). Во встроенном языке существует набор методов для работы с таблицей значений. После выполнения процедуры таблица значений удаляется.

Технология реализации задачи контроля при корректировке значения поля

В форме документа *Акт об оказании услу*г вызвать свойства поля ввода *Количество*. Для события *При изменении* запустить создание процедуры.

|  |  |
| --- | --- |
| *Процедура СпецификацияАктаПоУслугамКоличествоПриИзменении(Элемент)* |  |
| *СТЧ=ЭлементыФормы.СпецификацияАктаПоУслугам.ТекущиеДанные;* | Ссылку на текущую строку присваиваем переменной *СТЧ* |
| *Запрос=Новый Запрос;*  *Запрос.Текст=*  *"ВЫБРАТЬ*  *|ВыполнениеДоговораОстаткиИОбороты.КоличествоКонечныйОстаток КАК КоличествоОстаток*  *|ИЗ*  *|РегистрНакопления.ВыполнениеДоговора.ОстаткиИОбороты КАК ВыполнениеДоговораОстаткиИОбороты*  *|ГДЕ*  *|ВыполнениеДоговораОстаткиИОбороты.Договор.Ссылка = &Договор*  *|И ВыполнениеДоговораОстаткиИОбороты.Услуга.Ссылка = &Услуга";* | Формирование запроса (использовать конструктор) для выборки «остатка» конкретной услуги по договору, на основании которого сформирован акт. |
| *Запрос.УстановитьПараметр("Договор",Договор);*  *Запрос.УстановитьПараметр("Услуга",СТЧ.Услуга);* | Установка условий на поля *Договор* и *Услуга* |
| *ТаблЗнач=Запрос.Выполнить().Выгрузить();* | Выгрузить результат в таблицу значений |
| *Для каждого СтрокаТаблицыЗначений из ТаблЗнач цикл*  *Если СТЧ.Количество>СтрокаТаблицыЗначений.КоличествоОстаток тогда*  *Предупреждение("Вводимое значение превышает остаток по договору");*  *Иначе*  *СТЧ.Сумма=СТЧ.Количество\*СТЧ.Цена;*  *СТЧ.НДС=СТЧ.Сумма\*0.18;*  *СТЧ.СуммаСНдс=СТЧ.Сумма+СТЧ.НДС;*  *ИтогоПоУслугам=СпецификацияАктаПоУслугам.Итог("СуммаСНдс");*  *Итого=ИтогоПоМатериалам+ИтогоПоУслугам;*  *КонецЕсли*  *КонецЦикла* | Организация цикла по строкам таблицы значений (обратить внимание – строка в этой таблице одна).  Проверка изменяемого значения количества.  При превышении значения над договорным – вывод предупреждения.  Пересчет значений связанных с количеством полей. |
| *КонецПроцедуры* |  |

После завершения и отладки вышеуказанных процедур запустить приложение, проверить правильность заполнения *Спецификации акта об оказании услуг*, а также проверить формирование записей в регистре накопления *ВыполнениеДоговоров*. Для корректной проверки по одному договору формировать несколько актов.

Прикладная задача: расчет учетных цен (формирование записей регистра сведений УчетныеЦены)

Выше упоминалось о методах формирования учетных цен на материалы для нашей организации (см. стр. 8). Теперь пришло время рассмотреть сам механизм автоматизации расчета учетных цен (формирование записей в регистре сведений *УчетныеЦены*).

Еще раз обозначим, что в организации для определения фактической себестоимости материальных ресурсов и учетных цен применяется метод ЛИФО, то есть в качестве учетной цены (в несколько упрощенном варианте) используется цена материала по последней закупке.

Сведения об учетных ценах накапливаются в соответствующем регистре сведений, расчет цен производится автоматически, запуск процедуры расчета инициируется пользователем.

Алгоритм решения задачи

Найти последнюю дату поставки по каждому материалу (исходные таблицы – *Поставка, Спецификация поставки*)

Сформировать список всех цен по поставленным материалам по всем датам (исходные таблицы – *Поставка, Спецификация поставки*)

Сформировать список цен по каждому материалу по последней его (материала) поставке

Вложенные

запросы

ЗАПРОС

Выгрузка результатов запроса в таблицу значений и добавления даты формирования учетных цен (текущей даты)

Перенос записей таблицы значений в регистр сведений *Учетные Цены*

Рис. Алгоритм формирования записей регистра Учетные цены

Технология реализации

Создать форму списка регистра сведений *УчетныеЦены*. В форме разместить кнопку, вызвать окно свойств кнопки, задать имя – FIFO. Для события *Нажатие* запустить создание процедуры.

Сформировать процедуру, включающую запрос. Текст процедуры с комментариями представлен ниже:

|  |  |
| --- | --- |
| *Процедура FIFOНажатие(Элемент)* |  |
| *Запрос=Новый Запрос;*  *Запрос.Текст=* | Создание объекта запрос и вызов метода Текст для формирования текста запроса |
| Формирование запроса в режиме конструктора описано в комментариях после текста процедуры. | |
| *"ВЫБРАТЬ*  *|МатериалДата.Материа.Ссылка КАК Материал,*  *|МатериалДата.Цен КАК Цена*  *|ИЗ*  *|(ВЫБРАТЬ*  *| МАКСИМУМ(ПоставкаМатериалов.Дата) КАК Дата,*  *| ПоставкаМатериаловСпецификацияПоставки.Материал.Ссылка КАК Материал*  *|ИЗ*  *| Документ.ПоставкаМатериалов.СпецификацияПоставки КАК ПоставкаМатериаловСпецификацияПоставки*  *| ЛЕВОЕ СОЕДИНЕНИЕ Документ.ПоставкаМатериалов КАК ПоставкаМатериалов*  *| ПО ПоставкаМатериаловСпецификацияПоставки.Ссылка = ПоставкаМатериалов.Ссылка*  *|*  *|СГРУППИРОВАТЬ ПО*  *| ПоставкаМатериаловСпецификацияПоставки.Материал.Ссылка) КАК ДатаМакс*  *| ВНУТРЕННЕЕ СОЕДИНЕНИЕ (ВЫБРАТЬ*  *| ПоставкаМатериалов.Дата КАК Дата,*  *| ПоставкаМатериаловСпецификацияПоставки.Материал.Ссылка КАК Материа,*  *| ПоставкаМатериаловСпецификацияПоставки.ЦенаПоставки КАК Цен*  *| ИЗ*  *| Документ.ПоставкаМатериалов.СпецификацияПоставки КАК ПоставкаМатериаловСпецификацияПоставки*  *| ЛЕВОЕ СОЕДИНЕНИЕ Документ.ПоставкаМатериалов КАК ПоставкаМатериалов*  *| ПО ПоставкаМатериаловСпецификацияПоставки.Ссылка = ПоставкаМатериалов.Ссылка) КАК МатериалДата*  *| ПО ДатаМакс.Дата = МатериалДата.Дата*  *| И ДатаМакс.Материал.Ссылка = МатериалДата.Материа.Ссылка";* | |
| *Табл=Запрос.Выполнить().Выгрузить();* | Выполнить запрос, выгрузить результат в таблицу значений *Табл* |
| *Табл.Колонки.Вставить(1,"Период");* | Добавить колонку в таблицу значений на первую позицию |
| *Табл.ЗаполнитьЗначения(ТекущаяДата(), "Период");* | Заполнить новую колонку значением текущей даты (даты формирования учетных цен) |
| *Для каждого СтрокаТаблицыЗначений из Табл цикл* | Организовать цикл по перебору строк таблицы значений |
| *НоваяСтрока= РегистрыСведений.УчетныеЦены. СоздатьМенеджерЗаписи();* | Создает объект «НоваяСтрока» для управления записью регистра сведений. По-другому говоря, создает новую запись в регистре сведений. |
| *НоваяСтрока.Материал= СтрокаТаблицыЗначений. Материал;*  *НоваяСтрока.Период= СтрокаТаблицыЗначений. Период;*  *НоваяСтрока.ЦенаУчетная= СтрокаТаблицыЗначений. Цена;* | Присваивает значения измерениям и ресурсам добавленной строки регистра сведений из обрабатываемой строки таблицы значений. |
| НоваяСтрока.Записать();  КонецЦикла | Записываем новую строку регистра сведений |
| КонецПроцедуры |  |

После окончания запустить приложение, открыть регистр накопления *УчетныеЦены*, проверить формирование записей регистра.

**Комментарии:**

Для выборки требуемых сведений (см. Рис. 4) в качестве источников результирующего запроса необходимо сформировать два вложенных. Первый вложенный запрос (имя запроса *ДатаМакс*) сформирует список последних дат поставки по каждому материалу. Второй вложенный запрос (имя – *МатериалДата*) создаст список всех цен поставки по всем материалам с привязкой к дате поставки.

Разработка запроса *ДатаМакс*:

1. Для создания вложенного запроса в режиме конструктора запроса в окне *Таблицы* нажать кнопку *Добавить*. Откроется окошко вложенного запроса (см. Рис. 5).
2. Для вложенного запроса источником являются таблицы: *ПоставкаМатериалов, СпецификацияПоставки*. Поля выборки: *МатериалСсылка* (из спецификации поставки), дата поставки (из таблицы *ПоставкаМатериалов*).
3. Связь между таблицами устанавливается на вкладке *Связи* конструктора запроса по полю *ПоставкаМатериалов.Ссылка* (номер документа поставки).
4. На вкладке *Группировка* определить поле группировки - *МатериалСсылка*, суммируемое (точнее агрегируемое) поле – Дата поставки, функция агрегирования – максимум.

На вкладке *Объединения/Псевдонимы* задать имена полей выборки: Дата, Материал.

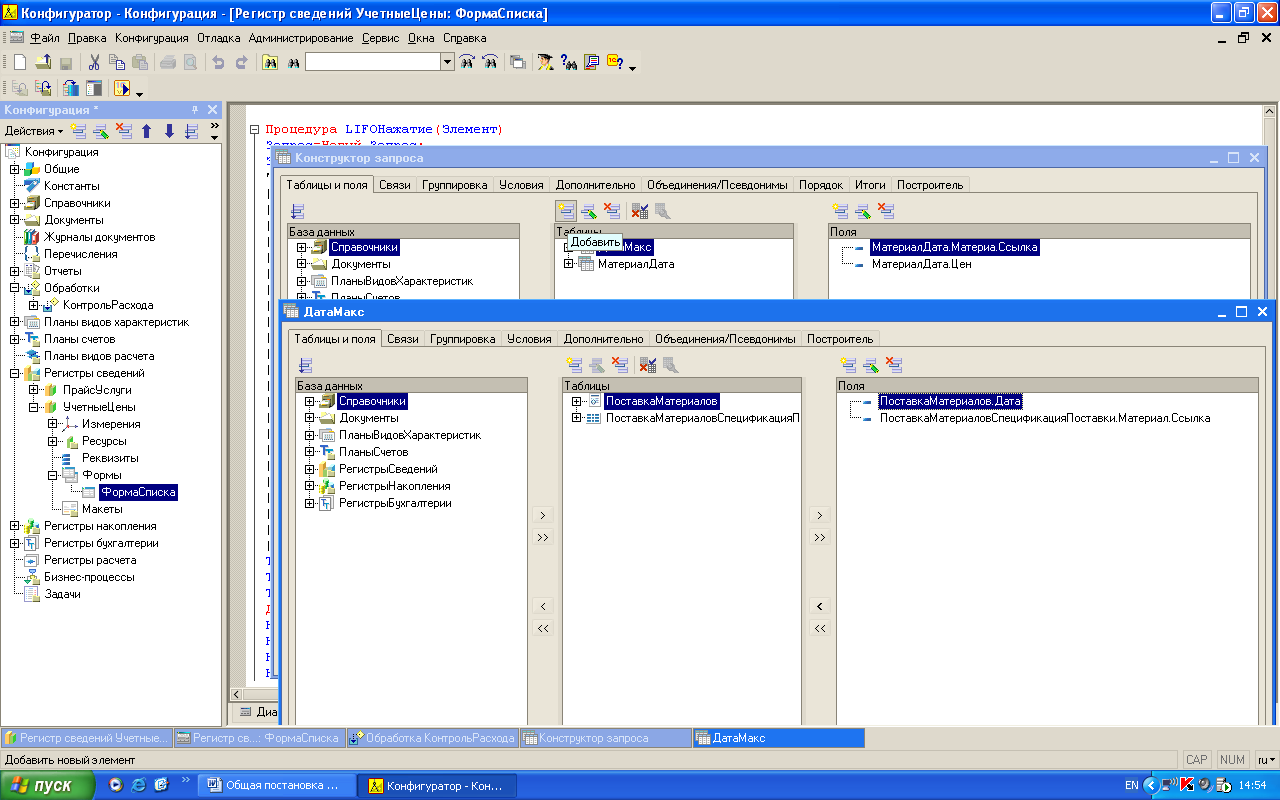


Рис. Создание вложенного запроса

1. Просмотреть запрос (кнопка *Запрос* в левом нижнем углу окна конструктора).
2. Завершить создание вложенного запроса (кнопка ОК). В окне конструктора главного запроса, в разделе *Таблицы* появился новый объект – *Вложенный запрос*. Используя контекстное меню переименовать его в *ДатаМакс*.

Разработка вложенного запроса *МатериалДата*:

1. Создать второй вложенный запрос
2. Источники: *ПоставкаМатериалов, СпецификацияПоставки*. Поля выборки: *МатериалСсылка* (из спецификации поставки), дата поставки (из таблицы *ПоставкаМатериалов*), *ЦенаПоставки*.
3. Проверить связи между таблицами. Задать имена полей выборки.
4. Завершить создание вложенного запроса, присвоить новому запросу имя *МатериалДата*.

После завершения создания вложенных запросов окно основного запроса имеет вид:

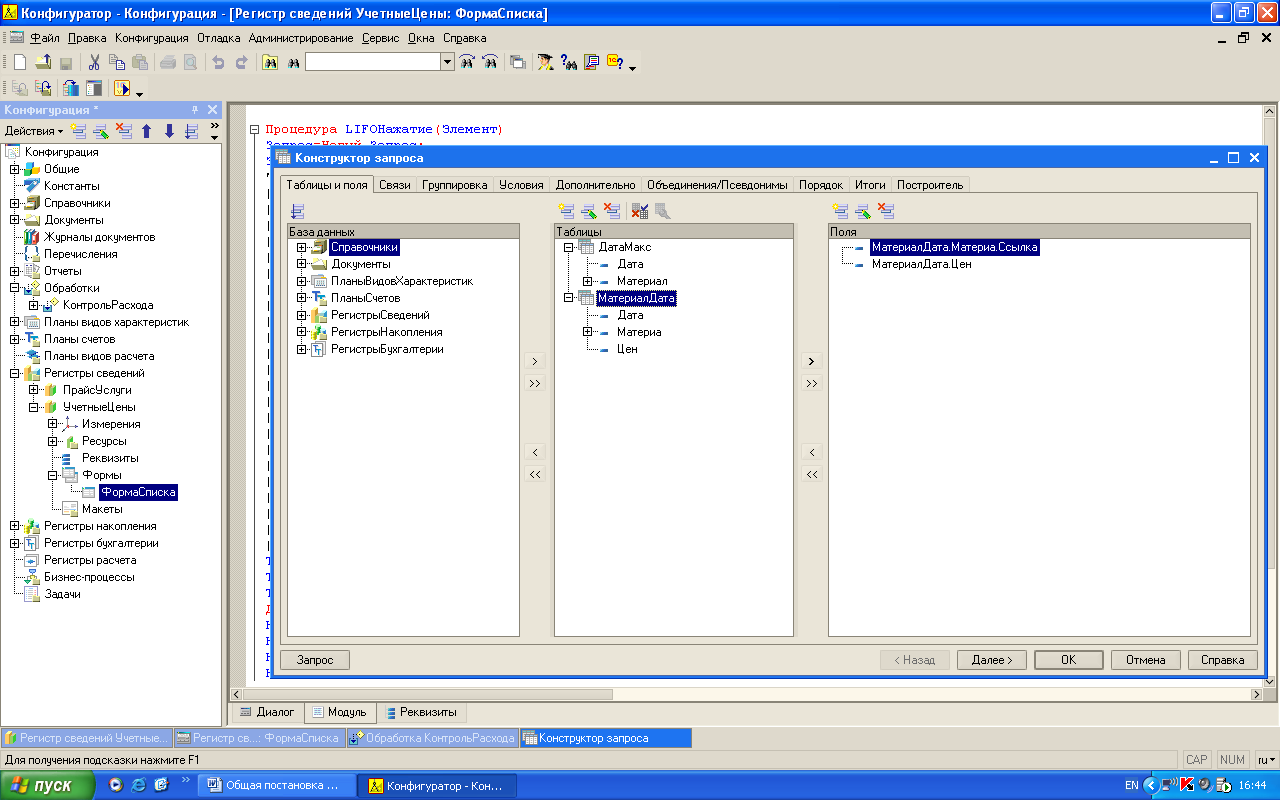


Рис. Вид основного запроса

Поля выборки основного запроса: *МатериалДата.Материал.Ссылка, МатериалДата.Цена.*

На вкладке Связи основного запроса установить связи между вложенными запросами по двум полям: *Дата, Материал*.

Переименовать поля выборки на вкладке *Объединения/Псевдонимы*.

Просмотреть итоговый текст запроса, завершить создание запроса, просмотреть текст запроса в процедуре.

**Самостоятельно** сформировать процедуры заполнения *Акта об оказании услуг* на основании последних по датам записей регистра сведений *Учетные* *цены*.

## Проведение бухгалтерских документов

Технологию бухгалтерского учета в автоматизированной бухгалтерской системе можно представить следующим образом:

1. Настройка и поддержка объектов аналитического учета.
2. Настройка и поддержка плана (планов) счетов с учетом аналитического учета.
3. Настройка регистров бухгалтерского учета на основании используемого плана счетов.
4. Проведение документов (хозяйственных операций) по счетам бухгалтерского учета, формирование записей регистра бухгалтерии.
5. Формирование сводных бухгалтерских регистров и итоговой бухгалтерской отчетности.

Для обеспечения бухгалтерского учета в инструментальной среде 1С используются следующие объекты конфигурации:

1. План видов характеристик: поддерживает ведение объектов аналитического учета.
2. План счетов: предназначен для описания структуры хранения информации о синтетических счетах организации с детализацией по аналитическим объектам.
3. Регистры бухгалтерии: предназначены для хранения данных о хозяйственных операциях, отраженных на счетах бухгалтерского учета.

### Хозяйственные операции в рассматриваемой задаче

При проведении хозяйственных операций формируются следующие проводки: см. Таблица 1.

Таблица Проводки хозяйственных операций

| **СчД** | **Субконто дебета 1** | **Субконто дебета 2** | **СчК** | **Субконто кредита 1** | **Субконто кредита 2** | **Количество** | **Характеристика** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *По документу Поставка материалов* | | | | | | | |
| 10.1 | Материал |  | 60.1 | Контрагент |  | Да | Акцептованы платежные требования за поступившие материалы |
| 19.3 | Контрагент |  | 60.1 | Контрагент |  |  | Выделен НДС по материалам |
| *По документу акт об указании услуг* | | | | | | | |
| 20 |  |  | 10.01 | Материалы |  |  | Списана стоимость материалов по учетным ценам |
| 43 | Услуга |  | 20 |  |  |  | Произведена готовая продукция (услуги) по учетным ценам |
| 90.02 | Услуга |  | 43 | Услуга |  |  | Списаны фактические затраты на продажи (по себестоимости) |
| 62.01 | Контрагент | Договор | 90.01 | Услуга |  |  | Проданы услуги Возникла дебиторская задолженность. Сумма рассчитывается по продажной цене услуги |
| 90.3 |  |  | 68.02 |  |  |  | На сумму НДС по оказанным услугам |
| *По документу Наряд на выполненный работы* | | | | | | | |
| 20 |  |  | 70 | Договор | Вид работ |  | На сумму начисленной ЗП |

Таблица . Используемые счета

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Счет** | **Наименование** | **Субконто1** | **Субконто2** | **Тип счета** |
| 10.1 | Сырье и материалы | Материал | Договор | Активный |
| 19.3 | НДС по приобретенным МПЗ | Контрагент |  | Активный |
| 20 | основное производство | Услуга | Договор | Активный |
| 43 | готовая продукция | Услуга | Договор | Активный |
| 60.1 | расчеты с поставщиками и подрядчиками в рублях | Контрагент |  | Пассивный |
| 62.01 | расчеты с покупателями и заказчиками | Контрагент | Договор | Активный |
| 68.02 | налог на добавленную стоимость |  |  | Активный/ Пассивный |
| 90.01 | выручка | Услуга | Договор | Пассивный |
| 90.02 | себестоимость | Услуга | Договор | Активный |
| 91.01 | прочие доходы |  |  | Активный |
| 91.02 | прочие расходы |  |  | Активный |
| 90.03 | налог на добавленную стоимость (продажи) |  |  | Активный |
| 70 | Расчеты с персоналом по оплате труда | Договор | Работник | Пассивный |
|  |  |  |  |  |

### Формирование структур для поддержки объектов аналитики

Для ведения бухгалтерского учета необходимо обеспечить аналитический учет на большинстве из счетов. Для обозначения объектов аналитики в терминах 1С используется понятие субконто.

Для формирования субконто используется объект метаданных *План видов характеристик*. Данный объект предназначен для описания структуры хранения информации о некоторых характеристиках, задаваемых пользователем. Информация состоит из названия характеристики и ее типа. Причем, тип характеристик может являться составным – набор разных документов, справочников, чисел и др. Например, мы хотим создать дополнительный разрез представления услуг в регистре сведений по дополнительной характеристике (к примеру, предварительная подготовка). В свою очередь дополнительная подготовка включает другие услуги, поставку материалов и пр. Создавать дополнительный справочник видов дополнительной подготовки нет смысла. Поэтому создается *План видов характеристик* с перечнем видов подготовки, составным типом данных – справочниками услуги, материалы; документом Поставка материалов.

С позиции бухгалтерского учета для ведения аналитики создается план видов характеристик, включающий все возможные объекты (справочники, перечисления, документы), являющимися объектами аналитического учета. В дальнейшем данный план характеристик используется при настройке планов счетов.

Технология разработки плана видов характеристик *ВидыСубконто*.

1. Создать новый план видов характеристик, имя – *ВидыСубконто*. Тип значения характеристик – составной тип. В окне редактирования типа отметить все справочники и документы (аналитический разрез возможен в разрезе любого из указанных объектов), а также типы: булево, строка.
2. На вкладке *Иерархия* установить признак иерархичности записей объекта.
3. На вкладке *Прочие* задать предопределенные значения субконто: *Договор* (тип - документ *Договор)*, *Контрагент*, *Материал*, *Услуга ВидРабот, Сотрудник, СтатьяЗатрат* (типы – соответствующие справочники).

### Технология разработки и настройки Плана счетов

Создать новый план счетов, имя объекта – *Бухгалтерский*.

На вкладке *Данные* установить признак учета - *Количественный*, для него установить тип данных – *Булево*. Количественный признак учета обозначает возможность учета по счету в разрезах установленных субконто (аналитиках) в единицах измерения.

На вкладке *Субконто* задать максимальное количество субконто – 3. В качестве видов субконто выбрать план характеристик *ВидыСубконто*. Также установить признак учета аналогично предыдущему пункту.

На вкладке *Прочее* задать предопределенные счета в соответствии с Таблица 2. При создании каждого счета указывать: имя счета, код (номер) счета, тип счета, признак учета, субконто.

### Разработка бухгалтерских регистров

По своим свойствам регистры бухгалтерии похожи на регистры накопления. Записи регистров бухгалтерии также формируются при проведении документов. Отличием является его связь с планом счетов. Для поддержки формирования проводок по документу (двойной записи) регистры бухгалтерии содержат обязательные реквизиты *Счет Дт* и *Счет Кт*, а также механизм поддержки аналитического учета в соответствии с установленными для счетов субконто. По одному документу может быть сформировано несколько записей (несколько проводок). На основании записей бухгалтерских регистров формируются бухгалтерские итоги – остатки и обороты по счетам.

Технология разработки регистра бухгалтерии *Бухгалтерский*.

Создать новый регистр бухгалтерии, присвоить ему имя. Связать с планом счетов *Бухгалтерский*. Указать признак формирования корреспондирующей записи (двойной записи).

На вкладке *Данные* задать измерения: *Количество*, *Сумма*, что соответствует правилам формирования проводок. Если вдруг появится необходимость отображения дополнительной информации в бухгалтерском регистре, то можно создать дополнительные реквизиты.

На вкладке *Регистраторы* указываются документы, по которым формируются записи хозяйственных операций. В рассматриваемом примере такими документами являются: *Поставка материалов, Акт об оказании услуг, Наряд на выполненный работы.*

Для данного учебного примера формирование всех объектов, связанных с бухгалтерским учетом завершено.

### Разработка процедур проведения документов по регистру бухгалтерии

Все процедуры движения документа формируются в модуле документа. Для разработки движений используются конструктора движения регистров, с которыми знакомились в разделе «Разработка движений документов».

**Проведение документа Поставка материалов**

Для указанного документа должно быть сформировано две проводки в соответствии с Таблица 1.

1. Открыть документ *Договор*, перейти на вкладку *Движения*. Активизировать конструктор движений, добавить новое движение по регистру *Бухгалтерский*.
2. Указать счета дебета и *кредита (Сырье и материалы, Расчеты с поставщиками и подрядчиками в рублях*) в соответствующих разделах. Выбрать табличную часть – *Спецификация поставки*. Поставить в соответствие ресурсам и субконто регистра реквизиты документа. Окно конструктора движения по бухгалтерскому регистру представлено на Рис. 7.
3. Добавить следующее движение по регистру бухгалтерии для данного документа. Счет дебета - *НДС по приобретенным МПЗ*, счет кредита - *Расчеты с поставщиками и подрядчиками в рублях*. Заполнить субконто и ресурсы проводки (обратить внимание, что возможны два одинаковых значения субконто). Для заполнения ресурса *Сумма* сформировать выражение*: ИтогоПоДокументу-ИтогоБезНДС* (соответствующие реквизиты документа выводить двойным щелчком).
4. После завершения создания движений (выхода из конструктора движений) на экране отобразится сформированная процедура в модуле объекта *Поставка материалов*. Фрагмент процедуры, связанный с наполнением бухгалтерских регистров представлен ниже.

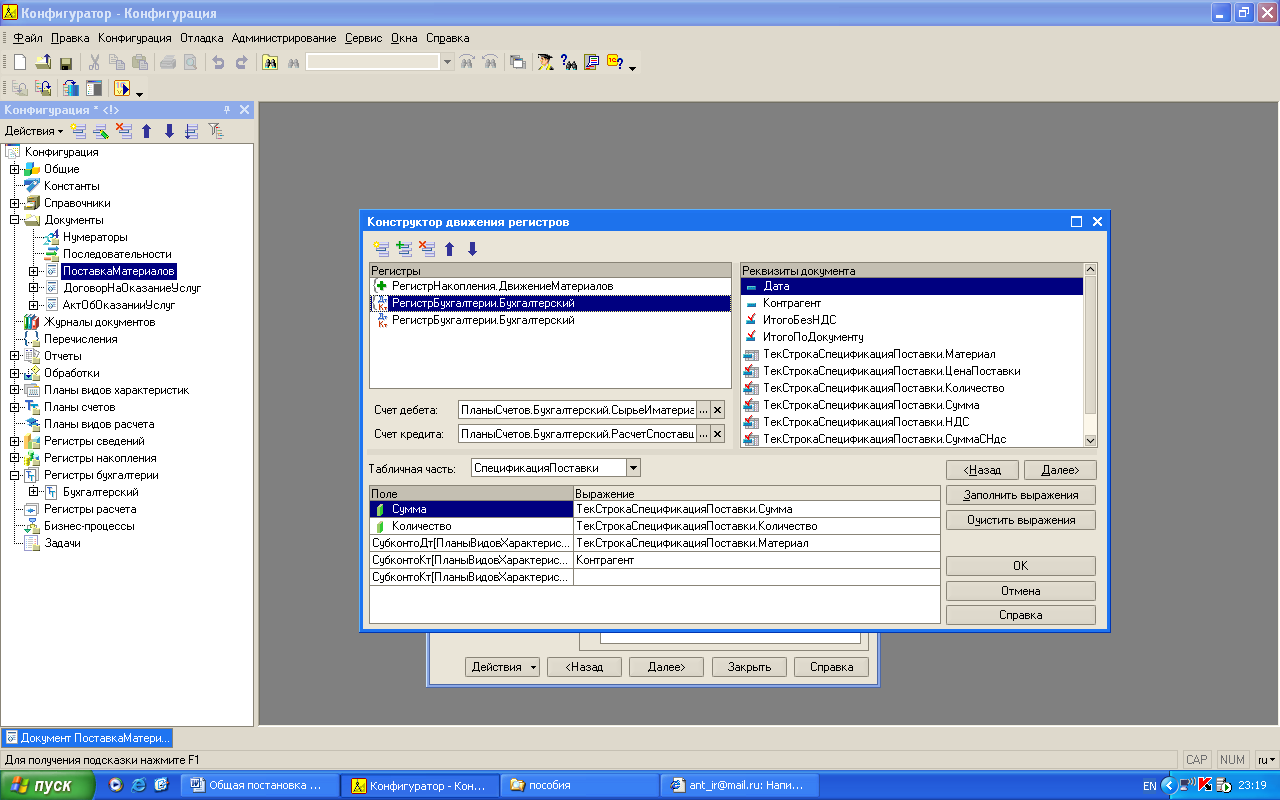


Рис. Формирование записей регистра бухгалтерии в режиме конструктора

*Для Каждого ТекСтрокаСпецификацияПоставки Из СпецификацияПоставки Цикл*

*// регистр Бухгалтерский*

*Движение = Движения.Бухгалтерский.Добавить();*

*Движение.СчетДт = ПланыСчетов.Бухгалтерский.СырьеИматериалы;*

*Движение.СчетКт = ПланыСчетов.Бухгалтерский. РасчетСпоставщикамиВруб;*

*Движение.Период = Дата;*

*Движение.Сумма = ТекСтрокаСпецификацияПоставки.Сумма;*

*Движение.Количество = ТекСтрокаСпецификацияПоставки.Количество;*

*Движение.СубконтоДт[ПланыВидовХарактеристик.ВидыСубконто.Материал] = ТекСтрокаСпецификацияПоставки.Материал;*

*Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Контрагент] = Контрагент;*

*КонецЦикла;*

*// регистр Бухгалтерский*

*Движение = Движения.Бухгалтерский.Добавить();*

*Движение.СчетДт = ПланыСчетов.Бухгалтерский.НДСПоПриобретеннымМПЗ;*

*Движение.СчетКт = ПланыСчетов.Бухгалтерский.РасчетСпоставщикамиВруб;*

*Движение.Период = Дата;*

*Движение.Сумма = ИтогоПоДокументу-ИтогоБезНДС;*

*Движение.СубконтоДт[ПланыВидовХарактеристик.ВидыСубконто.Контрагент] = Контрагент;*

*Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Контрагент] = Контрагент;*

*// записываем движения регистров*

*Движения.ДвижениеМатериалов.Записать();*

*Движения.Бухгалтерский.Записать();*

Для проверки запустить приложение. Открыть любой документ поставки (создать), провести документ. Через меню *Операции/Регистры* *бухгалтерии* открыть *Бухгалтерский* регистр, просмотреть сформированные записи.

**Самостоятельно**. Провести остальные документы.

## Поддержка управленческого учета

В бухгалтерском и управленческом учете и управленческом учете несколько различны направления использования полученной информации. Управленческий учет требует более глубокой детализации, а также значительного объема дополнительной информации.

Детализация (аналитика) учета затрат будет проводиться по следующим основаниям:

* по статьям затрат;
* по объектам затрат;

Места возникновения затрат (центры затрат) в данной учебной задаче рассматриваться не будут.

Статьи затрат представлены в справочнике *Статьи калькуляции*.

В данном примере используются следующие статьи:

Прямые материальные затраты;

Прямые расходы на оплату труда (основные рабочие)

Косвенные затраты: оплата труда АУП, прочие затраты.

Объектом затрат является *Услуга по договору* (конкретная услуга по определенному договору).

План счетов управленческого учета включает следующие статьи:

20 «Основное производство» (договор, услуга, материал)

26 «Общехозяйственные расходы»

27 «Распределение общих затрат» (договор, услуга, статья затрат, вид работ, материал)

70 «Расчеты с персоналом по оплате труда» (Договор, сотрудник)

10 «Материалы» (договор, материал)

### Формирование структур для поддержки объектов аналитики

Для ведения и бухгалтерского и управленческого учета необходимо обеспечить аналитический учет на большинстве из счетов. Для обозначения объектов аналитики в терминах 1С используется понятие субконто.

Для формирования субконто используется объект метаданных *План видов характеристик*. Данный объект предназначен для описания структуры хранения информации о некоторых характеристиках, задаваемых пользователем. Информация состоит из названия характеристики и ее типа. Причем, тип характеристик может являться составным – набор разных документов, справочников, чисел и др. Например, мы хотим создать дополнительный разрез представления услуг в регистре сведений по дополнительной характеристике (к примеру, предварительная подготовка). В свою очередь дополнительная подготовка включает другие услуги, поставку материалов и пр. Создавать дополнительный справочник видов дополнительной подготовки нет смысла. Поэтому создается *План видов характеристик* с перечнем видов подготовки, составным типом данных – справочниками услуги, материалы; документом Поставка материалов.

С позиции бухгалтерского учета для ведения аналитики создается план видов характеристик, включающий все возможные объекты (справочники, перечисления, документы), являющимися объектами аналитического учета. В дальнейшем данный план характеристик используется при настройке планов счетов.

Технология разработки плана видов характеристик *ВидыСубконто*.

1. Создать новый план видов характеристик, имя – *ВидыСубконто*. Тип значения характеристик – составной тип. В окне редактирования типа отметить все справочники и документы (аналитический разрез возможен в разрезе любого из указанных объектов), а также типы: булево, строка.
2. На вкладке *Иерархия* установить признак иерархичности записей объекта.
3. На вкладке *Прочие* задать предопределенные значения субконто: *Договор* (тип - документ *Договор)*, *Контрагент*, *Материал*, *Услуга, СтатьяКалькуляции, ВидРабот* (типы – соответствующие справочники).

### Технология разработки и настройки Плана счетов

Создать новый план счетов, имя объекта – *Управленческий*.

На вкладке *Данные* установить признак учета - *Количественный*, для него установить тип данных – *Булево*. Количественный признак учета обозначает возможность учета по счету в разрезах установленных субконто (аналитиках) в единицах измерения.

На вкладке *Субконто* задать максимальное количество субконто – 3. В качестве видов субконто выбрать план характеристик *ВидыСубконто*. Также установить признак учета аналогично предыдущему пункту.

На вкладке *Прочее* задать предопределенные счета в соответствии с установленным в начале раздела списком. При создании каждого счета указывать: имя счета, код (номер) счета, тип счета, признак учета, субконто.

### Разработка управленческих регистров

По своим свойствам регистры бухгалтерии (а управленческие регистры по логике схожи с бухгалтерскими) похожи на регистры накопления. Записи регистров также формируются при проведении документов. Отличием является его связь с планом счетов. Для поддержки формирования проводок по документу (двойной записи) регистры бухгалтерии содержат обязательные реквизиты *Счет Дт* и *Счет Кт*, а также механизм поддержки аналитического учета в соответствии с установленными для счетов субконто. По одному документу может быть сформировано несколько записей (несколько проводок). На основании записей бухгалтерских регистров формируются бухгалтерские итоги – остатки и обороты по счетам.

Технология разработки регистра бухгалтерии *Управленческий*.

Создать новый регистр бухгалтерии, присвоить ему имя. Связать с планом счетов *Управленческий*. Указать признак формирования корреспондирующей записи (двойной записи).

На вкладке *Данные* задать измерения: *Количество*, *Сумма*, что соответствует правилам формирования проводок. Если вдруг появится необходимость отображения дополнительной информации в бухгалтерском регистре, то можно создать дополнительные реквизиты.

На вкладке *Регистраторы* указываются документы, по которым формируются записи хозяйственных операций. В рассматриваемом примере такими документами являются: *Поставка материалов, Акт об оказании услуг, НарядНаВыполненные работы.*

Для данного учебного примера формирование всех объектов, связанных с бухгалтерским учетом завершено.

### Разработка процедур проведения документов по регистру Управленческий

Все процедуры движения документа формируются в модуле документа. Для разработки движений используются конструктора движения регистров, с которыми знакомились в разделе «Разработка движений документов».

По существующим документам сформируем следующие движения по счетам управленческого учета

| **СчД** | **Субконто дебета** | **СчК** | **Субконто кредита 1** | **Количество** | **Характеристика** |
| --- | --- | --- | --- | --- | --- |
| *По документу акт об указании услуг* | | | | | |
| 27 | договор, услуга, статья затрат, материал | 20 | договор, услуга, материал |  | Списаны затраты по оказанным услугам |
| *По документу Наряд на выполненный работы* | | | | | |
| 20 | договор, статья затрат, вид работ, | 70 | Договор, сотрудник | Вид работ | Списаны затраты на сумму начисленной ЗП |

Самостоятельно сформировать проведение документов по счетам управленческого учета.

### Работа с документом Карточка себестоимости

Данный документ формируется до начала выполнения договора по потребности пользователя. Содержит сведения по нормативной и фактической калькуляции себестоимости услуги по договору. По договору формируется столько карточек, сколько оговорено наименований услуг.

**Создание документа Карточка себестоимости**

Создать документ следующей структуры:

Шапка документа: № документа, Дата, Договор, Услуга, МатЗатратыНорматив, МатЗатратыФакт, ЗП\_Норматив, ЗП\_Факт, ИтогоНорматив, ИтогоФакт, Отклонение

Табличная часть *Материальные затраты*: ВидРабот, Материал, НомаРасход, ФактРасход, НормаСумма, ФактСумма, Отклонение, ОтклонениеСумма.

Табличная часть *ЗП Основных рабочих*: ВидРабот, Должность, НомаРасход, ФактРасход, НормаСумма, ФактСумма, Отклонение, ОтклонениеСумма

Создать форму документа.

### Разработка обработки Формирование карточки

**Постановка задачи:**

Обеспечить создание карточки себестоимости по выбранному договору.

Для реализации задачи можно воспользоваться объектом конфигурации *Обработка*. Обработки предназначены для реализации различных обработок информации и сервисных функций. Обработка, как правило, содержит процедуры, реализующие алгоритмы обработки и представления информации, формы, при помощи которых организуется интерфейс для ввода параметров алгоритмов и представления результатов их работы, а также может содержать макеты, используемые для вывода результатов выполнения алгоритма обработки в виде отчета.

Создать новую обработку, присвоить имя. Создать реквизиты обработки, через представление которых в форме будут вводиться параметры для процедуры: *Договор*.

Примерный вид формы обработки представлен на рисунке

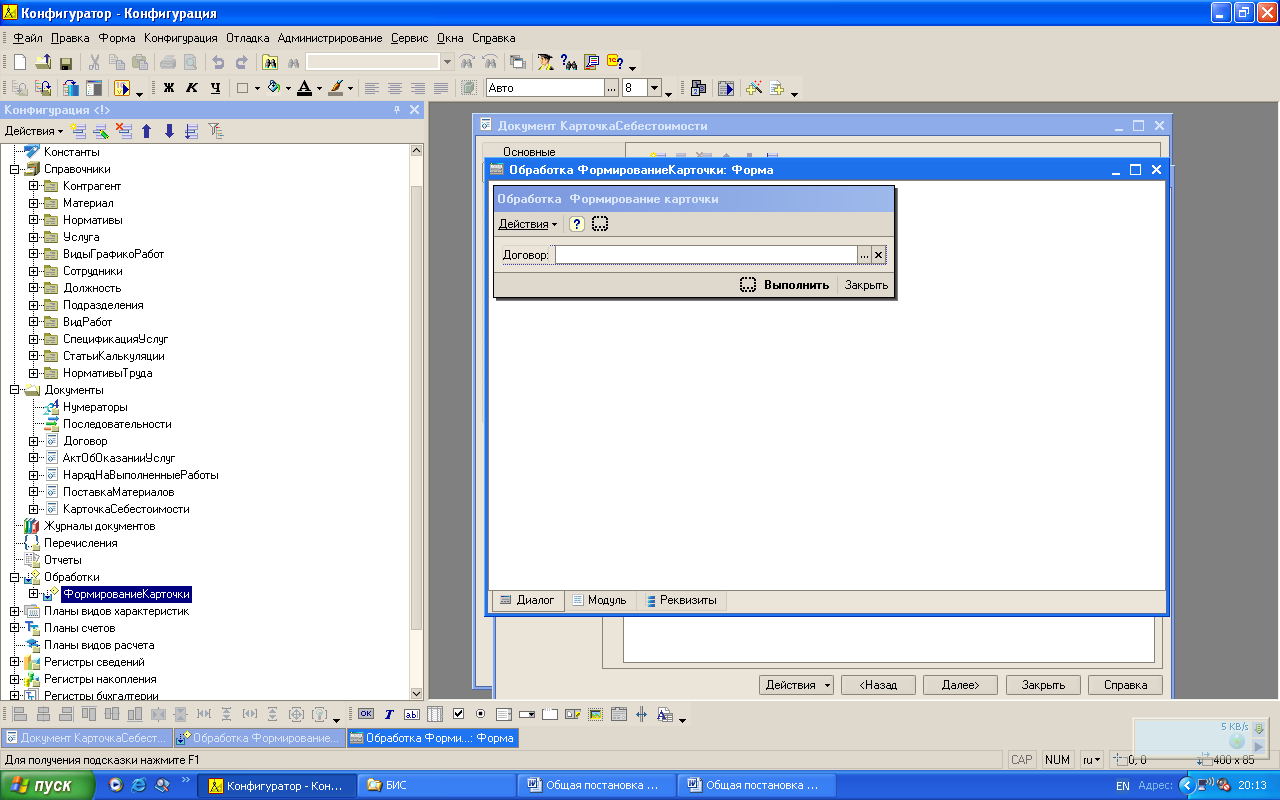


Рис.

Создать новую форму (имя – Форма), куда добавляем все реквизиты. В окне формы вызываем свойства кнопки Выполнить. Далее перейти на вкладку Модуль и раскрыть заготовку процедуры *КнопкаВыполнитьНажатие*, где и будет формироваться процедура.

Алгоритм решения задачи

Для выбранного номера договора осуществляется поиск соответствующего документа. Для каждой строки спецификации договора формируется документ *Карточка себестоимости*, куда переносятся значения текущей даты, договора, услуги. Для заполнения табличной части Материальные затраты создается запрос по следующей схеме:

Для выбранной услуги формируется таблица следующей структуры: ВидРабот Количество

Услуга

Спецификация услуги

*УслугаРабота*

Формируется таблица следующей структуры: ВидРабот, Материал, Количество

*РаботаМатериал*

НормативМатериалов

Спецификация Норматива

Рассчитывается Количество материалов, необходимое по каждому виду работ для данной услуги

Технология реализации

Сформировать процедуру, включающую запрос. Текст процедуры представлен ниже:

*Процедура КнопкаВыполнитьНажатие(Кнопка)*

*Дог=Документы.Договор.НайтиПоНомеру(Договор.Номер);*

*Для каждого СтрокаТабличнойЧасти из Дог.Спецификация цикл*

*НовыйДокумент=Документы.КарточкаСебестоимости.СоздатьДокумент();*

*НовыйДокумент.Дата=ТекущаяДата();*

*НовыйДокумент.Договор=Договор;*

*НовыйДокумент.Услуга=СтрокаТабличнойЧасти.Услуга;*

*Запрос=Новый Запрос;*

*Запрос.Текст=*

*"ВЫБРАТЬ*

*| РаботаМатериал.ВидРабот,*

*| РаботаМатериал.Материал,*

*| СУММА(РаботаМатериал.Количество \* УслугаРабота.КоличествоНаЕИ) КАК Количество*

*|ИЗ*

*| (ВЫБРАТЬ*

*| СпецификацияУслугСпецификация.ВидРабот КАК ВидРабот,*

*| СпецификацияУслугСпецификация.КоличествоНаЕИ КАК КоличествоНаЕИ*

*| ИЗ*

*| Справочник.СпецификацияУслуг.Спецификация КАК СпецификацияУслугСпецификация*

*| ЛЕВОЕ СОЕДИНЕНИЕ Справочник.СпецификацияУслуг КАК СпецификацияУслуг*

*| ПО СпецификацияУслугСпецификация.Ссылка = СпецификацияУслуг.Ссылка*

*| ГДЕ*

*| СпецификацияУслуг.Услуга = &Услуга) КАК УслугаРабота*

*| ВНУТРЕННЕЕ СОЕДИНЕНИЕ (ВЫБРАТЬ*

*| Нормативы.ВидРабот КАК ВидРабот,*

*| НормативыСпецификация.Материал КАК Материал,*

*| НормативыСпецификация.Количество КАК Количество*

*| ИЗ*

*| Справочник.Нормативы.Спецификация КАК НормативыСпецификация*

*| ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Нормативы КАК Нормативы*

*| ПО НормативыСпецификация.Ссылка = Нормативы.Ссылка) КАК РаботаМатериал*

*| ПО УслугаРабота.ВидРабот = РаботаМатериал.ВидРабот*

*|*

*|СГРУППИРОВАТЬ ПО*

*| РаботаМатериал.ВидРабот,*

*| РаботаМатериал.Материал";*

*Запрос.УстановитьПараметр("Услуга",СтрокаТабличнойЧасти.Услуга);*

*Табл=Запрос.Выполнить().Выгрузить();*

*Для каждого СтрокаТаблЗначений из Табл Цикл*

*НоваяСтрока=НовыйДокумент.МатериальныеЗатраты.Добавить();*

*НоваяСтрока.ВидРабот=СтрокаТаблЗначений.ВидРабот;*

*НоваяСтрока.МатериалГруппа=СтрокаТаблЗначений.Материал;*

*НоваяСтрока.НорматРасход=СтрокаТаблЗначений.Количество;*

*Отбор=Новый Структура("Материал",СтрокаТаблЗначений.Материал);*

*Значение=РегистрыСведений.ОтпускныеЦеныМатериалов.ПолучитьПоследнее(ТекущаяДата(), Отбор);*

*НоваяСтрока.НормСумма=Значение.ОтпускнаяЦена\*СтрокаТаблЗначений.Количество;*

*КонецЦикла;*

*НовыйДокумент.ИтогоНорм=НовыйДокумент.МатериальныеЗатраты.Итог("НормСумма");*

*НовыйДокумент.Записать();*

*НовыйДокумент.ИтогоНорм=НовыйДокумент.МатериальныеЗатраты.Итог("НормСумма");*

*НовыйДокумент.ПолучитьФорму("ФормаДокумента").Открыть();*

*КонецЦикла*

*КонецПроцедуры*

## Поддержка организации и контроля бизнес-процессов

**Бизнес-процесс** - устойчивая целенаправленная совокупность взаимосвязанных видов деятельности, которая по определенной технологии преобразует входы в выходы, представляющая ценность для потребителя. Потребителем может являться другой процесс или внешний субъект (физическое, юридическое лицо, функциональное подразделение). **Регламент бизнес-процесса** — документ, описывающий последовательность операций, ответственность, порядок взаимодействия исполнителей и порядок принятия решения по улучшениям.

При построении информационной системы модель бизнес-процессов является основой. Именно она определяет набор модулей, функций, операций и, в конце концов, модель данных и алгоритмы обработки. Но кроме поддержки операций, происходящих в бизнес-системе, необходимо отслеживать и цепочки взаимосвязанных операций, входящих в бизнес-процесс; максимально автоматизировать обработку тех или иных событий, возникающих при выполнении операций; запускать задачи, соответствующие логике бизнес-процеса; контролировать прохождение бизнес-процесса.

1С:Предприятие включает инструменты, поддерживающие вышеобозначенные задачи. Логика технологии поддержки бизнес-процесса представлена на Рис. 9.

Взаимосвязь процессов представляется с помощью карты маршрута *бизнес-процесса*. На карте отражаются все операции, исполнители (пользователи), а также алгоритм прохождения бизнес-процесса. К каждому объекту карты можно «привязать» процедуру обработки. Переход по маршруту возможен как автоматически, так и интерактивно. Запуск и завершение операций (задач) можно инициировать из любого объекта метаданных. Перед формированием объектов бизнес-процессов необходимо установить правила адресации исполнителей бизнес-процессов.

Старт бизнес-процесса

Исполнитель

Операция бизнес-процесса (точка действия)

Исполнитель

Операция бизнес-процесса (точка действия)

Условие (точка условия)

Исполнитель

Операция бизнес-процесса (точка действия)

Точка ветвления

Исполнитель

Операция бизнес-процесса (точка действия)

Исполнитель

Операция бизнес-процесса (точка действия)

Точка слияния

Завершение бизнес-процесса

Исполнитель

Операция бизнес-процесса (точка действия)

Завершение бизнес-процесса

Объект «Задача»: данные о задаче (процессе).

Выполнено=Истина

Запуск процесса (создание задачи)

Завершение процесса (присвоение статуса «выполнено»)

Рис. Схема объекта бизнес-процесс в среде 1С:Предприятие

### Описание бизнес-процесса Контроль выполнения договоров (Договора)

Бизнес-процес «*Договора*» включает операции, связанные с отслеживанием выполнения договора от момента подписания до полного выполнения всех обязательств (оказания всего перечня услуг). Данный бизнес-процесс запускается для каждого договора после его подписания. Операции (задачи) бизнес процесса представлены в

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Задача (точка действия)** | **Описание** | **Запуск** | **Завершение** | **Исполнитель** |
| *Запустить договор в исполнение* | Исполнитель присваивает договору статус исполняемого. | Сразу после запуска бизнес-процесса (подписания договора). Договор считается подписанным, если в документе устанавливается значение *Истина* для реквизита *Подписан*. | Реквизиту *Исполняется* присваивается значение *Истина*. | Главный (менеджер) |
| *Формирование карточек* | Формирование карточек себестоимости по договору | Сразу после завершения задачи *Запустить договор в исполнение* | При первичном проведении карточки. В Бизнес-процессе формируется соответствующий признак. |  |
| *Сформировать акт* | Формирование и проведение акта об оказании услуг. | Сразу после завершения задачи *Формирование карточек* | При проведении *Акта об оказании услуг*. Рассчитывается долг по договору. Значение показателя заносится в поле *Долг* документа *Договор*. Событие завершает задачу. | Исполнитель |
| После выполнения задачи *Сформировать акт* осуществляется проверка полноты закрытия договора. Если поле *Долг* договора равно нулю, то возможен переход на задачу *Закрыть* *договор*. В противном случае еще раз запускается задача *Сформировать акт* | | | | |
| *Закрыть* *договор* | Закрытие договора. Исполнитель присваивает договору статус *Выполнено*. | После завершения задачи *Сформировать акт*, при условии нулевого долга по договору. | Реквизиту *Выполнено* присваивается значение *Истина*. Данная задача завершает бизнес-процесс Договора. | Главный (менеджер) |

### Программная реализация бизнес-процесса Договора

Предлагается следующий алгоритм:

1. Определение исполнителей задач (адресация бизнес-процесов);
2. Формирование *Задачи*. Структура задачи одинакова для всех точек действия бизнес-процесса.
3. Создание объекта метаданных бизнес-процесс *Договора*.
4. Разработка процедур перехода по маршруту бизнес-процесса.
5. Формирование системы контроля за ходом бизнес-процесса, а также системы оповещения.

В ходе реализации поставленной задачи потребуется выполнить действия по изменению структур некоторых документов, разработке дополнительных обработок.

Создание системы адресации бизнес-процессов

Можно выделить два вида адресации:

* Персональная – исполнитель (пользователь) определяется в явном виде.
* Ролевая – исполнитель определяется исходя из назначенных ему ролей или других реквизитов адресации.

Данные для системы адресации определяются в регистре сведений. Каждое измерение такого регистра определяет определенный разрез адресации. В нашей задаче исполнитель будет назначаться либо в явном виде (конкретный пользователь), либо по должности.

Создать регистр сведений. Имя регистра – *Адресация*. Регистр независимый, непериодический. Измерения регистра:

*Сотрудник* (тип данных – справочник *Сотрудники*);

*Роль исполнителя* (тип данных – справочник *Должности*).

Ресурсов и реквизитов нет.

В рассматриваемом примере установлено, что основная адресация будет идти по конкретным сотрудникам. Считаем, что есть два условных сотрудника, которые участвуют в данном бизнес-процессе*: Главный, Исполните*ль. В справочнике *Сотрудники* в режиме конфигуратора задать данных сотрудников как предопределенных (вкладка *Прочие* окна справочника *Сотрудники*).

Для предопределенных сотрудников необходимо создать *Параметры сеанса*. В дальнейшем при запуске системы пользователю можно поставить необходимый параметр сеанса и определить его для системы адресации бизнес-процесса. Параметры сеанса создаются в общих объектах метаданных.

Создать два *параметра сеанса*:

Имя – *Главный*. Тип - *СправочникСсылка*.*Сотрудники*.

Имя – *Исполнитель*. Тип - *СправочникСсылка.Сотрудники*.

Процедура присвоения параметров сеансов пользователям будет рассмотрена позже.

**Дополнение**. К данному этапу необходимо создать пользователей системы. Их будет два: Главный, Исполнитель.

Разработка Задачи

При продвижении по маршруту бизнес-процесса на каждой точке действия будут создаваться задачи. Можно назвать их пользовательскими заданиями, которые будут выполняться заданными пользователями. Объект задача включает следующие атрибуты:

* Предопределенные: *Дата, Номер, Наименование, Выполнена* (тип – булево), *Бизнес-процесс* (ссылка на бизнес-процесс), *ТочкаМаршрута* (ссылка на точку маршрута бизнес-процесса).
* Пользовательские: любые реквизиты, необходимые для идентификации и контроля для задачами.

В рассматриваемом примере структура задач для всех точек действия бизнес-процесса будет одинакова.

Создание задачи для бизнес-процесса:

1. Создать задачу. Имя – *ЗадачаДляДоговора*.
2. На вкладке *Адресация* (см. Рис. 10) в одноименном списке выбрать регистр сведений Адресация.

Текущий исполнитель – параметр сеанса *Главный*.

Задать реквизиты адресации: *Сотрудник*, Роль исполнителя (тип – справочник *должности*).

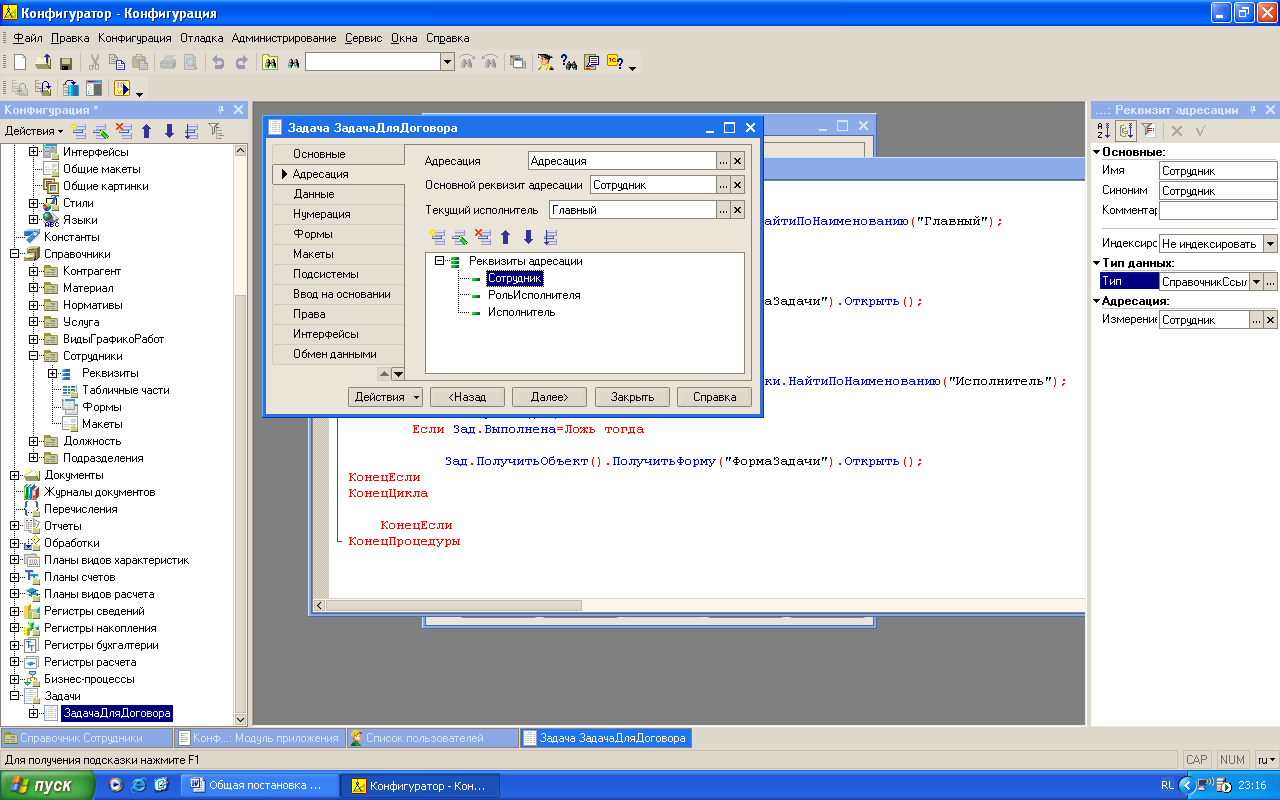


Рис.

В списке *Основной реквизит адресации* установить реквизит *Сотрудник*.

1. На вкладке Данные задать реквизиты:

*Сотрудник* (тип – справочник Сотрудник);

*Договор* (тип – документ Договор).

1. Создать форму списка и форму задачи (см. Рис. 11).

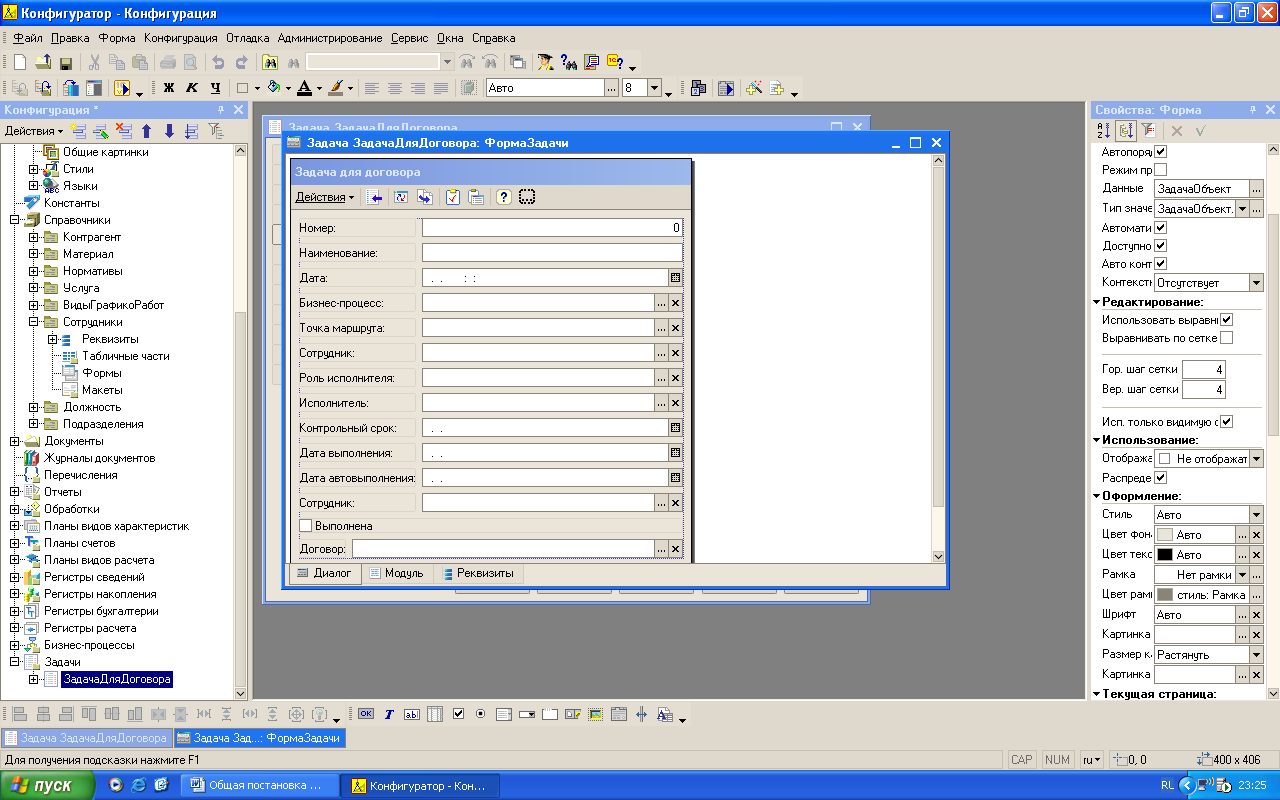


Рис. Форма задачи в режиме конструктора

Разработка бизнес-процесса

1. Создать объект конфигурации бизнес-процесс. Имя – Договора. Выбрать задачу для бизнес-процесса – *ЗадачаДляДоговора.*
2. На вкладке Данные создать реквизиты – *Договор* (тип данных – документ *ДоговорНаОказаниеУслуг*), НаличиеКарточки (тип данных – булево).
3. Создать форму бизнес-процесса.

На вкладке *Прочие* активизировать создание карты маршрута. Карта маршрута представлена на рис.

Для каждой точки действия в разделе Адресация (в свойствах точки маршрута) указать сотрудника и (при необходимости) роль исполнителя для бизнес-процесса. Указанный сотрудник является ответственным за выполнение задачи, соответствующий пользователь будет иметь доступ к задаче, а также имеет право запускать процедуры, связанные с точкой действия бизнес-процесса.

Разработка процедур перехода по маршруту бизнес-процесса.

Прежде чем формировать объявленные процедуры, создадим процедуру *ПриЗапускеСистемы* в общем модуле приложения. Данная процедура запускается каждый раз при запуске системы и инициирует пользователей системы. Кроме того для каждого пользователя задается вывод тех задач, выполнение которых сейчас необходимо в соответствии с картой бизнес-процесса.

Для формирования процедуры необходимо выбрать объект *Конфигурация*, вызвать контекстное меню, выбрать команду Открыть модуль приложения. В окне модуля сформировать следующую процедуру:

*Процедура ПриНачалеРаботыСистемы()*

*Если ИмяПользователя()="Главный" тогда*

*ПараметрыСеанса.Главный=Справочники.Сотрудники.НайтиПоНаименованию("Главный");*

*Зад=Задачи.ЗадачаДляДоговора.Выбрать();*

*Пока Зад.Следующий() Цикл*

*Если Зад.Выполнена=Ложь тогда*

*Зад.ПолучитьОбъект().ПолучитьФорму("ФормаЗадачи").Открыть();*

*КонецЕсли*

*КонецЦикла*

*КонецЕсли*

*КонецПроцедуры*

**Запуск бизнес-процесса. Формирование задачи *Запустить договор в исполнение***

Бизнес-процесс *Договора* запускается в тот момент, когда пользователь устанавливает в поле подписан *Договора на оказание услуг* значение *Истина*. В форме документа сформировать процедуру:

*Процедура ПодписанПриИзменении(Элемент)*

*Если Подписан=Истина тогда*

*БП=БизнесПроцессы.Договора.СоздатьБизнесПроцесс();*

*БП.Дата=ТекущаяДата();*

*БП.Договор=Ссылка;*

*БП.Записать();*

*БП.Старт();*

*КонецЕсли*

*КонецПроцедуры*

По данной процедуре создается бизнес-процесс *Договора* и запускается первая задача

Для пользователя будет удобно, если система его информирует о запуске бизнес-процесса и первой задаче. Для обеспечения указанных условий необходимо сделать следующее:

На карте маршрута бизнес процесса активизировать соответствующую задачу (точку маршрута). В окне свойств точки маршрута, в разделе события активизировать создание процедуры *ПередСозданиемЗадач*. Сформировать соответствующую процедуру:

*Процедура ЗапуститьДоговорВИсполнениеПередСозданиемЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, СтандартнаяОбработка)*

*Сообщить("Запускается бизне-процесс Договора и первая задача");*

*КонецПроцедуры*

Следующая процедура – *ПриСозданииЗадач* (активизируется в том же окне свойств). Каждый раз при создании задачи формируются значения некоторых реквизитов, создается соответствующее сообщение. Процедура представлена ниже:

*Процедура ЗапуститьДоговорВИсполнениеПриСозданииЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, Отказ)*

*Для каждого Задача из ФормируемыеЗадачи цикл*

*Задача.Договор=Ссылка.Договор;*

*КонецЦикла;*

*Сообщить("Запущена задача Запустить договор в исполнение");*

*КонецПроцедуры*

Теперь есть смысл запустить приложение. Создать новый договор, установить ему статус Подписано, проверить работу процедур задачи. Также есть смысл открыть список бизнес-процессов и задач, проанализировать данные списка.

**Завершение задачи *Запустить договор в исполнение*. Формирование задачи *Формирование карточек*.**

Задача проверки запаса завершается как только установлено значение *Истина* для поля *Исполняется*. Процедура, сформированная в форме обработки представлена ниже:

*Процедура ВыполняетсяПриИзменении(Элемент)*

*Если Выполняется=Истина тогда*

*ЗП=Задачи.ЗадачаДляДоговора.Выбрать();*

*Пока ЗП.Следующий() Цикл*

*Если ЗП.Наименование="Запустить договор в исполнение" И ЗП.Выполнена=Ложь тогда*

*Если ЗП.Договор=Ссылка тогда*

*Сообщить(ЗП.Наименование);*

*ЗП.ПолучитьОбъект().ВыполнитьЗадачу();*

*КонецЕсли*

*КонецЕсли*

*КонецЦикла*

*КонецЕсли*

*КонецПроцедуры*

После завершения автоматически запускается следующая задача. Пользователя необходимо информировать об этом.

На карте маршрута бизнес процесса активизировать соответствующую задачу (точку маршрута). В окне свойств точки маршрута, в разделе события активизировать создание процедуры *ПередСозданиемЗадач*. Сформировать соответствующую процедуру:

*Процедура ФормированиеКарточекПередСозданиемЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, СтандартнаяОбработка)*

*Сообщить("Запускается задача Формирование карточек. Создайте карточки себестоимости");*

*КонецПроцедуры*

Далее сформировать процедуру *ПриСозданииЗадач*:

*Процедура ФормированиеКарточекПриСозданииЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, Отказ)*

*Для каждого Задача из ФормируемыеЗадачи цикл*

*Задача.Договор=Ссылка.Договор;*

*КонецЦикла;*

*КонецПроцедуры*

Данная процедура также создает обработку для формирования карточек себестоимости по выбранному договору.

*Процедура ФормированиеКарточекПриСозданииЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, Отказ)*

*Для каждого Задача из ФормируемыеЗадачи цикл*

*Задача.Договор=Ссылка.Договор;*

*КонецЦикла;*

*Обр=Обработки.ФормированиеКарточки.Создать();*

*Обр.Договор=Ссылка.Договор;*

*Обр.ПолучитьФорму("Форма").Открыть();*

*КонецПроцедуры*

**Завершение задачи Формирование Карточки**

Данная задача завершается при первом проведении карточки себестоимости по заданному договору. Процедура проведения представлена ниже:

*Процедура ОбработкаПроведения(Отказ, Режим)*

*ЗП=Задачи.ЗадачаДляДоговора.Выбрать();*

*Пока ЗП.Следующий() Цикл*

*Если ЗП.Наименование="Формирование карточек" И ЗП.Выполнена=Ложь тогда*

*Если ЗП.Договор=Договор Тогда*

*ЗП.ПолучитьОбъект().ВыполнитьЗадачу();*

*БП=ЗП.БизнесПроцесс.ПолучитьОбъект();*

*БП.НаличиеКарточки=Истина;*

*КонецЕсли*

*КонецЕсли*

*КонецЦикла*

*КонецПроцедуры*

Автоматически запускается следующая задача *Сформировать акт*. Для соответствующей точки бизнес-процесса сформировать процедуры *Перед создание задачи*, *При создании задачи*:

*Процедура СформироватьАктПередСозданиемЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, СтандартнаяОбработка)*

*Сообщить("Запускается задача Формирование акта. По готовности создать акт");*

*КонецПроцедуры*

*Процедура СформироватьАктПриСозданииЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, Отказ)*

*Для каждого Задача из ФормируемыеЗадачи цикл*

*Задача.Договор=Ссылка.Договор;*

*КонецЦикла;*

*КонецПроцедуры*

**Завершение задачи *Сформировать акт*. Формирование задачи *Проверить долг*.**

Задача формирования акта завершается при каждом формировании акта по указанному договору независимо от долга по договору. Акт считается (для системы) вступившим в действие, если он проведен. Процедура завершения данной задачи формируется как дополнение к процедуре *ОбработкаПроведения* в модуле *документа АктОбОказании услуг*. Текст дополнения представлен ниже:

*ЗП=Задачи.ЗадачаДляДоговора.Выбрать();*

*Пока ЗП.Следующий() Цикл*

*Если ЗП.Наименование="Сформировать акт" и ЗП.Выполнена=Ложь тогда*

*Если ЗП.Договор=Договор тогда*

*ЗП.ПолучитьОбъект().ВыполнитьЗадачу();*

*КонецЕсли*

*КонецЕсли*

*КонецЦикла*

Автоматически запускается следующая задача *Проверить долг.* Для комфортного выполнения задачи создадим обработку Контроль договора.

1. Сформировать обработку *Контроль договора (см.* Рис. 12*)*.

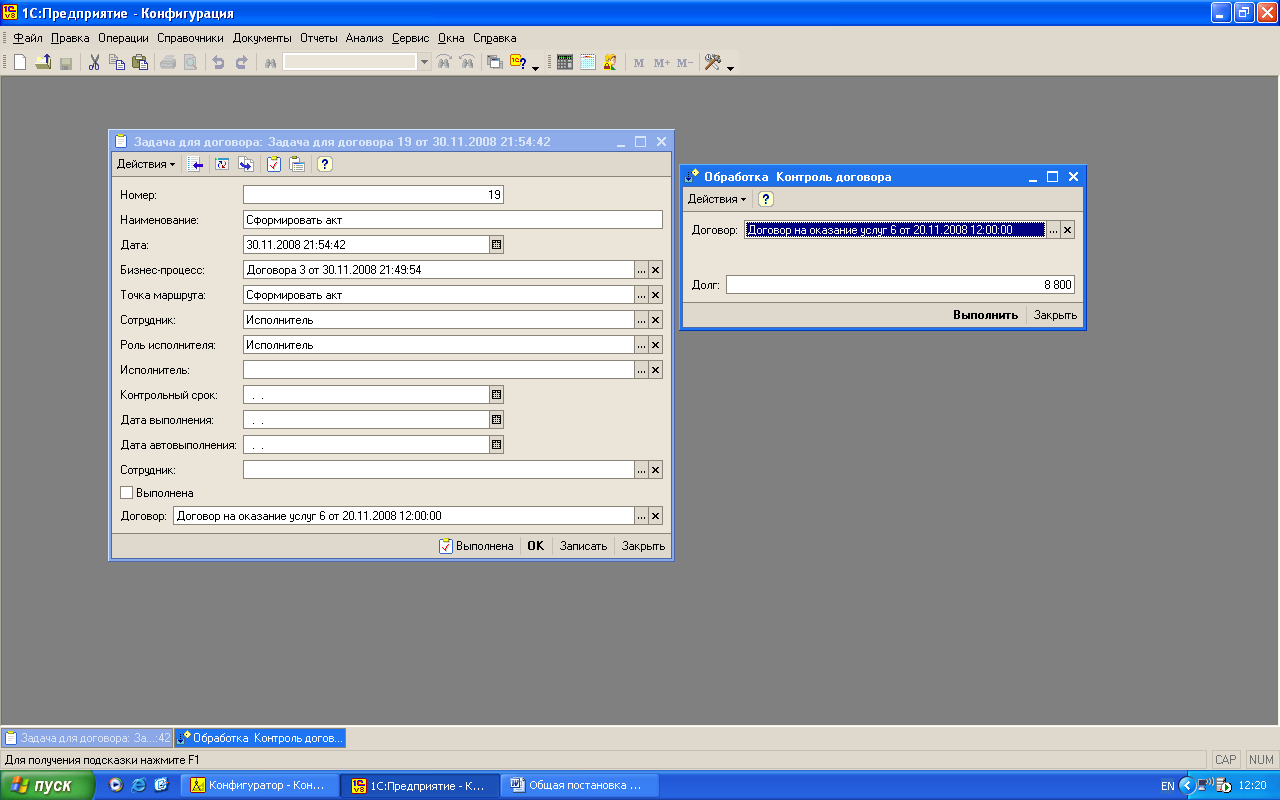


Рис. Форма обработки Контроль договора

Данная обработка запускается после выполнения задачи *Сформировать акт*. Пользователь анализирует долг по договору. Система проверяет поле долг и при записывает значение долга в соответствующий договор. Процедура обработки запускается нажатием кнопки Выполнить. Независимо от наличия долга задача *Сформировать акт* завершается и осуществляется переход на точку условия бизнес-процесса.

Для соответствующей точки бизнес-процесса сформировать процедуры *Перед создание задачи*, *При создании задачи*:

*Процедура ПроверитьДолгПриСозданииЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, Отказ)*

*Для каждого Задача из ФормируемыеЗадачи цикл*

*Задача.Договор=Ссылка.Договор;*

*КонецЦикла;*

*Обр=Обработки.КонтрольДоговора.Создать();*

*Обр.Договор=Ссылка.Договор;*

*Обр.ПолучитьФорму("Форма").Открыть();*

*КонецПроцедуры*

Процедура заполнения обработки представлена ниже.

*Процедура КнопкаВыполнитьНажатие(Кнопка)*

*Запрос=Новый Запрос;*

*Запрос.Текст=*

*"ВЫБРАТЬ*

*| СУММА(ВыполнениеДоговоровОстаткиИОбороты.СуммаКонечныйОстаток) КАК СуммаКонечныйОстаток*

*|ИЗ*

*| РегистрНакопления.ВыполнениеДоговоров.ОстаткиИОбороты КАК ВыполнениеДоговоровОстаткиИОбороты*

*|ГДЕ*

*| ВыполнениеДоговоровОстаткиИОбороты.Договор = &Договор";*

*Запрос.УстановитьПараметр("Договор",Договор);*

*Табл=Запрос.Выполнить().Выгрузить();*

*Для каждого СтрокаТаблицыЗначений из Табл Цикл*

*Долг=СтрокаТаблицыЗначений.СуммаКонечныйОстаток;*

*Док=Документы.Договор.НайтиПоНомеру(Договор.Номер);*

*Объект=Док.ПолучитьОбъект();*

*Объект.Долг=СтрокаТаблицыЗначений.СуммаКонечныйОстаток;*

*Объект.Записать();*

*КонецЦикла;*

*ЗП=Задачи.ЗадачаДляДоговора.Выбрать();*

*Пока ЗП.Следующий() цикл*

*Если ЗП.Наименование="Проверить долг" тогда*

*Если ЗП.Выполнена=Ложь и ЗП.Договор=Договор тогда*

*ЗП.ПолучитьОбъект().ВыполнитьЗадачу();*

*КонецЕсли*

*КонецЕсли*

*КонецЦикла*

*КонецПроцедуры*

После завершения данной задачи маршрут бизнес-процесса подошел к точке условия. Проверка условия осуществляется по документу договор, ссылка на который установлена в бизнес-процессе (каждый бизнес-процесс соответствует одному договору).

Для формирования процедуры проверки на карте маршрута бизнес-процесса активизировать точку условия *Выполнены обязательства*, в разделе события активизировать событие *проверка условия*. Текст процедуры представлен ниже:

Процедура ВыполненыОбязательстваПроверкаУсловия(ТочкаМаршрутаБизнесПроцесса, Результат)

*Если Договор.Долг>0 тогда*

*Результат=Ложь*

*Иначе*

*Результат=Истина*

*КонецЕсли*

*КонецПроцедуры*

В соответствии с результатом проверки или опять создается задача *Сформировать акт* (вторая такая задача), или запускается задача *Закрыть договор*.

Переход на задачу *Закрыть договор* возможен только тогда, когда долг будет равен нулю. По одному договору может быть сформировано несколько актов (несколько этапов), соответственно и задач *Сформировать акт* может быть несколько.

Для задачи Закрыть договор создать процедуру, открывающую форму закрываемого договора:

*Процедура ЗакрытьДоговорПриСозданииЗадач(ТочкаМаршрутаБизнесПроцесса, ФормируемыеЗадачи, Отказ)*

*Для каждого Задача из ФормируемыеЗадачи цикл*

*Задача.Договор=Договор;*

*Дог=Документы.Договор.НайтиПоНомеру(Договор.Номер);*

*Дог.ПолучитьФорму("ФормаДокумента").Открыть();*

*КонецЦикла;*

*КонецПроцедуры*

**Завершение задачи *Закрыть договор*. Завершение бизнес-процесса.**

Задача закрытия договора завершается, как только установлено значение *Истина* для поля *Выполнено* в форме документа соответствующего договора.

Процедура завершения указанной задачи формируется в модуле формы договора:

*Процедура ВыполненПриИзменении(Элемент)*

*Если Выполнен=Истина тогда*

*ЗП=Задачи.ЗадачаДляДоговора.Выбрать();*

*Пока ЗП.Следующий() цикл*

*Если ЗП.Наименование="Закрыть договор" и ЗП.Выполнена=Ложь тогда*

*Если ЗП.Договор=Ссылка тогда*

*ЗП.ПолучитьОбъект().ВыполнитьЗадачу();*

*КонецЕсли*

*КонецЕсли*

*КонецЦикла*

*КонецЕсли*

*КонецПроцедуры*

Завершение данной задачи завершает бизнес-процесс. Пользователю будет комфортно, если он получит соответствующее сообщение. Для этого активизировать блок завершение в карте маршрута бизнес-процесса и сформировать процедуру:

*Процедура ЗавершениеПриЗавершении(ТочкаМаршрутаБизнесПроцесса, Отказ)*

*Предупреждение("Бизнес-процесс завершен",,"Окончание")*

*КонецПроцедуры*

## Поддержка периодических расчетов

В экономических системах достаточно распространены сложные периодические расчеты. Самый наглядный пример – расчет заработной платы. При расчете заработной платы (выплат сотрудникам) приходится учитывать большое количество составляющих: оклад, премия, больничный лист, штрафы, исполнительные листы, отсутствие на работе и т.д. Влияние каждой составляющей связано с определенным алгоритмом. Причем данное влияние рассматривается не только с точки зрения правил расчета, но и сроком (отрезком времени), в течение которого составляющая влияет. Общий алгоритм взаимосвязи и взаимовлияния составляющих представлен ниже.

Невыход

Больничный

*Период расчета*

Премия= %\*ОКЛАД

ОКЛАД

Оплата больничного листа

ОКЛАД

Рис. . Общая схема периодического расчета выплат сотрудникам

В представленном случае можно выделить следующие составляющие (*виды расчета*): оклад, премия, невыход, больничный. Премия и больничные рассчитываются на основе оклада, или – оклад является *базой* для расчета премии и оплате по больничному листу. Причем, расчеты премии идут не просто на основании установленного оклада, а на окладе, приведенном к периоду. То есть если сотрудник работал только часть периода расчета (часть месяца), то и размер оклада и премия рассчитываются только за отработанные дни. Такая зависимость называется *зависимостью по базовому периоду*. В рассматриваемом примере период действия соответствует месяцу (смотри период расчета на Рис. 13). *Фактическим периодом действия* является: *Период расчета - Дни невыхода – Дни на больничном*. Такое влияние называется *вытеснением по периоду действия*.

Кроме того, для каждого расчета определяют:

*Базовые виды расчета* – те виды расчета, которые являются базой для устанавливаемого вида;

*Вытесняющие виды расчета* - те виды расчета, которые вытесняют устанавливаемый вид по периоду действия;

*Ведущие виды расчета* - те виды расчета, изменение результатов которых приводит к перерасчету устанавливаемого вида расчета.

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| *Виды начисления* | *Базовые* | *Вытесняющие* | *Ведущие* |
| Оклад |  | Невыход  Больничный |  |
| Невыход |  |  |  |
| Премия | Оклад | Больничный | Оклад  Невыход |
| Больничный | Оклад |  | Оклад |

### Формирование структур для поддержки периодических расчетов

Для поддержки обработки периодических расчетов в среде 1С:Предприятие существуют следующие объекты:

*План видов расчета* – предназначен для описания структур хранения информации о видах расчета. Заполняется пользователем, но есть возможность разработчиком задавать предопределенные виды расчета.

*Регистр расчета* – описывает структуру накопления данных, являющихся результатом расчетов. Данный регистр не предназначен для интерактивного редактирования пользователем.

Разработка планов видов расчетов

Создать новый план видов расчетов – *ОсновныеНачисления*. На вкладке Расчет устанавливаем флажок *Зависит по периоду действия*. Установка данного свойства обозначает, что в этом плане будут находиться виды расчетов, которые вытесняют другие по периоду действия. Например, Невыход вытесняет Оклад.

В качестве базовых планов видов расчетов указывается сам план *ОсновныеНачисления.*

На вкладке Прочее создать предопределенные планы видов расчета в соответствии с Таблица 3.

Создание вспомогательных справочников и регистров

Для расчета заработной платы потребуется информация о графике работы разных категорий работников: администрации и исполнителей.

Прежде всего необходимо создать два дополнительных справочника:

Должности. Реквизиты: Код, Наименование, Оклад.

Сотрудники: Код, Наименование, Должность.

Необходимо создать справочник *ВидыГрафиковРабот*. Реквизиты – Код, Наименование. Задать два предопределенных видов графиков: *ГрафикАдминистрации, ГрафикИсполнителей*.

Непосредственно сам график на каждый месяц будет формироваться в регистре сведений *ГрафикиРабот*. Необходимо создать соответствующий регистр сведений. Измерения регистра: График (тип данных – ссылка на *ВидыГрафиковРабот), Дата*. Ресурсы регистра – Значение, число длиной 1 (может принимать два значения: 1 – если день рабочий, в противном случае пусто).

Перейти в режим приложения и заполнить график в рамках календаря данного месяца.

Разработка регистра расчета

Особенностями данного регистра является периодичность, возможность применения вытеснения по периоду действия и зависимостью по базовому периоду.

*Периодичность* расчета определяет интервал, к которому будет относиться расчет. Независимо от того, когда будет в рамках указанной периодичности формироваться запись регистра программа всегда приведет значение Периода регистрации к началу того периода (в соответствии с периодичностью), в который он попадает. Например, периодичность – месяц, период регистрации всегда будет первое число месяца.

*Применение вытеснения по периоду действия*. То есть при появлении вида расчета с вытесняющим для данного действием (Невыход для Оклада) определяется фактический период действия, о котором говорилось выше (см. стр. 41).

*Зависимость по базовому периоду*. Позволяет осуществить расчет зависимых записей регистра на основании рассчитанных базовых. Если установлена зависимость по периоду действия, то для расчета будут выбираться те записи, у которых фактический период пересекается с базовым. Например, премия исчисляется из размера оклада за период. И фактический период расчета премии пересекается с периодом базовым, для которого рассчитывается оклад.

У регистра расчета могут существовать подчиненные объекты Перерасчет. Данные объекты предназначены для регистрации появления в регистре записей, которые влияют на значения уже существующих записей. То есть Перерасчет хранит информацию о том, какие записи подлежат перерасчету.

Необходимо создать новый регистр расчета с именем ЗП\_Расчет.

На вкладке *Основные*: план видов расчета – *ОсновныеНачисления, Период действии –* активизировать, график – регистр сведений *ГрафикиРабот,* значение – в ресурсе значение, дата – в измерении *Дата*. Базовый период – месяц.

Вкладка *Данные*: измерение – Сотрудник, ресурс – Результат (число), реквизит – *ГрафикРабот (*тип данных *– ВидыГрафиковРабот,* связь с графиком по измерению *График*), реквизит - *ИсходныеДанные* (число, длина 15). Реквизит исходные данные потребуется, чтобы хранить исходные данные, которые понадобятся при расчете или перерасчете.

Закладка *Перерасчеты*. Создать объект перерасчет, измерение *Сотрудник*, в установке связи указать измерении регистра *Сотрудник*, в качестве данных ведущих регистров также указать *Сотрудник*.

Несколько слов о виртуальном представлении регистра расчета, с чем разработчик может столкнуться при создании запросов:

| **Наименование виртуальной таблицы** | **Поля** | **Характеристика** |
| --- | --- | --- |
| *(ИмяРегистра)ДанныеГрафика*  Отображает сведения по периодам действия расчета - базового периода, фактического периода и др. Сведения могут быть представлены как в виде «крайних» дат периодов, так и дней в периоде. При расчете фактических периодов учитываются вытесняющие виды расчета. | ПериодРегистрации | Дата начала периода, соответствующего дате регистрации |
| Регистратор | Ссылка на документ |
| НомерСтроки | Номер строки записи регистра |
| ВидРасчета |  |
| ПериодДействия | Дата начала периода, соответствующего указанной дате начала периода действия вида расчета (по документу-регистратору) |
| ПериодДействияНачало | Дата начала периода действия вида расчета (по документу-регистратору) |
| ПериодДействияКонец | Дата окончания периода действия вида расчета (по документу-регистратору) |
| БазовыйПериодНачало | Для вторичных расчетов Дата начала базового периода |
| БазовыйПериодКонец | Для вторичных расчетов Дата окончания базового периода |
| *Измерения, Ресурсы…* |  |
| ЗначениеПериодДействия | Дней в периоде действия в соответствии указанным рабочим графиком (например, в месяце) |
| ЗначениеФактическийПериодДействия | Фактически дней для расчета в периоде (с учетом всех вытесняющих видов расчета) |
| ЗначениеБазовыйПериодДействия | Для вторичных расчетов - фактически дней в периоде для базовых расчетов (с учетом всех вытесняющих видов расчета) |
| *(ИмяРегистра)База((ИмяРегистра)*  Отображает информацию для вторичных расчетов. Определяет фактическую базу для расчета с учетом всех вытеснений в установленных измерениях | ПериодРегистрации |  |
| Регистратор |  |
| ……. |  |
| *Измерения, Ресурсы…* |  |
| РезультатБаза | Базовый результат первичных расчетов с учетом вытеснений (фактического базового периода). |

### Разработка процедур начислений сотрудникам

**Создание документа о начислении сотрудникам**

Создать документ *Начисления*. На вкладке Данные создать табличную часть *РасшифровкаНачислений*, которая включает следующие реквизиты: *Сотрудник, ГрафикРаботы, ДатаНачала, ДатаОкончания, ВидРасчета* (План видов расчета Основные начисления), *Результат* (Число).

Анализ алгоритма проведения расчетов

На Рис. 14Алгоритм расчета по окладу представлен алгоритм расчета составляющей по окладу. Особенностью представленных расчетов является то, что сначала документ Начисление проводится по регистру расчета (без расчета результата), а потом на основании созданных записей рассчитываются значений оклада. При этом учитывается как нормативный период, так и фактический период работы, так и вытесняющие расчеты. Результат заносится в регистр.

На Рис. 15 представлен алгоритм расчета составляющей по премии. Здесь учитываются только те сотрудники, у которых вид начисления – премия. При расчете используется база расчетов (указана для премии в плане видов расчета). Причем, не нормативная база, а с учетом фактического периода действия, то есть вытесняющих видов расчета.

Проведение по регистру расчета документа *Начисления* без заполнения ресурса *Результат:* формирование записей регистра, заполнение всех измерений и реквизитов на основании данных документа.

Сформировать список сотрудников, присутствующих в документе *Начисления*

Начисления

Определение Фактического периода действия и Нормативного периода (Значение период действия) действия по каждой записи регистра расчета

Регистр ЗП\_Расчет. Виртуальное представление ЗП\_РасчетДанные Графика

Регистратор

ВидРасчета – Оклад

Список сотрудников

*Расчет Оклада*

Для записей, совпадающих по номеру строки расчет составляющей по окладу и запись в соответствующие строки регистра

Регистр ЗП\_Расчет

Фактический и нормативный периоды для строк регистра, соответствующих документу Начисления

*Расчет премии*

Рис. Алгоритм расчета по окладу

Извлечь базу для расчета премии для тех записей регистра, которые относятся к документу *Начисления*

Регистр ЗП\_Расчет. Виртуальное представление ЗП\_РасчетБазаЗП\_Расчет

Регистратор

ВидРасчета – Премия

Для записей, совпадающих по номеру строки расчет составляющей по премии и запись в соответствующие строки регистра

База расчета строк регистра, соответствующих документу Начисления

Регистр ЗП\_Расчет

Рис. Фрагмент алгоритма расчета премии

В рассматриваемом примере процедуры расчетов пока используются только в документе начисления. Но в дальнейшем эти процедуры будут необходимы и в других документах (например, расчет больничных листов пойдет отдельным документом). Причем, сама процедура не меняется, меняются только исходные данные. Поэтому, основную процедуру есть смысл создавать в общем модуле конфигурации, а процедуру формирования передаваемых параметров – в модуле документа.

Проведение документа Начисления по регистру расчета

В документе Начисления вызвать Конструктор движений и сформировать движения документа. Вид движения представлен на Рис. 16.

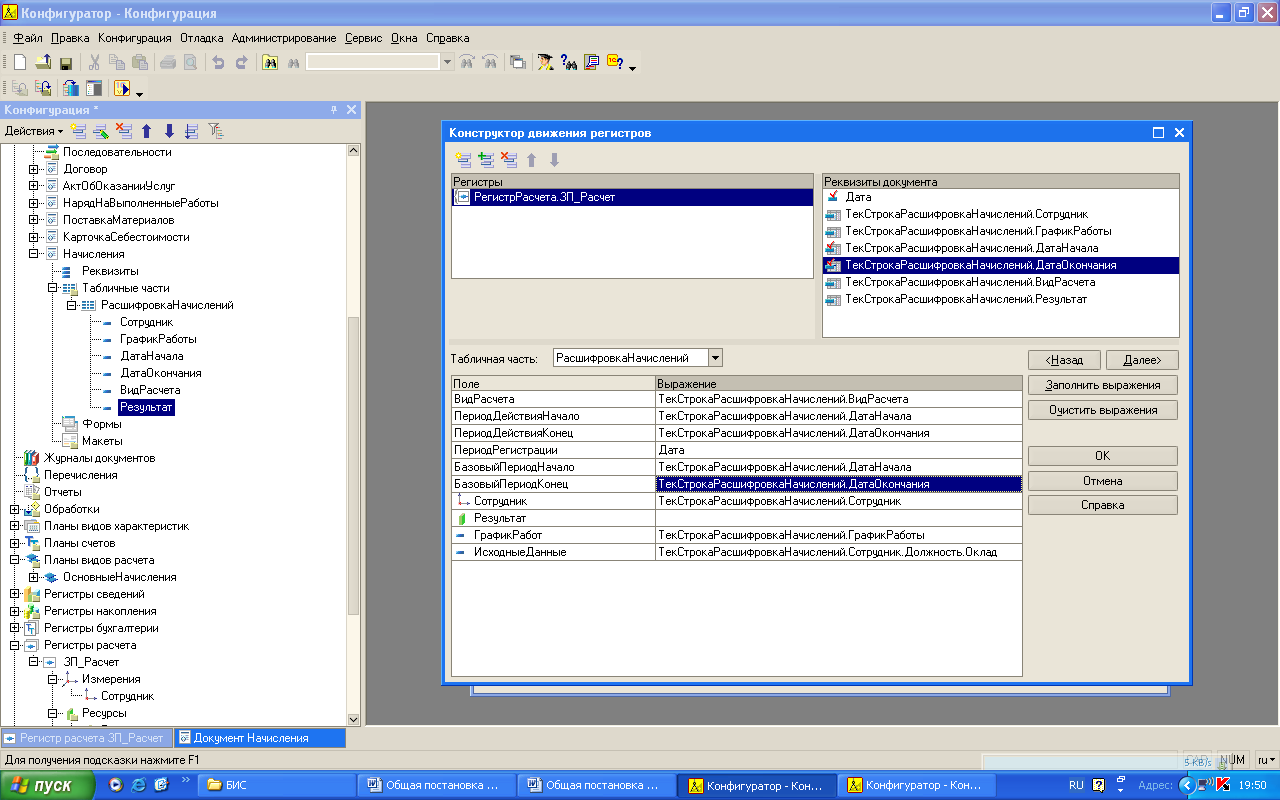


Рис. Движения документа Начисления в режиме конструктора

Обратить внимание, что не заполняется ресурс *Результат*. В качестве исходных данных берется Оклад для сотрудника из справочника должностей.

Создание процедур расчета в общем модуле

Создать новый общий модуль, имя – Расчеты. В модуле сформировать следующую процедуру.

|  |  |
| --- | --- |
| *Процедура Рассчитать(ЗаписиРегистра, ВидРасчета,СписокРаботников) Экспорт* | Начало процедуры. В скобках – входящие параметры, которые будут формироваться в документах, использующих процедуру |
| *Если ВидРасчета=ПланыВидовРасчета. ОсновныеНачисления.Оклад тогда* | Начало расчетов по *ВидуРасчета Оклад* |
| *Запрос=Новый Запрос;*  *Запрос.Текст=* | Организация нового запроса к регистру ЗП\_Расчет, виртуальное представление – Данные графика |
| Формирование запроса в режиме конструктора. Результатом выполнения запроса является: *ПоНорме* (нормативный период действия – дней в месяце), *ПоФакту* – фактический период действия с учетом вытесняющих расчетов для строк регистра (*НомерСтроки*). Ограничения (условия): по заданному виду расчета (оклад), по ограниченному списку сотрудников, анализируются записи регистра, соответствующие заданному документу (регистратору). | |
| *"ВЫБРАТЬ*  *| ЗП\_РасчетДанныеГрафика.ЗначениеФактическийПериодДействия КАК ПоФакту,*  *| ЗП\_РасчетДанныеГрафика.НомерСтроки,*  *| ЗП\_РасчетДанныеГрафика.ЗначениеПериодДействия КАК ПоНорме*  *|ИЗ*  *| РегистрРасчета.ЗП\_Расчет.ДанныеГрафика КАК ЗП\_РасчетДанные Графика*  *|ГДЕ*  *| ЗП\_РасчетДанныеГрафика.Регистратор = &Регистратор*  *| И ЗП\_РасчетДанныеГрафика.ВидРасчета = &ВидРасчета*  *| И ЗП\_РасчетДанныеГрафика.Сотрудник В(&СписокСотрудников)";* | |
| *Запрос.УстановитьПараметр("Регистратор", ЗаписиРегистра.Отбор.Регистратор. Значение);* | Установка параметра Регистратор. Будет соответствовать ссылке на регистратор для переданных Записей регистра (которые, в свою очередь будут соответствовать документу Начисления) |
| *Запрос.УстановитьПараметр ("ВидРасчета",ВидРасчета);* | Установка параметра *Вид расчета* (Оклад). |
| *Запрос.УстановитьПараметр ("СписокСотрудников",СписокРаботников);* | Установка параметра *СписокСотрудников*, который будет передан из документа при вызове процедуры. |
| *ВыборкаРезультата=Запрос.Выполнить().Выбрать();* | Выполнение запроса, результату выполнения присвоено имя – *ВыборкаРезультата* |
| *Для каждого ЗаписьРегистра из ЗаписиРегистра Цикл* | Начало цикла для перебора переданных записей регистра расчета |
| *СтруктураНомер=Новый Структура("НомерСтроки");* | Создание временной структуры, состоящей из одной колонки (*НомерСтроки*) |
| *СтруктураНомер.Номер Строки= Запись Регистра НомерСтроки;* | Передача в структуру, созданную на предыдущем шаге номера записи регистра (Номера Строки) |
| *ВыборкаРезультата.Сбросить();* | Позиционирование выборки на первую запись |
| *Если ВыборкаРезультата.Найти Следующий(СтруктураНомер) тогда* | Если номер записи в выборке совпадает с номером записи из переданных записей регистра, то переход к расчету по окладу |
| *Если ВыборкаРезультата.ПоНорме=0 тогда*  *Сообщить("Нет рабочих дней");*  *ЗаписьРегистра.Результат=0* | Если отсутствует результат по норме (для сотрудника не установлен график), то формирование нулевого результата |
| *Иначе ЗаписьРегистра.Результат= (ЗаписьРегистра. ИсходныеДанные/ ВыборкаРезультата. ПоНорме)\* ВыборкаРезультата.ПоФакту;* | Расчет фактического оклада |
| *КонецЕсли*  *КонецЕсли*  *КонецЦикла* | Завершения расчета по окладу |
| *ИначеЕсли ВидРасчета=ПланыВидовРасчета.ОсновныеНачисления.Премия тогда* | Начало расчетов по *ВидуРасчета Премия* |
| Формирование запроса в режиме конструктора. Результатом является значение базы по каждой строке регистра. Ограничения аналогичны предыдущему запросу. Добавляются ограничения по измерениям основного и базового регистра. В данном примере это один регистр, требуемое измерение - Сотрудник | |
| *Запрос=Новый Запрос;*  *Запрос.Текст=*  *"ВЫБРАТЬ*  *| ЗП\_РасчетБазаЗП\_Расчет.НомерСтроки,*  *| ЗП\_РасчетБазаЗП\_Расчет.РезультатБаза КАК База*  *|ИЗ*  *| РегистрРасчета.ЗП\_Расчет.БазаЗП\_Расчет(&ИзмеренияОсновной, &ИзмеренияБазовый, , ) КАК ЗП\_РасчетБазаЗП\_Расчет*  *|ГДЕ*  *| ЗП\_РасчетБазаЗП\_Расчет.Регистратор = &Регистратор*  *| И ЗП\_РасчетБазаЗП\_Расчет.ВидРасчета = &ВидРасчета*  *| И ЗП\_РасчетБазаЗП\_Расчет.Сотрудник В(&СписокСотрудников)";* | |
| *Измерение=Новый Массив(1);* |  |
| *Измерение[0]="Сотрудник";* |  |
| *Запрос.УстановитьПараметр ("ИзмеренияОсновной",Измерение);* | Установка параметров для запроса |
| *Запрос.УстановитьПараметр ("ИзмеренияБазовый", Измерение);* |
| *Запрос.УстановитьПараметр ("Регистратор",ЗаписиРегистра.Отбор.Регистратор.Значение);* |
| *Запрос.УстановитьПараметр ("ВидРасчета",ВидРасчета);* |
| *Запрос.УстановитьПараметр ("СписокСотрудников",СписокРаботников);* |
| *ВыборкаРезультата=Запрос.Выполнить().Выбрать();* | Смотри комментарии выше для расчета оклада |
| *Для каждого ЗаписьРегистра из ЗаписиРегистра Цикл* |
| *СтруктураНомер=Новый Структура("НомерСтроки");* |
| *СтруктураНомер.НомерСтроки=ЗаписьРегистра.НомерСтроки;* |
| *ВыборкаРезультата.Сбросить();* |
| *Если ВыборкаРезультата.Найти Следующий(СтруктураНомер) тогда* |
| *ЗаписьРегистра.Результат=ВыборкаРезультата.База\*0.25;* |
| *КонецЕсли* |
| *КонецЦикла* |
| *КонецЕсли* |
| *КонецПроцедуры* |  |

Разработка процедуры обработки проведения (расчета) в документе Начисления

В модуле документа начисления уже присутствует процедура обработки проведения документа, сформированная конструктором движений (см. Рис. 16). Необходимо в текст обработчика добавить вызов процедуры из общего модуля.

|  |  |
| --- | --- |
| *Запрос=Новый Запрос;*  *Запрос.Текст=* |  |
| Формирование запроса в режиме конструктора, который формирует список сотрудников в соответствии с текущим документом Начисления. | |
| *"ВЫБРАТЬ РАЗЛИЧНЫЕ*  *| НачисленияРасшифровкаНачислений.Сотрудник*  *|ИЗ*  *| Документ.Начисления.РасшифровкаНачислений КАК НачисленияРасшифровкаНачислений*  *|ГДЕ*  *| НачисленияРасшифровкаНачислений.Ссылка = &Ссылка";* | |
| *Запрос.УстановитьПараметр("Ссылка",Ссылка);* | Установка ссылки на текущий документ для передачи в запрос |
| *ТаблЗнач=Запрос.Выполнить().Выгрузить();* | Выполнение запроса и выгрузка в таблицу значений |
| *СписокРаботников=Новый СписокЗначений;* | Создание нового списка значений |
| *СписокРаботников.ЗагрузитьЗначения (ТаблЗнач.ВыгрузитьКолонку("Сотрудник"));* | Выгрузка в список значений списка сотрудников |
| *Рассчитать(Движения.ЗП\_Расчет, ПланыВидовРасчета.ОсновныеНачисления.Оклад, СписокРаботников);* | Вызов процедуры из общего модуля, передача входных параметров |
| *Рассчитать(Движения.ЗП\_Расчет,ПланыВидов Расчета.ОсновныеНачисления.Премия, Список Работников);* |
|  |
| *Движения.ЗП\_Расчет.Записать(Истина);* | Записать результаты расчета |

Процедура, связанная с перерасчетом записей регистра при изменении ведущих видов начислений будет сформирована позже.

Перейти в режим приложения и проверить работу процедур расчетов.

### Сторнирование записей. Расчет больничного.

Теперь можно рассмотреть правила расчета денег по больничному листу. Некоторые виды расчетов формируются «с опозданием» после проведения основных расчетов. Например, расчет больничного листа может осуществляться спустя один или несколько периодов после проведения расчета по окладу (Рис. 1). А на период расчета по окладу соответствующих ведений просто не было. Тут механизм вытеснения не работает, так как период регистрации больше, чем период действия. Например, за октябрь сотруднику был начислен оклад, а больничный поступил в расчет в ноябре. Период фактического действия больничного «покрыт» окладом, что неправильно, так как получается, что сотрудник получит и оклад, а потом еще и больничный. В таком случае необходимо сформировать сторно-запись (корректирующую запись), которая за фактический период больничного скорректирует оклад (выведет со знаком «-»). Сторно-запись также формируется в регистре расчета, указывается период регистрации, период действия сторно-записи, размер перерасчета.

Оклад

Больничный

Расчет прошлого периода

Период сторно-записи

Рис. Использование сторно-записи

Технология формирования сторно-записей следующая:

1. В модуле документа *Начисления* сформировать процедуру, которая проверяет, не конфликтует ли фактический период записей регистра расчета по данному документу с фактическими периодами более ранних записей.
2. Для конкурирующих записей создать сторно-записей и также разместить их в регистр расчета.
3. Скорректировать процедуру в общем модуле по расчету оклада для расчета сторнированных записей.

Процедура формирования сторно-записей в регистре расчета представлена ниже. Данный блок необходимо разместить в модуле документа после формирования записей регистра, до начала расчетов.

|  |  |
| --- | --- |
| *Набор=РегистрыРасчета.ЗП\_Расчет .СоздатьНаборЗаписей();* | Создать набор записей регистра расчета |
| *Набор.Отбор.Регистратор.Значение=Ссылка;* | Установить отбор записей, относящихся к текущему документу. |
| *Набор.Прочитать();* | Считать записи по установленному отбору |
| *ТаблСторно=Набор.ПолучитьДополнение();* | Метод ПолучитьДополнение() позволит проанализировать конфликты по фактическому периоду формируемого набора записей с записями по более раннему периоду регистрации. Возвращает таблицу значений, включающую все сведения для формирования сторно-записей |
| *Для каждого СтрокаСторно из ТаблСторно Цикл*  *Движение=Движения.ЗП\_Расчет. Добавить();*  *Движение.Регистратор=Ссылка;*  *Движение.ВидРасчета=Строка Сторно.ВидРасчета;*  *Движение.ПериодРегистрации= СтрокаСторно.ПериодРегистрацииСторно;*  *Движение.ПериодДействияНачало= СтрокаСторно.ПериодДействияНачалоСторно;*  *Движение.ПериодДействияКонец= СтрокаСторно.ПериодДействияКонецСторно;*  *Движение.ГрафикРаботы= СтрокаСторно.ГрафикРаботы;*  *Движение.Сотрудник=СтрокаСторно .Сотрудник;*  *Движение.ИсходныеДанные= СтрокаСторно.ИсходныеДанные;*  *Движение.Сторно=Истина;*  *КонецЦикла;* | Цикл для формирования сторно-записей в регистре расчета. Из таблицы значений *ТаблСторно* переносятся все параметры сторно в регистр. |
| *Движения.ЗП\_Расчет.Записать();* |  |

Расчеты по сторно (корректировка) рассчитываются также, как и обычные записи. В общем модуле необходимо сделать следующую корректировку в блоке расчета оклада:

*Если ВыборкаРезультата.НайтиСледующий(СтруктураНомер) тогда*

*Если ВыборкаРезультата.ПоНорме=0 тогда*

*Сообщить("Нет рабочих дней");*

*ЗаписьРегистра.Результат=0*

*Иначе*

***М=(ЗаписьРегистра.ИсходныеДанные/ВыборкаРезультата.ПоНорме)\*ВыборкаРезультата.ПоФакту;***

***ЗаписьРегистра.Результат=М\*?(ЗаписьРегистра.Сторно,-1,1);***

*КонецЕсли*

Тем самым, при формировании результата учитывается наличие признака Сторно=1 (истина). Если такой признак есть, то результат записывается со знаком «-».

**Самостоятельно**. В общем модуле сформировать блок расчета оплаты по больничному листу. Больничный оплачивается как 80% от оклада.

Формирование отчета по начислениям сотрудникам

Для просмотра результата расчетов есть смысл сформировать отчет (листки) по начислениям сотрудникам.

Создать новый отчет. Имя – *Начисления сотрудникам*. Вызвать конструктор макета отчета.

Исходный регистр – регистр *ЗП\_расчет*. Выбрать следующие поля: *Сотрудник, Вид расчета, ПериодДействияНачало, ПериодДействияКонец, Регистратор, Результат*.

На вкладке *Итоги* установить: группировочное поле – *Сотрудник*, итоговое поле - *Результат*, выражение – *Сумма*.

Завершить создание приложения, перейти в режим приложения, посмотреть формирование отчета. Вид отчета представлен на Рис. 18.

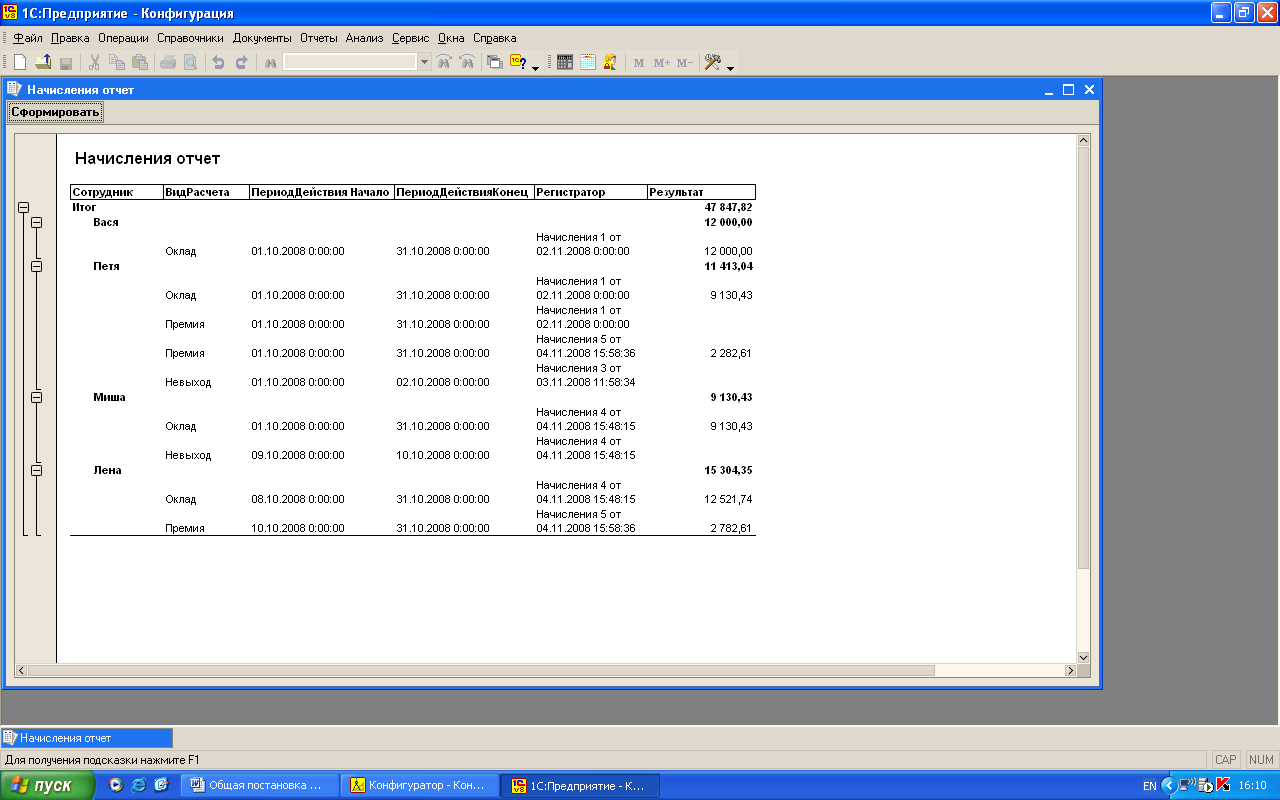


Рис. Примерный вид отчета «Начисления сотрудникам»

1. Если значению ресурса соответствует сумма с НДС, то удалить выражение, и двойным щелчком по соответствующему реквизиту документа (окошко Реквизиты документа) выводим требуемое значение [↑](#footnote-ref-1)
2. Обратить внимание на зеленый (+) около каждой записи, характеризующий вид движения. [↑](#footnote-ref-2)
3. Любопытно то, что 1С:Предприятие предлагает не только классический, англоязычный, но и русскоязычный вариант SQL [↑](#footnote-ref-3)