

Конфигурирование в системе  
"1С:Предприятие 8.0"

Решение задач бухгалтерского учета

*Методические материалы*

Конфигурирование в системе "1С:Предприятие 8.0"

**Решение задач бухгалтерского учета**

*Январь 2004г.*

## ПРАВО ТИРАЖИРОВАНИЯ ДОКУМЕНТАЦИИ ПРИНАДЛЕЖИТ ФИРМЕ «1С-Учебный центр №3»

Получив настоящие материалы для обучения, Вы тем самым  
даете согласие

не допускать их копирования без письменного  
разрешения фирмы «1С-Учебный центр №3»

© ООО «1С-Учебный центр №3», январь 2004 г.

Тел.:(095)681-17-03

Автор методических материалов - Козырев Д.В.

По вопросам совершенствования методических материалов  
просьба обращаться в фирму «1С-Учебный центр №3»

E-mail: [uc3@1c.ru](mailto:uc3@1c.ru).

# СОДЕРЖАНИЕ МАТЕРИАЛОВ

<b>ВВЕДЕНИЕ.....</b>	<b>5</b>
<b>ПОСТАНОВКА ЗАДАЧИ КУРСА.....</b>	<b>6</b>
<b>ТЕРМИНЫ И МЕТОДЫ БУХГАЛТЕРСКОГО УЧЕТА.....</b>	<b>7</b>
Виды учета.....	7
Предмет и метод бухгалтерского учета.....	9
Техника и формы бухгалтерского учета.....	12
Составим баланс вместе.....	13
Бухгалтерский учет: виды учета, степень детализации.....	21
Валюты в бухгалтерском учете.....	22
<b>МЕХАНИЗМЫ БУХГАЛТЕРСКОГО УЧЕТА В СИСТЕМЕ «1С:ПРЕДПРИЯТИЕ 8.0».....</b>	<b>25</b>
Регистры бухгалтерии.....	25
Синтетический учет.....	25
Аналитический учет.....	25
Консолидированный учет.....	26
Другие виды учета.....	26
<b>СИНТЕТИЧЕСКИЙ УЧЕТ.....</b>	<b>27</b>
План счетов.....	27
Регистр бухгалтерии.....	28
Ввод данных в регистр бухгалтерии.....	30
<i>Ручной ввод проводок.....</i>	<i>30</i>
<i>Формирование движений (проводок) в момент проведения документа.....</i>	<i>32</i>
<i>Регистр бухгалтерии как элемент табличной модели данных. Запрос по таблицам регистра бухгалтерии.....</i>	<i>36</i>
<i>Особенности работы со счетами, имеющими субсчета.....</i>	<i>38</i>
<i>Бухгалтерские отчеты.....</i>	<i>38</i>
<b>АНАЛИТИЧЕСКИЙ УЧЕТ.....</b>	<b>46</b>
План видов характеристик и виды субконто.....	46
Настройка плана счетов для ведения аналитического учета.....	47
Ввод данных по аналитике в регистр.....	48
<i>Документ Операция.....</i>	<i>48</i>
<i>Документ «Приходная накладная».....</i>	<i>49</i>
Отчеты по аналитике.....	49
<i>Движение товаров.....</i>	<i>50</i>
<i>Выручка в кассу — таблица «Обороты».....</i>	<i>52</i>
<b>КОЛИЧЕСТВЕННЫЙ УЧЕТ.....</b>	<b>54</b>
Изменения в плане счетов.....	54
Изменение в регистре.....	54
Изменения в документах.....	55
Обращение к результатам.....	55
<b>ВАЛЮТНЫЙ УЧЕТ.....</b>	<b>58</b>
Изменения в плане счетов.....	58
Изменения в регистре бухгалтерии.....	58
Изменения в документах.....	59
Получение результатов из регистров бухгалтерии.....	59
<b>КОНСОЛИДИРОВАННЫЙ УЧЕТ.....</b>	<b>61</b>
Изменения в регистре бухгалтерии.....	61
Изменения в документах.....	61
<i>Приходная накладная.....</i>	<i>61</i>
<i>Документ Операция.....</i>	<i>61</i>
Обращение к результатам.....	62

<b>ТРЕХВАЛЮТНЫЙ УЧЕТ</b>	<b>64</b>
РЕГИСТР БУХГАЛТЕРИИ	64
ИЗМЕНЕНИЯ В СПРАВОЧНИКАХ И АЛГОРИТМАХ РАСЧЕТА	64
ИЗМЕНЕНИЯ В ДОКУМЕНТАХ-ВВОД ДАННЫХ В ТРЕТЬЕЙ ВАЛЮТЕ	65
<i>Документ Операция</i>	66
<i>Прочие документы</i>	66
ОБРАЩЕНИЕ К РЕЗУЛЬТАТАМ	66
<b>РАСЧЕТ СЕБЕСТОИМОСТИ</b>	<b>69</b>
ПРОВЕДЕНИЕ ДОКУМЕНТА С АНАЛИЗОМ ДАННЫХ РЕГИСТРА	69
ПРОВЕДЕНИЕ ДОКУМЕНТОВ В РАЗЛИЧНЫХ ВЕРСИЯХ 1С:БУХГАЛТЕРИИ	69
1 С: Бухгалтерия 6.0	69
1 С:Предприятие 7.5	69
1С:Предприятие 7.7	70
1С: Предприятие 8.0	70
АЛГОРИТМ ПРОВЕДЕНИЯ В «1С:ПРЕДПРИЯТИЕ 8.0»	70
<i>Запрос по таблице документа</i>	71
<i>Запрос по таблице бухгалтерского регистра</i>	71
<i>Соединяем две таблицы в одну</i>	71
<i>Обработка проведения</i>	72
АЛЬТЕРНАТИВНЫЙ ВАРИАНТ ОБРАБОТКИ ПРОВЕДЕНИЯ ДОКУМЕНТА	74
ФОРМИРОВАНИЕ ДВИЖЕНИЙ РАЗНЫМИ ДАТАМИ	76
ОПЕРАТИВНОЕ И НЕОПЕРАТИВНОЕ ПРОВЕДЕНИЕ	77
<b>ПРИЗНАКИ УЧЕТА СУБКОНТО</b>	<b>79</b>
ТОЛЬКО ОБОРОТНОЕ СУБКОНТО	79
<i>Дополнительные признаки учета субконто</i>	80
<b>ТАБЛИЧНЫЙ ДОКУМЕНТ - УСОВЕРШЕНСТВУЕМ ОТЧЕТЫ</b>	<b>81</b>
РАСПОЛОЖЕНИЕ ТАБЛИЧНОГО ДОКУМЕНТА В КОНФИГУРАЦИИ	81
СВОДНАЯ ТАБЛИЦА ДЛЯ ПОСТРОЕНИЯ КРОСС-ОТЧЕТОВ	81
ПОСТРОИТЕЛЬ ОТЧЕТА	83
ГРУППИРОВКА	86
РАСШИФРОВКА	87
<i>Расшифровка отчета Оборота</i>	88
<i>Первый вариант решения</i>	88
<i>Второй вариант решения</i>	90
<i>Комментарии</i>	91
<b>САМОСТОЯТЕЛЬНАЯ РАБОТА</b>	<b>92</b>
<b>ЗАКЛЮЧЕНИЕ</b>	<b>94</b>



## Введение

Данный курс представляет собой вторую ступень изучения конфигурирования в системе «1С:Предприятие 8.0» и является продолжением курса «Введение в конфигурирование в системе «1С:Предприятие 8.0 Основные объекты».

Предполагается, что слушатель курса уже знаком с системой «1С:Предприятие 8.0» в объеме, изучаемом на курсе «Введение в конфигурирование в системе «1С:Предприятие 8.0». Основные объекты».

\*

Для выполнения курса требуется каркасная конфигурация, сопоставимая по составу объектов с заключительной конфигурацией, получаемой в результате изучения конфигурирования системы «1С:Предприятие 8.0» по названному выше курсу.

В данном курсе рассматриваются только объекты системы «1С:Предприятие 8.0», предназначенные для автоматизации задач бухгалтерского учета.

Основной целью учебного курса и данных методических материалов является освоение программы как инструмента по созданию собственных прикладных решений, развитие практических навыков по конфигурированию.

При подготовке сквозного примера, который лег в основу учебного курса, мы осознанно отошли от типовых конфигураций, с программным продуктом «1С Предприятие 8.0». Упростив задачу учета, мы стремимся пройти по всем этапам конфигурирования, добиваясь главной цели — получить навыки самостоятельной работы по созданию решений, предназначенных для ведения бухгалтерского учета, управления, анализа и контроля деятельности предприятия.

Так как на момент написания курса механизмы бухгалтерского учета находились в стадии бета-тестирования, то нельзя с полной уверенностью утверждать правильность предлагаемых подходов и то, что они сохранят свою актуальность при выходе последующих релизов системы «1С:Предприятие 8.0».





## Постановка задачи курса

Задача, решаемая в рамках данного курса предполагает использование в качестве каркасной конфигурации, конфигурацию, разработанную в рамках курса «Введение в конфигурирование в системе «1С:Предприятие 8.0». Основные объекты».

Задача - автоматизировать бухгалтерский учет операций основной деятельности, как одну из функций управления компанией, торгующей канцелярскими товарами. Конфигурация должна поддерживать следующие виды учета:

- Синтетический учет на счетах и субсчетах
- Аналитический учет в разрезе аналитических признаков
- Количественный учет (учет натуральных показателей)
- Валютный учет (называемый также многовалютным учетом)
- Консолидированный учет
- Трехвалютный учет (учет в «третьей» валюте - альтернативной валюте баланса).

При реализации описанных выше видов учета, потребуется рассмотреть вопросы расчета себестоимости списываемого товара на основании данных бухгалтерского учета.

## Термины и методы бухгалтерского учета

Сложность постановки и реализации задачи автоматизации бухгалтерского учета зачастую определяется недостатком понимания предметной области бухгалтерского учета сотрудниками, выполняющими конфигурирование. Результатом может быть нерациональное, а иногда и неправильное решение. Поэтому целесообразно иметь хотя бы поверхностное знакомство с бухгалтерской терминологией. Так как изучение теории бухгалтерского учета не является целью данного курса, то ниже рассматриваются лишь основные элементы, понимание которых представляется желательным. Более подробную информацию можно найти в соответствующей литературе, посвященной теории и практике бухгалтерского учета.

### Виды учета

По поводу правильной классификации видов учета вполне можно вести научные споры, что, однако, не входит в наши планы. Наша задача - увидеть перед собой «непечатый край» работы и понять основные особенности каждого вида учета с целью более правильного выбора механизмов его автоматизации в дальнейшем.

**Оперативный учет** - ограничивается участком выполнения работ. Используется для текущего (оперативного) управления предприятием. Пример: учет явки на работу, наличия материальных запасов и др. При его автоматизации решается локальная задача - получение (улучшение) оперативного управления ресурсами (трудовыми, денежными) на одном участке работ. Как правило, наиболее трудоемким для пользователя и важным (первоочередным) с точки зрения автоматизации является **оперативный учет основной деятельности** компании. Для торговой компании это складской учет, учет реализации и закупки товаров; для производственной - производственный учет (учет калькуляции себестоимости и др.).

Статистический учет - изучает явления, носящие массовый характер, использует данные оперативного и бухгалтерского учета.

Бухгалтерский учет - сплошное, непрерывное, взаимосвязанное отражение *всей* хозяйственной деятельности предприятия (на всех участках), на основании документов в различных измерителях, где обобщающим является денежный измеритель. Единый денежный измеритель позволяет учитывать и трудовые ресурсы (у сотрудника есть зарплата) и складские резервы (у товара есть себестоимость). В некоторых источниках выделяют из бухгалтерского учета подвиды: бухгалтерский управленческий, бухгалтерский финансовый и налоговый учет.

Бухгалтерский **управленческий учет** предназначен для формирования информации о деятельности предприятия для внутреннего пользования (собственников, руководителей, сотрудников). Учет ведется «для себя» и не регламентируется ничем, кроме желания директора (коммерческого директора, финансового директора, менеджера... в общем сотрудника, который заинтересован в повышении управляемости компании). Что, однако, не означает, невозможность использования каких-либо стандартов при разработке учетной политики управленческого учета. В основу могут быть положены как стандарты РФ, так и МСФО.

Бухгалтерский **финансовый учет** ориентирован на получение информации необходимой для составления бухгалтерской отчетности, подготовки информации для внешних пользователей (инвесторов, кредиторов, государственных контролирующих органов). Ведется в соответствии со стандартами бухгалтерского финансового учета и регламентируется законодательством (В РФ стандарты представлены ПБУ, за рубежом - GAAP, IAS и др.).

**Налоговый** учет также как и финансовый предназначен для составления «внешней» отчетности для налоговых органов, но ведется в соответствии с налоговым законодательством, которое может отличаться (ж отличается!) от стандартов бухгалтерского финансового учета. Место налогового учета в этой классификации спорно. До недавнего времени (до вступления в силу 25 главы НК РФ), налоговый учет можно было считать одним из подвидов бухгалтерского. Основная его задача - определение налоговой базы и исчисление налогов (в узком смысле - налога на прибыль) - решалась с использованием данных бухгалтерского учета (расчет налога на прибыль выполнялся на основании балансовой прибыли, скорректированной в соответствии с налоговым законодательством). Теперь вполне можно рассматривать налоговый учет как самостоятельный вид учета наряду с бухгалтерским учетом.

Бухгалтерский финансовый и налоговый учет могут быть объединены в **регламентированный** учет. Критерием объединения в данном случае будет являться то, что оба вида учета регламентированы государством (бухгалтерское и налоговое законодательство). Еще один критерий — степень охвата видами учета операций и сами операции. Наибольшую степень охвата, по нашему мнению, имеет бухгалтерский управленческий учет, в котором находят отражения все совершаемые хозяйственные операции. Финансовый и налоговый учет охватывают лишь часть операций, результаты которых могут быть показаны внешнему по отношению к компании пользователю и/или подлежат отражению в соответствии со стандартами учета.

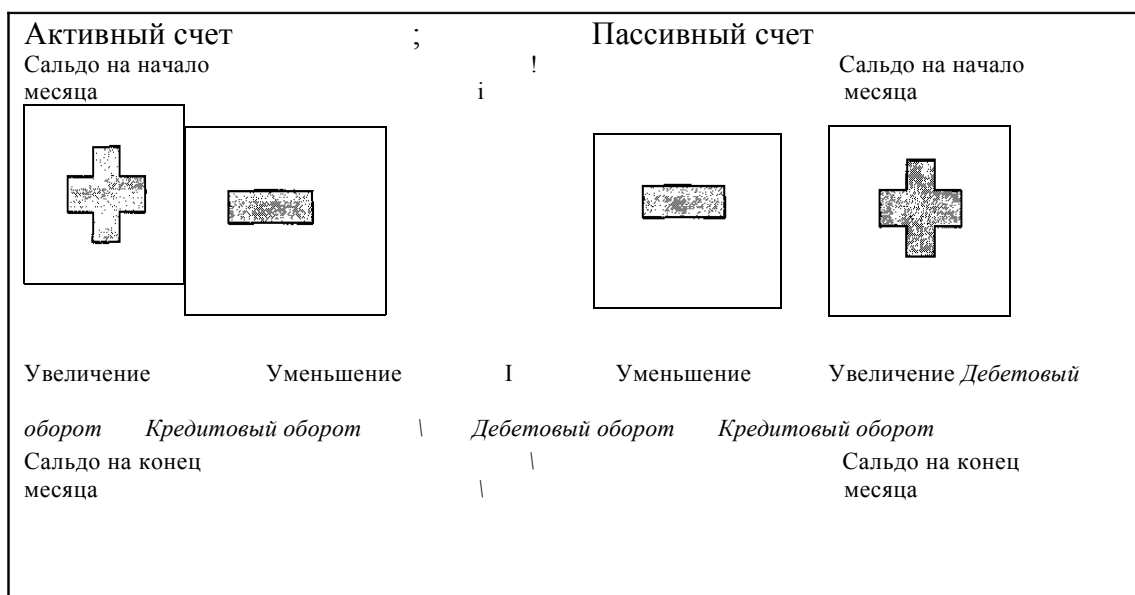
Задача, которую будем решать мы в рамках этого курса -автоматизация бухгалтерского управленческого учета, который будет вестись методом двойной записи на счетах бухгалтерского учета и будет охватывать деятельность всего предприятия в целом.

## Предмет и метод бухгалтерского учета

Предметом бухгалтерского учета в обобщенном виде выступает вся хозяйственная деятельность предприятия. Методы, которые используются для его ведения: инвентаризация (проверка фактического наличия), документация (письменной свидетельство) и двойная запись на счетах.

Счет - таблица с двусторонней записью, предназначенная для группировки однородных хозяйственных операций.

Делятся на активные и пассивные. Свое название получили от названия сторон баланса. Активные счета предназначены для учета хозяйственных средств по их составу и размещению, пассивные — для учета источников хозяйственных средств по их целевому назначению. Каждый счет - таблица с двусторонней записью. Левая сторона -дебет, правая - кредит.



Кроме активных и пассивных счетов, сальдо которых predetermined, существуют счета активно-пассивные (по крайней мере, в бухгалтерском учете по стандартам РФ). Сальдо по ним может быть, как дебетовым, так и кредитовым и зависит от оборотов за период. В зависимости от этого они попадают в соответствующую часть баланса (свернуто или развернуто по субсчетам или объектам аналитического учета).

Хозяйственная операция - отражение в бухгалтерском учете события в хозяйственной жизни предприятия. В нашей компьютерной бухгалтерии одна хозяйственная операция будет соответствовать группе проводок одного документа, который в свою очередь является электронным аналогом первичного документа в бухгалтерии. Так же как каждый документ может содержать в себе более детальную информацию, так и операция может содержать в себе одну или несколько «строк» - проводок по счетам бухгалтерского учета.

Проводка, как правило, содержит в себе две стороны - дебет и кредит. При такой организации учета, каждая сумма обязательно находит отражение по дебету одного счета и по кредиту другого, что и принято называть «двойной записью». Правило «двойной записи» демонстрирует - «ничто не берется из ниоткуда и не уходит в никуда». Если где-то чего-то стало больше, значит где-то должно и убыть, причем в том же объеме.

Возможна и альтернативная организация учета, применяемая в западных системах учета, при которой сумма проводки может отражаться по дебету одного (или нескольких счетов) и по кредиту одного (или нескольких счетов). При такой организации учета каждая операция может содержать несколько зависимых проводок, каждая из которых содержит движение только по одному счету (дебету или кредиту), что, однако, не нарушает правила «двойной записи», которое поддерживается для всех записей одной операции - сумма всех дебетовых записей должна соответствовать сумме всех кредитовых.

Выбор схемы учета при его автоматизации (с поддержкой корреспонденции - два счета в проводке, или без поддержки корреспонденции) зависит от поставленной задачи. «1С:Предприятие 8.0» позволяет реализовать обе учетные схемы.

Исключение из правила двойной записи составляют только забалансовые счета. При формировании движения (проводки) по забалансовому счету не требуется указывать корреспондирующий счет (при выборе схемы учета с поддержкой корреспонденции) или формировать еще одну зависимую запись с противоположным типом движения (если выбрана схема без поддержки корреспонденции). На забалансовых счетах учитывается, например, имущество, которое не является собственностью организации. Это могут быть товары, принятые на комиссию (которые продолжают считаться собственностью комитента), арендованные основные средства и др.

Двойную запись на счетах (т.е. бухгалтерские проводки) принято показывать или «строчками» или «самолетиками». Опытные бухгалтера-практики обычно записывают проводку строчкой, хотя рисование проводки «самолетиками», на наш взгляд, может сильно помочь разобраться в новой учетной схеме. Самолетики позволяют

увидеть движения и остатки средств на счетах. Ниже приведены оба варианта отражения проводок.

Проводки отражают две хозяйственные операции:

- 1.Обналичиваем средства - снимаем с банковского счета в кассу.
- 2.Выдаем деньги сотруднику в подотчет.

Проводки «самолетиками»			Проводки «строчками»			
Банк	Касса	Сотрудник	Дебет	Кредит	Сумма	Содержание
100			Касса	Банк	10	Обналичка
	Обналичка		Сотрудник	Касса	7	В подотчет
	10	В подотчет				
		7				
0	10	10	7	7	0	
90	3	7				

Обычно у слушателя, который раньше не сталкивался с бухгалтерским учетом, при виде самолетиков возникает вопрос: если мы все время рисуем стрелки слева направо, значит, где-то слева эта цепочка должна заканчиваться? Если бы все счета были активными (а у нас на рисунке Касса и Банк - активные счета), то так бы оно, наверное, и было. Но есть и пассивные счета (счета источников средств). Поэтому первая проводка на любом предприятии затрагивает все-таки два счета - дебет счета активного (например, средств, или задолженности учредителей по вкладам в уставный капитал) и кредит пассивного (капитал). Т.е. первая проводка увеличивает сразу два счета. И, как результат, мы получаем первый, вступительный, баланс фирмы.

По способу группировки и обобщения счета делятся на синтетические и аналитические.

■ Синтетические счета предназначены для укрупненной группировки информации в едином денежном измерителе.

■ Аналитические счета хранят детальные, частные сведения. Аналитический счет раскрывает синтетический в разрезе его видов, частей, статей и, где это требуется, с оценкой информации в натуральном, трудовом и денежном выражении.

Взаимосвязь синтетических и аналитических счетов: сальдо одного синтетического счета должно быть равно сумме сальдо всех аналитических счетов, к нему относящихся. То же касается оборотов.

Все используемые на предприятии счета объединяются в план счетов.

Результирующая информация о деятельности компании может быть предоставлена, пользователю в виде отчетных форм, содержащих показатели, полученные путем группировки данных на счетах бухгалтерского учета. К основным отчетным формам можно отнести баланс, отчет о прибылях и убытках, отчет о движении денежных средств, отчет о дебиторской и кредиторской задолженности, отчет об изменении капитала собственника и др. Главная из них - баланс, отчет который базируется на двойной записи.

Баланс в бухгалтерском учете рассматривается как способ обобщения и группировки хозяйственных средств предприятия и их источников на определенную дату. По своему строению представляет собой двустороннюю таблицу, левая сторона которой - актив - отражает состав и размещение хозяйственных средств, а правая - пассив - отражает источники образования хозяйственных средств и их целевое назначение. В балансе должно быть обязательное равенство итогов: сумма всех статей актива баланса должна быть равна сумме всех статей пассива. Сумму статей актива/пассива называют так же валютой баланса.

Полнота и степень детализации этих отчетов зависит от степени охвата учетом деятельности компании (управленческий или финансовый учет).

## Техника и формы бухгалтерского учета

Под техникой бухгалтерского учета понимается способ его ведения, т.е. регистрация учетных записей. Для этого в «бумажной бухгалтерии» предназначены учетные регистры (карточки, сводные листы, книги). Под формой бухгалтерского учета принято понимать определенное сочетание построения и взаимосвязи регистров, а также порядок и способы регистрации и группировки в них учетной информации. На практике до нас «дожили» несколько форм бухгалтерского учета: журнально-ордерная, мемориально-ордерная (в основном бюджетные организации), журнал-главная (для небольших предприятий)

Нам эта тема может быть интересна только с точки зрения лучшего понимания заказчика, потому что сами мы будем использовать только одну форму учета - машинно-ориентированную (или компьютерную).

Отличаются формы в первую очередь набором учетных регистров - различного вида таблиц, в которые записываются данные из первичных документов. Наиболее часто используемая классификация регистров: по назначению. По назначению они делятся на:

- хронологические, куда факты записываются по мере их возникновения без иной систематизации;

- систематические, которые ведутся в разрезе группировочных признаков;
- комбинированные (сочетает в себе первые два варианта).

Компьютерная форма учета отличается тем, что учетные регистры строятся на основании таблиц базы данных и теряют часть своих функций, а именно - группировку и контроль. Группировкой занимается компьютер, а контроль правильности расчетов становится не нужен, так как в арифметике компьютер не ошибается. Единственная их функция - представить информацию в удобном для бухгалтера виде.

В некоторых случаях Вам придется убеждать бухгалтера, что данная форма отчета не нужна, так как нет необходимости проверять компьютер при подсчете итогов по тем или иным счетам. А те формы (отчеты), которые есть в программе, полностью справляются с «возложенными на них обязанностями».

## Составим баланс вместе

Давайте попробуем закрепить все полученные знания о теории бухгалтерского учета на практике.

Задача, Некто Иванов Иван Иванович., хорошо освоивший (не без помощи знаний, полученных на наших курсах!) «1С:Предприятие», решил организовать собственный бизнес по оказанию услуг внедрения этого программного продукта. Само собой, ему потребовалось вести учет хозяйственной деятельности своего, пусть очень маленького, но, все-таки, предприятия. И, как и положено «сапожнику без сапог», решил он вести бухгалтерский учет своего предприятия на бумаге. В нашу с Вами задачу входит помочь ему составить баланс за первый месяц деятельности его предприятия.

Дальше по тексту Иванов И.И. приводит нам свои хозяйственные операции, которые он совершил в течение первого месяца работы в качестве внедренца. Мы должны оформить эти операции в проводки, систематизировать и, в конечном итоге, составить баланс.

### 1. Инвестиция в бизнес

Любой бизнес начинается с инвестиций в него определенной суммы денег - начального капитала. Иванов мог позволить себе вложить в дело 100=00 денег.

1 Инвестиция средств в бизнес.		
Счет дебета	Счет кредита	Сумма
Наличные	Капитал	100=00

В результате первой операции у нас появляются два счета: «Наличные» и «Капитал». Счет средств «Наличные» активный и в результате операции увеличивается на сумму операции. Счет





источника средств «Капитал» (капитал собственника) - пассивный, тоже увеличивается на сумму операции.

В нашем примере понятие «Счет» и «Статья баланса» - одно и то же. В реальном бухгалтерском учете статья баланса может аккумулировать информацию нескольких синтетических счетов.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	100=00	Капитал	100=00
Всего актив	100=00	Всего пассив	100=00

## 2. Приобретение офиса

Первым делом наш бизнесмен купил себе офис, который обошелся ему в 60=00.

2	Покупка офиса за наличные деньги		
	Счет дебета	Счет кредита	Сумма
	Офис	Наличные	60=00

В нашей бухгалтерии появился еще один активный счет «Офис», на котором мы будем учитывать стоимость долгосрочного актива. В результате этой операции, в которой участвуют два активных счета, мы увеличили счет «Офис» и уменьшили счет «Наличные» на сумму потраченных на покупку офиса денег.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	40=00	Капитал	100=00
Офис	60=00		
Всего актив	100=00	Всего пассив	100=00

## 3. Приобретение мебели в долг

Иванов договорился с поставщиком мебели о покупке мебели для своего офиса в долг.

3	Покупка мебели в долг		
	Счет дебета	Счет кредита	Сумма
	Мебель	Обязательства к выдаче	15=00

В учете появляются два новых счета. «Мебель» - активный счет, на котором будем учитывать стоимость мебели в нашем офисе. «Обязательства к выдаче» (он же: расчеты с поставщиками, счета к оплате, короче: «мы должны» или «кредиторка», как говорят бухгалтера) - пассивный счет, который будет увеличиваться по мере роста нашего долга перед поставщиками и уменьшаться, когда мы будем платить по долгам.

Все счета, о которых идет речь - синтетические. Пример аналитического счета: мы могли бы разбить счет «Обязательства к выдаче» в разрезе аналитического счета

«Поставщики», содержащего список наших контрагентов. Тогда мы могли бы получать информацию о том, сколько мы должны каждому поставщику. На синтетическом счете «Обязательства к выдаче» хранится информация нашего долга перед всеми поставщиками.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	40=00	Капитал	100=00
Офис	60=00	Обязательства к выдаче	15=00
Мебель	15=00		
Всего актив	115=00	Всего пассив	115=00

#### 4. Оказал услуги по настройке программы за наличные

Нашелся первый клиент, которому нужно было что-то в программе донстроить. Иванов договорился с ним об оплате наличными сразу по окончании работы.

4 Оказал услуги за наличные деньги		
Счет дебета	Счет кредита	Сумма
Наличные	Капитал	10=00

В результате у Иванова стало больше денег в кассе, и вырос капитал его фирмы.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	50=00	Капитал	110=00
Офис	60=00	Обязательства к выдаче	15=00
Мебель	15=00		
Всего актив	125=00	Всего пассив	125=00

#### 5. Оказал услуги по настройке программы в долг

Второй клиент уговорил нашего внедренца настроить ему программу в долг. Оплатит - потом, когда-нибудь, когда деньги появятся.

5 Оказал услуги в долг		
Счет дебета	Счет кредита	Сумма
Обязательства к получению	Капитал	5=00

Денег больше не стало. Но зато появилась расписка покупателя в том, что он обязуется эти деньги вернуть, а это тоже актив. Хотя и менее ликвидный, чем наличные средства (которые можно сразу потратить, степень ликвидности именно так и определяется: самые ликвидные - наличные деньги, менее ликвидные безналичные, еще менее ликвидные - долги, совсем не ликвидные - запасы, которые сначала надо продать, потом долг получить, а только потом потратить!). Капитал тоже увеличился на сумму этой «филькиной грамоты». Саму «филькину грамоту» будем учитывать на счете «Обязательства к



получению» (он же: расчеты с поставщиками, счета к получению, короче: «нам должны» или «дебиторка», как говорят бухгалтера).

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	50=00	Капитал	115=00
Офис	60=00	Обязательства к выдаче	15=00
Мебель	15=00		
Обязательства к получению	5=00		
<b>Всего актив</b>	<b>130=00</b>	<b>Всего пассив</b>	<b>130=00</b>

## 6. Зарплата наемному работнику

В течение всего этого времени в офисе работала, ну, например, уборщица. Ей пришлось заплатить зарплату.

6 Зарплата наемному работнику		
Счет дебета	Счет кредита	Сумма
Капитал	Наличные	3=00

Денег стало меньше, и капитал уменьшился. Деньги ушли из бизнеса.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	47=00	Капитал	112=00
Офис	60=00	Обязательства к выдаче	15=00
Мебель	15=00		
Обязательства к получению	5=00		
<b>Всего актив</b>	<b>127=00</b>	<b>Всего пассив</b>	<b>127=00</b>

## 7. Отдал часть долга поставщику мебели

Как это ни неприятно, но долги иногда приходится отдавать.

7 Погасил часть долга поставщику мебели		
Счет дебета	Счет кредита	Сумма
Обязательства к выдаче	Наличные	9=00

Уменьшился долг перед поставщиком и наличные в кассе

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	38=00	Капитал	112=00
Офис	60=00	Обязательства к выдаче	6=00
Мебель	15=00		
Обязательства к получению	5=00		
<b>Всего актив</b>	<b>118=00</b>	<b>Всего пассив</b>	<b>118=00</b>

**8. Решил потратить свои деньги, в бизнес не инвестированные**

Если деньги в бизнесе не участвуют, то и проводки в бухучете предприятия делать не надо. Потрачены деньги, которые не были инвестированы в бизнес.

**9. Получил долг с клиента**

Как это ни странно, но долг нашему внедренцу отдали. Причем полностью.

9 Клиент отдал долг за оказанные ранее услуги		
Счет дебета	Счет кредита	Сумма
Наличные	Обязательства к получению	5=00

Денег стало больше, а задолженности перед нами - меньше. Оба счета активные - валюта баланса (итоги актива, пассива) не меняется.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	43=00	Капитал	112=00
Офис	60=00	Обязательства к выдаче	6=00
Мебель	15=00		
Обязательства к получению	0=00		
Всего актив	118=00	Всего пассив	118=00

**10. Продал часть офиса**

Иванов решил, что офис для него слишком большой, а денег, как известно, слишком много не бывает, и решил часть офиса продать (скажем, одну из двух комнат). Сделка состоялась, продаваемая часть офиса был оценена в 30=00.

10 Продажа части офиса за наличные		
Счет дебета	Счет кредита	Сумма
Наличные	Офис	30=00

Денег стало больше, а офиса - меньше, Как и в предыдущем случае, валюта баланса не меняется.

Продал он его по себестоимости, т.е. без прибыли, т.е. были две одинаковые совершенно комнаты купленные вместе за 60, теперь одну из них продали... Если бы прибыль возникла, проводка (а точнее, проводки) отличались бы от приведенной выше.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	73=00	Капитал	112=00
Офис	30=00	Обязательства к выдаче	6=00
Мебель	15=00		
Обязательства к получению	0=00		
Всего актив	118=00	Всего пассив	118=00



### 11. Изъял часть денег из бизнеса на личные нужды

Понадобились ему деньги на собственные расходы, на что-то и жить надо.

11	Изъял часть наличных из бизнеса на собственные нужды	
	Счет дебета	Счет кредита
Капитал	Наличные	40=00

Денег стало меньше, и капитал фирмы уменьшился.

БАЛАНС			
АКТИВ		ПАССИВ	
Статья	Сумма	Статья	Сумма
Наличные	33=00	Капитал	72=00
Офис	30=00	Обязательства к выдаче	6=00
Мебель	15=00		
Обязательства к получению	0=00		
<b>Всего актив</b>	<b>78=00</b>	<b>Всего пассив</b>	<b>78=00</b>

### Результат

В результате деятельности фирмы за первый месяц мы получили баланс со следующими цифрами, отражающими состояние нашего бизнеса. Все средства, которые участвуют в бизнесе, были или инвестированы владельцем фирмы (Ивановым - «Капитал»), или получены в долг («Обязательства к выдаче»). Средства представлены в виде наличных денег, вложены в офис и мебель. На момент составления баланса нам никто не должен.

### Практикум № 1

*Составьте оборотный баланс деятельности Иванова ИИ.*

*Как выглядит оборотный баланс? Таблица, строки которой счета, а колонки: сальдо на начало (дебетовое и кредитовое), обороты за период (дебетовые и кредитовые), сальдо на конец (дебетовое и кредитовое). Отличие от баланса, который уже приведен в наших конспектах - наличие оборотов (сальдо на начало у нас нет — предприятие начало свою деятельность в текущем отчетном периоде). Ваша задача составить оборотный баланс и проверить баланс, который мы получили к концу примера.*

*Итак, вспомним, какие операции были в течение месяца работы на фирме.*

№	Счет дебета	Счет кредита	Сумма	Комментарий
1.	Наличные	Капитал	100=00	Инвестиция
2	Офис	Наличные	60=00	Купил офис
3	Мебель	Обязат к выдаче	15=00	Купил мебель в долг



4,	Наличные	Капитал	10=00	Оказал услуги за нал.
5.	Обязат. к получению	Капитал	5=00	Оказал услуги в долг
6.	Капитал	Наличные	3=00	З.п. работнику
7.	Обязат. к выдаче	Наличные	9=00	Отдал долг поставщику
8.				Нет проводки.
9.	Наличные	Обязат. к получению	5=00	Получ. долг с клиента
10.	Наличные	Офис	30=00	Продал часть офиса
11.	Капитал	Наличные	40=00	Взял деньги из бизнеса

Для решения поставленной задачи воспользуйтесь комбинированным регистром учета - формой «Журнал-главная». Она лучше всего подходит для решения задачи учета маленькой фирмы. Часть этого регистра (уже заполненная) приведена ниже.

Порядковый номер операции, содержание

Наличные

На начало	0=00	
1. Инвестиция	100=00	60=00  3=00 9=00  40=00
2. Купил офис		
4. Оказал услуги за наличные	10=00	
6. Выплатил зарплату сотруднику		
7. Отдал долг за мебель		
9. Получил долг с заказчика	5=00	
10. Продал офис	30=00	
11. Изъял деньги из бизнеса		
<b>Обороты</b>	<b>145=00</b>	<b>112=00</b>
<b>На конец</b>	<b>33=00</b>	

Не забываем, каждая сумма в бухгалтерском учете проходит по двум счетам - дебету одного и кредиту другого. Это значит, если по дебету кассы прошло 100=00, значит, эти же 100=00 должны быть учтены по кредиту другого счета (в данном случае «Капитал»).

Когда все счета (колонки формы) заполнены, подводим итог по каждому - обороты и сальдо, и записываем их в оборотку.

Оборотный баланс предприятия Иванова И.И. за XX месяц 20XX г.						
Счет (статья)	Сальдо на начало		Обороты		Сальдо на конец	
	Дебет	Кредит	Дебет	Кредит	Дебет	Кредит
Наличные	0=00		145=00	112=00	33=00	
Офис	0=00					
Мебель	0=00					



Обязат. к получению	0=00					
Капитал		0=00				
Обязат. к выдаче		0=00				
<b>Итого:</b>	<b>0=00</b>	<b>0=00</b>				

*Проверка: дебетовый оборот по всем счетам должен быть равен кредитовому (ведь любая сумма проходила сразу по двум счетам - по дебету одного и по кредиту другого), и сальдо на конец дебетовое по всем счетам должно быть равно кредитовому и валюте баланса на предыдущей странице.*

*Пустая форма «Журнал-главной» для решения этого примера приводится в приложении.*

## Бухгалтерский учет: виды учета, степень детализации

Единственным обязательным видом учета в бухгалтерии является синтетический учет на счета и субсчетах в денежном выражении в валюте учета. Будем считать, что все суммы, которые проходят по счетам бухгалтерского учета, отражаются в этой валюте. Для финансового и налогового учета в РФ валютой учета является национальная валюта (рубль) и именно в этой валюте подготавливается баланс и другие сводные бухгалтерские отчеты. Отдельные виды средств предприятия требуют хранения информации и в количественном (натуральном) выражении. Это все (или почти все) материальные средства - материалы, товары, продукция, и т.д. Учет в натуральном выражении подразумевает, что на отдельных счетах (не всех, а только нужных - не ясно, например, что можно учитывать в натуральном выражении в кассе: количество монет или бумажек :-)) мы будем хранить информацию по еще одному виду учета - количественному учету.

Отдельные счета, связанные с учетом денежных средств, обязательств в разных валютах требуют хранения сумм не только в валюте учета, но в других валютах (отдельно по каждой валюте) и эквивалента в валюте учета. Например, в кассе могут лежать разные валюты (если сделки совершаются за разные валюты), тогда необходимо помнить, сколько в кассе рублей, сколько долларов, сколько евро и какова сумма эквивалента в валюте учета по каждой валюте. При этом эквивалент в валюте учета будет рассчитываться как сумма иностранной валюте, умноженная на курс этой валюты к валюте учета.

Отдельно следует рассмотреть задачу ведения учета в «третьей валюте» («первая» валюта — валюта учета, «вторая» — валюта сделки при ведении мультивалютного учета). Так, например, третьей валютой может быть валюта холдинга, которая может отличаться от валюты учета (валютой учета, валютой составления баланса для финансового учета в РФ является рубль, а валютой холдинга может быть доллар США или евро). В отличие от валюты сделки при мультивалютном учете, в третьей валюте должен формироваться баланс, что подразумевает необходимость отражения всех операций на всех счетах, как в валюте учета, так и валюте холдинга.

Степень детализации хранимых данных может не ограничиваться счетами и субсчетами, но иметь также и дополнительные разрезы. Если дополнительные разрезы не нужны, такой учет (и счет на котором он ведется) называют синтетическим. В таком случае мы можем анализировать лишь сводные обороты и остатки по счету в целом.

Если нужно детализировать какой-то из счетов статьям, номенклатуре товаров или контрагентов, другим признакам, то такой вид учета



ведется дополнительно к синтетическому и называется аналитическим. Ведение списков аналитических признаков в системе «1С:Предприятие 8.0» реализуется с помощью понятия «вид субконто», отдельный объект аналитического учета называется «субконто».

В отличие от субсчетов, учет на которых также детализирует счет в целом, учет на аналитических счетах (видах субконто) может вестись параллельно по нескольким аналитическим срезам (например, товары и склады: один и тот же товар может лежать на разных складах и, наоборот, на одном складе могут лежать товары разных видов).

При определении необходимой детализации учета также следует учитывать необходимость получения всей отчетности (а не только отчета по выбранному счету, как в случае ведения аналитического учета) в разрезе некоего признака. В этом случае признак разделения записей в учете относится не к конкретному (одному) счету проводки, а к проводке в целом, что и обеспечивает возможность построения сводной отчетности, включая баланс. Пример - учет холдинга в разрезе входящих в него компаний, учет предприятия в разрезе центров учета (центров финансовой ответственности), проектов и др. Подобная возможность была реализована в системе «1С:Предприятие 7.7» с помощью разделителя учета.

## Валюты в бухгалтерском учете

Чтобы не запутаться во всех этих «первых, вторых и т.д.» валютах попробуем определиться с терминологией.

Каждая страна имеет свою **национальную валюту**. Для РФ это рубль. Именно по отношению к этой валюте заводятся курсы для других валют. Когда мы смотрим новости, или проходим мимо «обменника» и видим цифру «30» напротив валюты USD, то это означает, что сейчас можно купить (продать) один доллар за 30 рублей. Таким образом, будем называть валюту, относительно которой указывается курс остальных - **национальной (или базовой) валютой**.

Бухгалтерский учет, как мы помним, ведется в денежном измерителе. Именно он (единый денежный измеритель) и позволяет сравнивать различные активы и обязательства организации. Если активы выражать в одной валюте, а обязательства — в другой, навряд ли их можно будет сравнить. Валюту единого денежного измерителя будем называть **валютой учета**. Для бухгалтерского финансового учета в РФ валютой учета является национальная валюта. Для управленческого учета, как правило, выбирается та, курс которой более стабилен. До недавних пор это был Доллар США. Последнее время в качестве валюты учета все чаще выбирается Евро.

Многовалютный учет подразумевает оценку активов (обязательств) и регистрацию хозяйственных операций не только в валюте учета, но и в других валютах. При этом в учете нужно отразить и сумму в валюте сделки (проводки, операции, документа...) и ее эквивалент в валюте учета. Пересчет производится через установленный на день сделки (проводки, операции, документа) курс. Курс - это то, на что надо умножить сумму в валюте сделки, чтобы получить эквивалент в валюте учета. Для валют с очень маленьким значением курса введено понятие кратности. Кратность - то, на что надо поделить сумму в валюте сделки, чтобы получить сумму в валюте учета.

Трехвалютный учет подразумевает отражение всех хозяйственных операции в двух валютах учета. Таким образом, мы по-прежнему отражаем все операции в валюте учета, которая была выбрана в качестве единого денежного измерителя. И параллельно храним данные о совершенной операции еще в одной валюте, которая также является единым измерителем, но уже не для нас, а, например, для хозяев холдинга, в который входит наша организация. Пример: организация «Сигма» ведет свой регламентный учет, как и положено в рублях (в соответствии с законодательством РФ). Т.е. валюта учета - «рубли». Кроме того, входя в холдинг, хозяева которого - иностранцы, должна составлять аналогичную российской по составу показателей (упростим задачу) отчетность в евро. Тогда евро — та самая «третья» валюта.

И при пересчете валюты сделки (многовалютный учет) в валюту учета и при пересчете валюты сделки или валюты учета в «третью» валюту используются понятие курса. Как мы уже говорили - курс это количество единиц национальной валюты за одну единицу валюты иностранной. Если одна единица валюты национальной равна нескольким единицам валюты иностранной (1 рубль = 1000 тугриков), то вводится понятие **кратности**. Так, например, для валюты «Тугрики» курс 1 кратность 1000. Если за 3 рубля дают 1000 тугриков, то курс Тугриков к рублю 3, а кратность 1000. То же самое можно записать в виде одного курса - курс 0,003. В том случае, если валюта национальная не равна валюте учета, и необходимо выполнить пересчет из валюты сделки в валюту учета, вводится понятие **кросс-курса**. **Кросс курс** - отношение курса валюты сделки к курсу валюты учета. Пример: национальная (базовая, относительно которой вводятся курсы в справочник валют) валюта - рубль. Валюта учета - доллар. А валюта сделки (документа, проводки, операции) - евро...

Валюта	Курс	Кратность
Рубль	1	1
Доллар США	30	1
Евро	35	1





Кросс-курс = 35/30.

Совершена сделка на 100 евро.

Эквивалент в валюте учета составит  $100 \cdot 35/30 = 116.67\$$

При оценке активов (обязательств) более чем в одной валюте появляется понятие курсовой разницы. Курсовая разница это разница в оценке активов (обязательств) в разных валютах, возникающая при изменении курса валюты. Как правило, переоценка валютных остатков на счетах производится от валюты сделки к валюте учета. Рассмотрим возникновение курсовой разницы при многовалютном учета в бухгалтерском финансовом учете РФ. Условия - валюта учета равна национальной (базовой). Валюта сделки - доллар США. Предположим, что на 01 января в кассе организации лежало 100 долларов США. Курс составлял 30 рублей за доллар. И эквивалент этих 100 долларов в валюте учета (рубли), таким образом, составлял 3000 рублей. 02 января курс доллара к рублю увеличился и составил 32 рубля за доллар. Таким образом, наша организация, удачно вложив денежные средства в иностранную валюту «выиграла». И «выиграла» она 200 рублей ( $100\$ \cdot 32 = 3200\text{р} - 3000\text{р} = 200\text{р}$ ). Это и есть положительная курсовая разница. Мы ничего не делали, но доход получили. Во-первых, нам необходимо отразить полученный доход, во-вторых, отразить увеличение активов (все это, понятно, в валюте учета).

Проводка: Дт. Касса / Кт. Доходы(Капитал...) 200 рублей

Если бы курс «упал» возникла бы отрицательная курсовая разница, и мы потерпели убыток и уменьшение активов, что отразили бы в учете обратной проводкой.

Если в кассе лежат средства в разных валютах - такие проводки нужно делать по каждой валюте. Если на счете ведется аналитический учет -по каждому объекту аналитического учета.

## Механизмы бухгалтерского учета в системе «1С:Предприятие 8.0»

«1С:Предприятие 8.0» содержит в себе следующие механизмы для организации бухгалтерского учета различных видов.

### Регистры бухгалтерии

Регистрация, группировка, контроль данных осуществляется в бумажной бухгалтерии в учетных регистрах. Их формы и состав зависит от применяемой на предприятии формы счетоводства (журнально-ордерная, мемориально-ордерная, журнал-главная и др.). При ведении учета в системе «1С:Предприятие 8.0» бухгалтерские проводки записываются в регистр бухгалтерии, который позволяет хранить данные хозяйственных операций с необходимой степенью детализации и группировать их по нужных признакам. Единственным обязательным (минимальным) разрезом хранимых в регистре данных являются счета и субсчета, что обеспечивает ведение синтетического учета. Дополнительно могут быть организовано ведение и других видов учета. Регистров бухгалтерии может быть несколько. Необходимость создания двух регистров может быть обусловлена, например, необходимостью вести учет по разным схемам (с поддержкой и без поддержки корреспонденции) или по разным планам счетов.

### Синтетический учет

В основе бухгалтерского учета всегда лежит план счетов. План счетов это совокупность синтетических счетов (счетов и субсчетов) предназначенных для группировки сводных данных. Планов счетов может быть несколько. Каждый из планов счетов может включать неограниченное число счетов и субсчетов. Счета могут заводиться как в режиме конфигурирования (предопределенные счета), так и в пользовательском режиме.

### Аналитический учет

Для ведения аналитического учета используются понятия Субконто (объект аналитического учета) и Вид субконто (аналитический разрез, совокупность однородных объектов аналитического учета). Для каждого счета (субсчета) может быть назначено несколько разных видов субконто для ведения параллельного (многомерного) учета в нескольких различных разрезах. Для организации хранения списка и наполнения видов субконто используется объект «Планы видов характеристик». План видов характеристик хранит в себе



характеристики, каждая из которых имеет тип значения. В плане видов характеристик, предназначенном для хранения видов субконто, характеристики, как правило, имеют ссылочный тип значения: СправочникСсылка, ПеречислениеСсылка и т.д. Таким образом, каждая характеристика хранит ссылку на некий массив информации, который и будет содержать субконто.

## Консолидированный учет

В отличие от аналитического учета, который предназначен для детализации итогов на отдельном счете, консолидированный учет подразумевает возможность консолидации итогов на всех счетах с возможностью получения сводной отчетности по предприятию в целом. В регистре бухгалтерии может быть заведено неограниченное количество подобных измерений.

## Другие виды учета

Кроме рассмотренных выше синтетического, аналитического и консолидированного учета, регистр бухгалтерии позволяет организовать ведение и других видов учета.

Учет многовалютный может быть организован в регистре с использованием небалансового измерения (валюта) и небалансового ресурса (валютная сумма)

Учет в третьей валюте потребует добавления в регистр нового балансового ресурса.

## Синтетический учет

Начнем решение нашей задачи с самого простого вида учета: синтетический учет в денежном выражении (в валюте учета). Основная задача - отражение хозяйственных операций на счетах бухгалтерского учета. Основным результатом - бухгалтерский баланс.

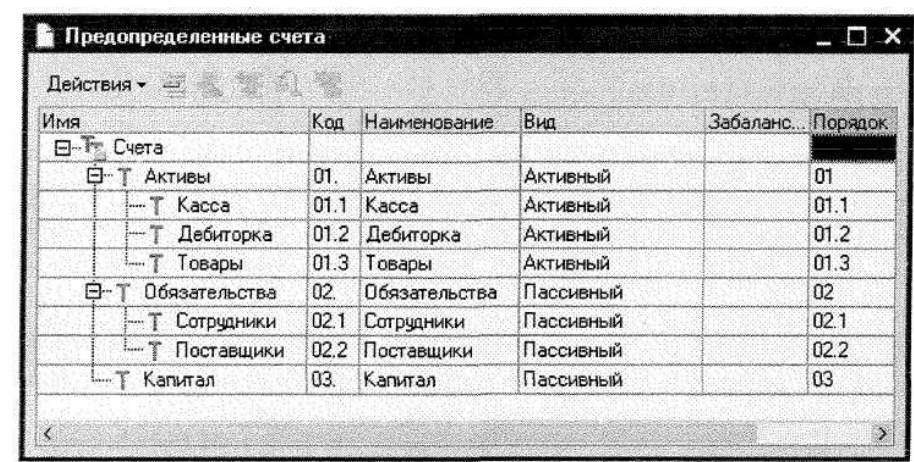
### План счетов

Синтетический учет начинается с плана счетов. Создаем новый план счетов. Имя **«Основной»**. На закладке «Данные» большую часть полей не изменяем. Заполняем маску кода (хотя он нам практически и не понадобится) @.@.@ - т.е. два знака отводится на номер счета, один знак — на номер субсчета, два уровня вложенности, код может содержать цифровые и строковые символы в верхнем регистре.

На самом деле, реальная иерархия счетов и иерархия кодов счетов-субсчетов (маска кода) могут отличаться. Так, например, счет, имеющий код 01.1, может не быть иерархически подчинен счету с кодом отличным от 01. Тем не менее, рекомендуется назначать счетам коды с учетом их иерархии.

Устанавливаем длину порядка 4 - нам хватит. Это поле позволит нам упорядочить счета в произвольном (удобном пользователю) порядке (не по коду, не по наименованию, а как-то еще). Основное представление в **виде наименования** (так нам будет проще работать с учебным планом счетов).

Далее переходим на закладку «Прочие» и открываем список предопределенных счетов. Как и справочник, план счетов может иметь счета, вводимые в пользовательском режиме, и те, которые задаются на этапе конфигурирования (предопределенные) и не изменяются пользователем. К таким счетам можно обращаться по имени из модулей. Заведем несколько предопределенных счетов.

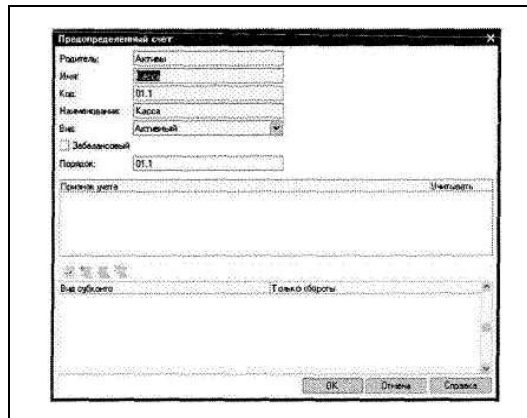


Имя	Код	Наименование	Вид	Забаланс...	Порядок
Счета					
Активы	01.	Активы	Активный		01
Касса	01.1	Касса	Активный		01.1
Дебиторка	01.2	Дебиторка	Активный		01.2
Товары	01.3	Товары	Активный		01.3
Обязательства	02.	Обязательства	Пассивный		02
Сотрудники	02.1	Сотрудники	Пассивный		02.1
Поставщики	02.2	Поставщики	Пассивный		02.2
Капитал	03.	Капитал	Пассивный		03



Для каждого счета мы указываем: Имя, Код, Наименование, Вид, Порядок.

Поле	Комментарий
Имя	Уникальное имя счета, которое позволит нам обращаться к нему «из языка»
Код	Код счета, который будет виден пользователю, если основным представлением для счета выбран «код»
Наименование	Наименование счета, которое будет видно пользователю, если основным представлением выбрано «наименование»
Вид	Определяет тип сальдо на счете. Счет может быть активным (дебетовое сальдо), пассивным (кредитовое сальдо) или активно-пассивным (тип сальдо зависит от оборотов и может быть как дебетовым, так и кредитовым)
Порядок	Поле используется для сортировки. В нашем случае может содержать код счета



## Регистр бухгалтерии

Следующая наша задача - создать регистр для хранения бухгалтерских проводок. Создаем новый регистр бухгалтерии. Задаем ему имя «Основной». Один регистр основан на одном плане счетов (при этом один план счетов может быть использован в нескольких регистрах). Если потребуется вести учет по нескольким планам счетов, то необходимо создать несколько регистров. На закладке «Основные» выбираем в качестве Плана счетов - «Основной». Устанавливаем флаг поддержки корреспонденции - он позволит делать проводки с указанием дебета и кредита, и, в результате, пользователь сможет видеть обороты между счетами. Регистр без поддержки

корреспонденции - позволит видеть лишь остатки и обороты отдельных счетов.

Регистр с поддержкой корреспонденции, который мы получим в результате, можно представить в виде таблицы записей:

Счет дебета	Счет кредита	Сумма	Содержание
Касса	Капитал	100	Инвестиция

В случае, если поддержку корреспонденции отключить, то регистр приобретет вид, как на картинке внизу, и та же операция будет отражена:

Счет	Вид движения	Сумма	Содержание
Касса	Дебет	100	Инвестиция
Капитал	Кредит	100	Инвестиция

Анализируя таблицы, можно заметить, что в первом случае мы сможем увидеть:

- Движения
- Остатки по счету
- Обороты по счету
- Обороты между счетами

Во втором, все то же самое, но без оборотов между счетами.

В курсе будет рассмотрена работа с регистром поддерживающем корреспонденцию

Итак, нам корреспонденции понадобятся.

Закладка «Данные» позволяет нам настроить измерения и виды учета. Как уже было сказано, синтетический учет — учет в денежном измерителе на счетах. Детализация по счетам в регистре бухгалтерии уже присутствует, осталось лишь создать ресурс, который будет хранить сумму в валюте учета. Добавляем новый ресурс

«Сумма», в котором устанавливаем свойство «Балансовый». Балансовый ресурс - присутствует в проводке в единственном экземпляре, т.е. одна и та же сумма будет «проходить» и по дебету и по кредиту проводки. Небалансовый ресурс создаст два поля - для дебета и для кредита.

Реквизиты регистра предназначены для хранения какой-либо дополнительной информации о проводке. В нашем случае, наверное, было бы неплохо разрешить пользователю оставить информацию о



смысле сделанной проводки. Создаем реквизит «Содержание» - строка 30 символов.

## Ввод данных в регистр бухгалтерии

Объект «Регистр сведений», который был рассмотрен в курсе «Введение в конфигурирование в системе «1С:Предприятие 8.0». Основные объекты», мог заполняться двумя способами: непосредственно пользователем в форме списка или из документа (свойство регистра - режим записи: подчинение регистратору).

Регистр бухгалтерии всегда подчинен регистратору, т.е. документу. Все движения, которые сделал один документ (регистратор), «держатся вместе» и по своей сути являются хозяйственной операций в бухгалтерском учете, хотя и могут содержать записи с различными периодами. Связывает записи регистратор - ссылка на документ, которому они принадлежат.

В системе «1С:Предприятие 8.0» существует несколько вариантов заполнения регистра бухгалтерии. Рассмотрим каждый из них.

### Ручной ввод проводок

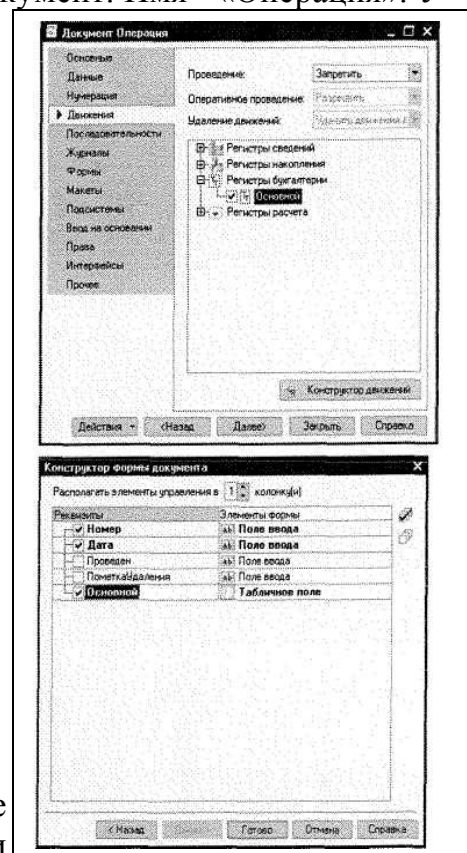
Для этого мы должны создать новый документ. Имя - «Операция». У него не будет дополнительных реквизитов. На закладке «Движения» устанавливаем: **проведение** - **запретить**. Проведение подразумевает, что должен быть написан алгоритм формирования проводок, который будет расположен в модуле объекта в процедуре обработчике события «Обработка проведения».

На этой же закладке в списке регистров выбираем - регистр бухгалтерии - «Основной».

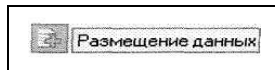
После чего переходим к проектированию форм. В форме документа размещаем Номер, Дату, и Табличное поле «Основной».

Помощник размещения данных разместит в диалоге табличное поле.

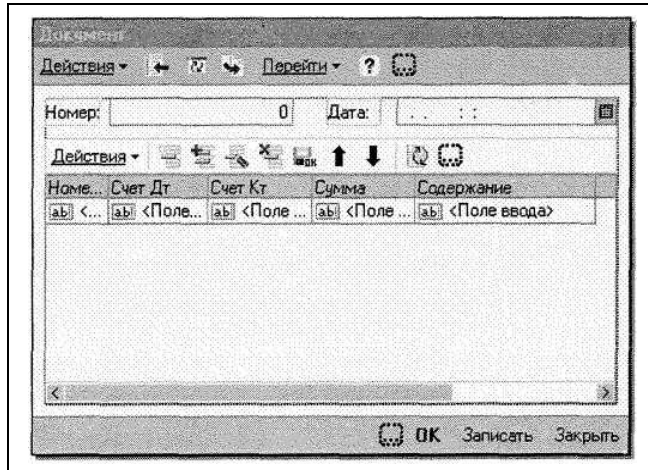
Колонки в табличном поле не появились. Щелчком правой кнопки мыши на табличном поле вызываем контекстное меню и выбираем пункт «Размещение данных» (или по



кнопке с панели инструментов колонки. Регистратор (собственно этот документ), активность (влияние этой записи регистра на итоги по регистру), период (в нашем случае будем считать, что все движения должны быть датой документа) - не вставляем. Конструктор размещения реквизитов разместит реквизиты, как на приведенном рисунке.



). Выбираем нужные



Теперь позаботимся о периоде добавляемых в регистр записей. Для этого воспользуемся обработчиком события объекта ДокументОбъект «Перед записью»

Процедура ПередЗаписью(Отказ, РежимЗаписи, РежимПроведения)

Для Каждого Проводка Из Движения.Основной Цикл

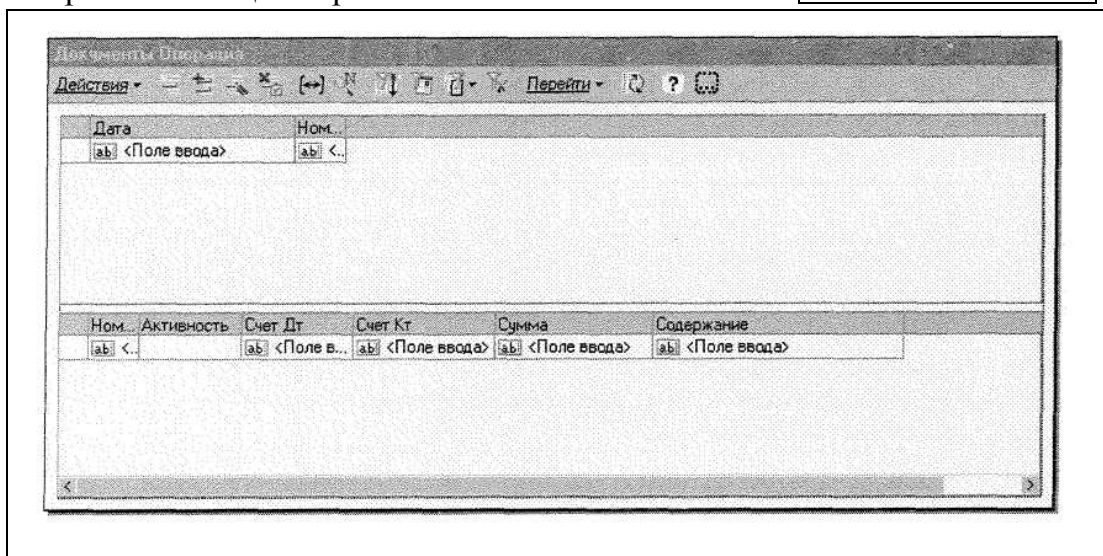
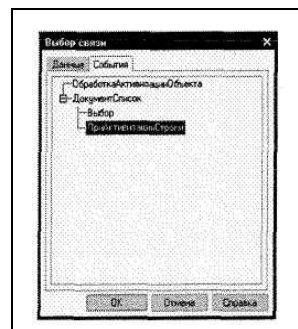
Проводка.Период = Дата;

КонецЦикла; КонецПроцедуры

Событие возникает перед выполнением записи документа (до начала записи, но уже в транзакции)

Обратите внимание, что форма документа также имеет событие ПередЗаписью. Отличия - обработчик события формы будет вызываться только при интерактивной записи документа из этой формы.

Осталось создать форму списка, где можно поместить табличное поле типа РегистрБухгалтерииСписок.Основной и связать его по регистратору (свойство табличного поля) - При активизации строки. Готово.

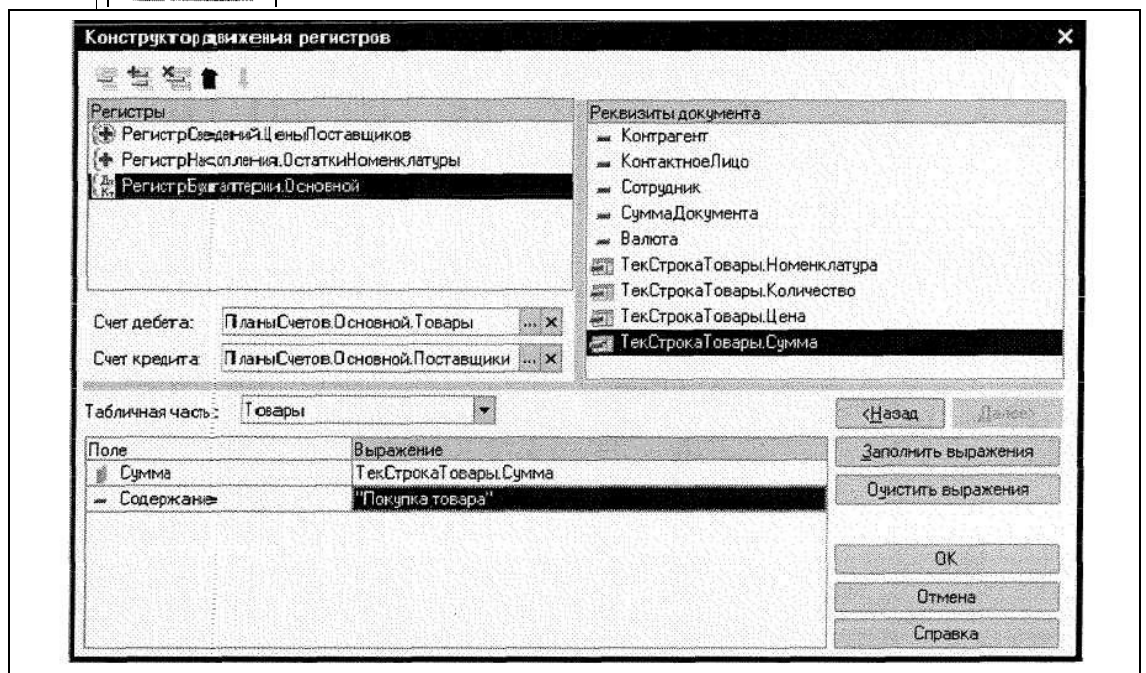
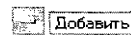


## Формирование движений (проводок) в момент проведения документа

Давайте научимся проводить документ «Приходная». Для этого на закладке «Движения» должны быть отмечены:

- 1.разрешим проведение;
- 2.оперативное проведение (мы пока не будем использовать);
- 3.удалять движения автоматически;
- 4.установим, что документ делает движения по регистру бухгалтерии «Основной».

Далее воспользуемся конструктором движений. По кнопке



32

с панели инструментов конструктора мы можем добавить новое

движение и заполнить его значениями реквизитов документа.

Счет дебета — Товары, счет кредита - Поставщики. Табличная часть -Товары, данные из колонки «Сумма» пойдут в ресурс «Сумма» регистра бухгалтерии.

В результате получим следующий модуль объекта документ будет дополнен следующими строками

```
Для Каждого ТекСтрокаТовары Из Товары Цикл //1
    // регистр Основной
    Движение = Движения.Основной.Добавить(); 1/2
    Движение.СчетДт = ПланыСчетов.Основной.Товары; //3
    Движение.СчетКт = ПланыСчетов.Основной.Поставщики;
    Движение.Период = Дата;
    Движение.Регистратор = Ссылка;
    Движение.Сумма = ТекСтрокаТовары.Сумма;
    Движение.Содержание = "Покупка
товара"; КонецЦикла;
// запись аем движения
регистров Движения .0
сновной.Записать(); //4
```

Комментарии к коду:

ОбработкаПроведения - это процедура-обработчик события объекта ДокументОбъект, именно в эту процедуру и были дописаны новые строки кода.

//1 в цикле обходим коллекцию строк табличной части документа.

//2 используя свойство «Движения» объекта типа «ДокументОбъект», получаем доступ к коллекции наборов записей по каждому регистру, привязанному к этому документу . И добавляем (метод «ДобавитьО» объекта типа «РегистрБухгалтерииНаборЗаписей») новое движение в набор записей по нужному регистру

//3 заполняем счета, ресурсы и реквизиты записи регистра. Счета «Товары» и «Поставщики» - predeterminedенные счета, к которым мы можем обращаться по их имени, при этом в движение будет записано значение типа ПланыСчетовСсылка.

//4 записываем набор записей в регистр

Посмотреть на результаты проведения можно из формы списка документа по кнопке «Перейти - Основной».

**Примечание:** существует возможность создавать, заполнять и проводить документы из обработки проведения другого документа. Метод *Записать(РежимЗаписиДокумента.Проведение)*

с

параметром режима записи (системное перечисление *РежимЗаписиДокумента*) для одного документа может быть вызван из процедуры «обработка проведения» другого документа, таким образом, мы имеем возможность в момент проведения документа

#### ■ создавать

Док = Документы.ПриходнаяНакладная.СоздатьДокумент ( )

Заполнять (те из Вас, кто имеет опыт программирования в системе «1С:Предприятие 7.7» узнают в событии объекта ДокументОбъект «Обработка заполнения» predeterminedенную процедуру ВводНаОсновании. В системе «1С:Предприятие 8.0» событие может быть вызвано методом «Заполнить»

Док.Заполнить(Ссылка)

И проводить документ

Док.Записать (РежимЗаписиДокумента.Проведение)

#### Запись движений документа без его проведения

Существует возможность записи в регистр движений по непроведенному документу. По своей сути, проведение является лишь «флагом», который сигнализирует, что обработка документа закончена. Можно представить ситуацию, когда обработка документа закончена не полностью, но проводки по нему уже должны быть .



Любая процедура или функция с помощью свойства «Движения» может записать движения в регистр по любому документу.

Например, можно расположить в модуле документа процедуру и отметить ее как экспортную (ключевое слово Экспорт после имени процедуры). И вызвать ее из процедуры обработчика какого-то события формы (например, обработчика нажатия на кнопку).

С точки зрения описания движений ничего не меняется. Единственное, о чем нужно помнить, что документ на момент нажатия на эту кнопку может быть еще не записан, и, как следствие, не может являться регистратором. Поэтому в начале процедуры можно или поставить проверку на то, что документ существует в базе данных (Если НЕ ЭтоНовый() Тогда) или программно записать документ

| Записать() :

---

#### Запись движений документа «извне»

Задача формирования движений по группе документов без их проведения может быть решена с помощью следующей процедуры. Процедура располагается в модуле формы списка документа и вызывается по кнопке с командной панели. В свойствах табличного поля списка документов при этом необходимо установить в разделе «Характеристики» режим выделения «Множественный», что позволит выделить несколько строк журнала (например, удерживая клавишу Shift или Ctrl).

```
Процедура СделатьПроводкиКДокументу (Кнопка)
    Строки = ЭлементыФормы.ДокументСписок.ВыделенныеСтроки; //1
    Для Каждого Строка Из Строки Цикл //2
        Документ = Строка.Ссылка; //3 Движения =
РегистрыБухгалтерии.Основной.СоздатьНаборЗаписей(); //4
        Если Не Документ.Проведен Тогда //5
            Движения.Отбор.Регистратор.Установить (Документ); //6
Для Каждого ТекСтрокаТовары Из Документ.Товары Цикл //7
    Движение = Движения.Добавить ();
    Движение.СчетДт = ПланыСчетов.Основной.Товары;
    Движение.СчетКт = ПланыСчетов.Основной.Поставщики;
    Движение.Период = Документ.Дата;
    Движение.Регистратор = Документ.Ссылка;
    Движение.Сумма = ТекСтрокаТовары.Сумма;
    Движение.Содержание = "Покупка";
КонецЦикла;
    КонечЕсли;
КонецЦикла;
    Движения.Записать (); //8
КонецПроцедуры
```

Комментарии к модулю:

//1 Свойство табличного поля Выделенные строки содержит в себе коллекцию строк, которые выделены пользователем

//2 Организуем обход строк коллекции //3

получаем ссылку на документ



```
//4 создаем набор записей, объект который
//5 проверяем документ на то, проведен ли он, если нет.
//6 устанавливаем в регистре отбор по регистратору
//7 внутри цикла по строкам документа добавляем движения в набор записей
//8 записываем движения в регистр.
```

### Активность записей

В дальнейшем для обращения к записям регистров будут использоваться виртуальные таблицы регистра. О том, что это такое, будет рассказано ниже. Пока же можно сказать, что эти таблицы предоставляют доступ к остаткам и оборотам по счетам, основываясь только на активных записях регистра. Активность - свойство записи регистра. Управлять им можно, например, из формы списка операции (если речь идет об активности записей документа операции). Процедура, которая приведена ниже, позволяет включить/выключить активность записей, должна быть расположена в модуле формы списка документа «Операция»:

Процедура ВключитьВыключитьПроводки (Кнопка)

```
Документ = ЭлементыФормы.ДокументСписок.ТекущаяСтрока; //1
Если Документ = Неопределено Тогда //2
    Возврат;
КонецЕсли;
```

```
Проводки = РегистрыБухгалтерии.Основной.СоздатьНаборЗаписей(); //3
Проводки.Отбор.Регистратор.Установить(Документ); //4
Проводки.Прочитать(); //5
```

```
Если Проводки.Количество() > 0 Тогда //6
    ТекущаяАктивность = Проводки[0].Активность; //7
    Проводки.УстановитьАктивность(НЕ ТекущаяАктивность); //8
    Проводки.Записать(); //9 КонецЕсли;
```

КонецПроцедуры

Комментарии к модулю:

```
// 1 получили ссылку на документ (объект ДокументСсылка)
//2 если список документа пустой (отсутствуют документы,
удовлетворяющие условиям, установленным для списка), делать
ничего не надо
//3 создали новый набор записей
//4 нас интересуют записи только указанного документа
//5 получили их из базы данных
//6 узнали количество проводок документе
//7 узнаем текущую активность записей по активности первой записи
набора
```





//8 инвертируем активность

//9 записываем набор записей в базу данных.

Та же функция, но вызываемая из модуля формы документа может быть реализована с помощью процедуры, приведенной ниже

```
Процедура ВключитьВыключить (Кнопка)
    Проводки = Движения.Основной;
    Проводки.УстановитьАктивность (НЕ
Проводки[0].Активность] КонецПроцедуры
```

**Регистр бухгалтерии как элемент табличной модели данных. Запрос по таблицам регистра бухгалтерии.**

Как уже было замечено, для хранения проводок в системе «1С:Предприятие 8.0» используются регистры бухгалтерии. Этот объект доступен нам через объектную и через табличную модели данных.

Как элемент объектной модели регистр бухгалтерии позволяет нам строить только выборки движений.

Как элемент табличной модели данных, бухгалтерский регистр представлен рядом виртуальных таблиц, которые позволяют не только строить выборки, но и быстро получать остатки и обороты с различными условиями и степенью детализации, о чем и пойдет речь в этом разделе.

Прежде чем приступить к построению отчетов проведем краткий обзор таблиц - источников данных запроса.

Для предварительного изучения таблиц воспользуемся обработкой «консоль запросов». Бухгалтерский регистр представлен следующими таблицами:

#### Таблица записей

Основная таблица, которая хранит записи регистра.  
Приведенный

ВЫБРА  
ТЬ \*

ИЗ

РегистрБухгалтерии.Основной

В качестве варианта применения этой таблицы можно предложить поиск всех документов, проводки которых «выключены», т.е. активность записей равна Ложь.

#### Значения субконто

Хранит значения субконто записи регистра

Выбра  
ТЬ \*

Из

## РегистрБухгалтерии.Основной.Субконто

Примечание; таблица появится после того, как будет организован аналитический  
 учет по субконто

## Движения с субконто

Виртуальная таблица позволяет получить информацию о записях вместе со значениями субконто. Эту таблицу можно считать основой для построения отчета о проводках, карточки счета.

ВЫБРАТЬ\*

ИЗ

РегистрБухгалтерии.Основной.ДвиженияССубконто

## Остатки

Виртуальная таблица позволяет получать остатки на произвольную дату (дату, момент времени или границу) в разрезе счета/счетов, измерений и субконто. Данные таблицы могут быть использованы при построении баланса, вывода в отчеты остатков на начало и на конец периода, расчета себестоимости, зачета аванса и т.д..

ВЫБРАТЬ

ИЗ

РегистрБухгалтерии.Основной.Остатки

## Обороты

Виртуальная таблица позволяет получать обороты за период с заданной периодичностью в разрезе счета, корсчета (корреспондирующего счета проводки), измерений, субконто и корсубконто (субконто корреспондирующего счета проводки).

ВЫБРАТЬ \*

ИЗ

РегистрБухгалтерии.Основной.Обороты

## Остатки и обороты

Виртуальная таблица позволяет получить и остатки, и обороты за произвольный период с заданной периодичностью, в разрезе счета, субконто, измерений. Может быть рассмотрена как основа для всех видов оборотно-сальдовых ведомостей.

ВЫБРАТ

Ь \*

ИЗ

РегистрБухгалтерии.Основной.ОстаткиИОбороты



### Обороты Дт Кт

Виртуальная таблица позволяет получать обороты за произвольный период с заданной периодичностью в разрезе счета дебета, счета кредита, измерений, субконто дебета, субконто кредита.

ВЫБРАТЬ

ИЗ

РегистрБухгалтерии.Основной.ОборотыДтКт

Подходит, например, для построения шахматки (шахматного баланса, который представляет собой кросс-таблицу, описывающую обороты между счетами), сводных проводок, отчета по проводкам.

### Особенности работы со счетами, имеющими субсчета

«1С:Предприятие» позволяет создавать планы счетов и справочники с неограниченной вложенностью элементов. Так, например, наш план счетов имеет 2 уровня вложенности - (счет и субсчет 1-го порядка). Причем, как Вы могли заметить, по умолчанию, программа позволяет делать проводки и по счетам, имеющим вложенные субсчета, т.е. по счетам-группам или синтетическим счетам.

Например, счет-группа «Активы», предназначенный для группировки данных об имуществе нашей компании и имеющий субсчета «Касса», «Дебиторка», «Товары», может выступать и как счет-группа (итоги по счету «Активы» будут включать итоги по всем его субсчетам, т.е. сумму активов нашего предприятия), и как самостоятельный счет, который также может участвовать в корреспонденции и иметь свои собственные остатки и обороты.

Как уже было замечено выше, для получения информации об остатках и оборотах по счетам, мы должны использовать запрос к регистру бухгалтерии.

В том случае, если нас интересуют остатки и обороты по счету «Активы», как по счету-группе, объединяющему субсчета активов нашего предприятия, нам следует воспользоваться предложением языка запросов:

Счет В ИЕРАРХИИ (&Счет)

Если же нас интересуют остатки и обороты по счету Активы, основанные только на тех проводках, где указанный счет выступает в качестве одного из корреспондирующих, то предложение запроса изменяется

Счет = &Счет

Рассм  
отрим

возмо  
жные  
вариа  
нты  
испол  
ьзова  
ние  
табли  
ц  
регис  
тра  
бухга  
лтери  
и для  
постр  
оения  
станд  
артно  
й  
бухга  
лтерс  
кой  
отчет  
ности  
.

### Оборотно-сальдовая ведомость

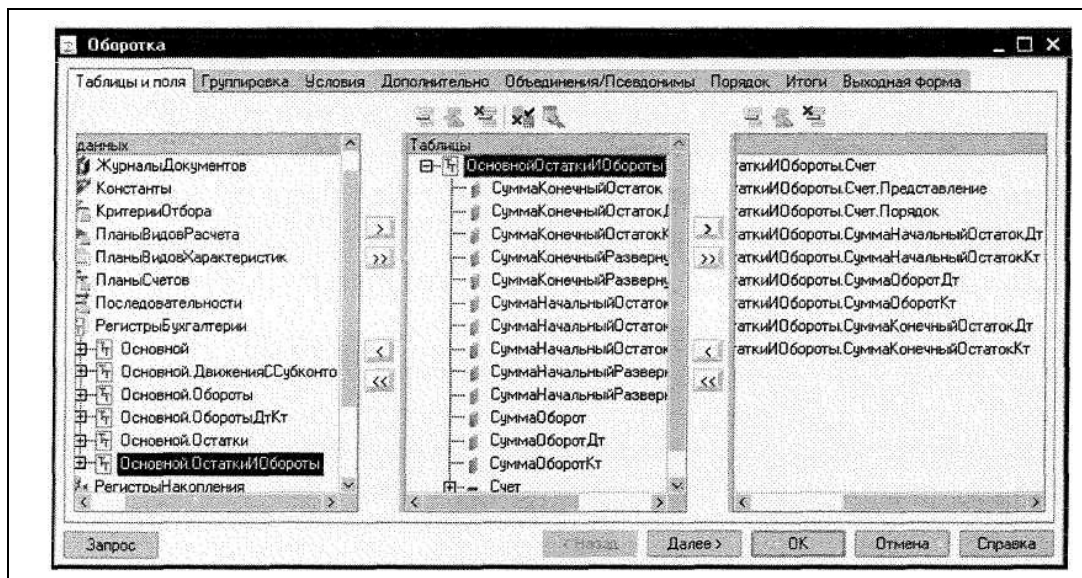
Наш первый отчет — оборотно-сальдовая ведомость. Для его построения нам потребуется обратиться к таблице регистра **ОстаткиИОбороты**. Создаем новый отчет «ОборотноСальдоваяВедомость». На закладке «Макеты» запускаем конструктор выходных форм. Идем по закладкам конструктора:


#### Таблицы и поля.

Среди таблиц находим РегистрыБухгалтерии - Управленческий.ОстаткиИОбороты и переносим список Таблиц, по которым будет выполняться запрос. Таблица содержит много разных полей. Среди них есть и нужные нам:

счет —

- Счет (с представлением), где сам ссылка (ПланыСчетовСсылка), представление - строка
- Счет.Порядок (свойство счета, которое можно «найти» открыв, щелкнув по «+» ветку «Счет») — потребуется нам, чтобы упорядочить результаты.
- Остальные имена полей «составные»: это имя ресурса регистра (мы назвали его Сумма) и имена остатков и оборотов этого ресурса, которые можно получить. Выбираем нужные:



Как уже было сказано выше, виртуальные таблицы в базе данных не хранятся, система автоматически собирает данные для них из реальных таблиц. И какие данные она «соберет» зависит от параметров, которые можно передать при обращении к этой виртуальной таблице. В конструкторе параметры в виртуальную таблицу можно передать, выделив в среднем списке (Таблицы) имя таблицы и нажав кнопку  **Параметры виртуальной таблицы**.







(группировку на себя возьмет виртуальная таблица, за которой «прячется» запрос, выполняемый системой «1С:Предприятие 8.0», чтобы подготовить для нас требуемую таблицу). На закладке «Условия» нужно написать одно условие (произвольное):

ОсновнойОстаткиИОбороты.Счет.Забалансовый = ЛОЖЬ

Это условие позволит не включать в результат запроса забалансовые счета (их пока нет, но вдруг появятся). На закладке «Порядок» указываем, что отсортировать надо по представлению счета, которое, в нашем случае, соответствует коду счета.

На закладке «Итоги» указываем, что нас интересуют итоги по счету (элементы и иерархия, что позволит включить в отчет итоги по счету и по счетам-группам), указываем суммируемые поля, флажок «общие итоги».

На закладке «Выходная форма» для параметров ДатаНачала и ДатаОкончания необходимо указать тип «Дата».

Получили почти готовый отчет. Почти - потому что остатки на начало и на конец ОБЩИЕ запрос не рассчитывает. Их необходимо накапливать внутри цикла по счетам, и Итоги По Общие из текста запроса можно убрать.

#### Отчет по проводкам - Таблица «Движения с субконто»

«Движения с субконто» - виртуальная таблица, которая позволяет получить информацию о записях регистра вместе со значениями субконто. Субконто, правда, в нашей конфигурации, пока нет. Наша конфигурация предназначена пока только для ведения учета синтетического, а субконто — элемент аналитического учета. В нашем случае вполне можно было бы использовать таблицу записей регистра бухгалтерии. Выбор пал на таблицу «Движения с субконто» исключительно в методических целях - чтобы с ней познакомиться. Эта таблица может быть положена в основу отчета о проводках, карточки счета и др. Разработать отчет о проводках можно с помощью конструктора выходных форм. Если задача - получить отчет, который

Мы  
в  
каче  
стве  
пар  
аме  
тров  
пер  
едад  
им  
лиш  
ь  
инт  
ерва  
л  
дат.  
Зак  
лад  
ки:  
Гру  
ппи  
ров  
ка,  
Доп  
олн  
ите  
льн  
о,  
Объ  
еди  
нен  
ия/  
Псе  
вдо  
ним  
ы  
мы  
про  
пус  
кае  
м

будет отбирать проводки по одному счету, неважно в дебете проводки или в кредите, то текст запроса будет следующий:

```
Процедура Отчет(ТабДок, Дата1, Дата2, Счет) Экспорт
    Макет = ПолучитьМакет("Отчет"); Запрос =
    Новый Запрос; Запрос.Текст = "ВЫБРАТЬ
    ОсновнойДвиженияССубконто.Период КАК Период,
    ОсновнойДвиженияССубконто.Регистратор,
    ОсновнойДвиженияССубконто.Представление,
    ОсновнойДвиженияССубконто.Сумма КАК Сумма |ИЗ
    | РегистрБухгалтерии.Основной.ДвиженияССубконто(&Дата1,
    &Дата2, СчетДт В ИЕРАРХИИ (&Счет) ИЛИ СчетКт В ИЕРАРХИИ (&Счет))
    КАК ОсновнойДвиженияССубконто";

    Запрос.УстановитьПараметр("Дата1", Дата1);
    Запрос.УстановитьПараметр("Дата2", Дата2);
    Запрос.УстановитьПараметр("Счет", Счет);
    Результат = Запрос.Выполнить();

    ОбластьЗаголовок = Макет.ПолучитьОбласть("Заголовок");
    ОбластьПодвал = Макет.ПолучитьОбласть("Подвал");
    ОбластьШапкаТаблицы = Макет.ПолучитьОбласть("ШапкаТаблицы");
    ОбластьПодвалТаблицы = Макет.ПолучитьОбласть("ПодвалТаблицы");
    ОбластьДетали = Макет.ПолучитьОбласть("Детали");

    ТабДок.Вывести(ОбластьЗаголовок);
    ТабДок.Вывести(ОбластьШапкаТаблицы);

    ВыборкаДетали = Результат.Выбрать();

    Пока ВыборкаДетали.Следующий{} Цикл
        ОбластьДетали.Параметры.Заполнить(ВыборкаДетали);
        ТабДок.Вывести(ОбластьДетали);
    КонечЦикла;

    ТабДок.Вывести(ОбластьПодвалТаблицы);
    ТабДок.Вывести(ОбластьПодвал); КонечПроцедуры
```

При желании, можно подвести итоги, хотя большого смысла они иметь не будут (будут складываться и дебетовые и кредитовые обороты счета). После добавления в нашу конфигурацию аналитического учета по субконто, можно будет включить в отчет информацию об аналитике проводки и/или установить отбор по аналитике. Отобрать по аналитике можно будет в том же параметре виртуальной таблицы, где сейчас мы описали условие по счету (третий параметр).

Если же хочется видеть в отдельных колонках суммы, которые прошли по дебету выбранного счета и те, которые были по кредиту, то отчет можно донстроить отчет:

1. Изменить текст запроса, добавив два поля - СчетДт и СчетКт.
2. Изменить макет отчета - вместо колонки сумма сделать две: СуммаДт, СуммаКт



3.       Добавить еще одну именованную область (строку)  
          «Обороты», содержащую два параметра - ОборотДт,  
          ОборотКт
4.       \_\_\_\_\_       Изменить процедуру обработки  
результата запроса.

ВыборкаДетали = Результат.Выбрать ( ) ;

ОборотДт\* = 0 ;

ОборотКт = 0 ;

Пока ВыборкаДетали.Следующий ( ) Цикл

    ОбластьДетали.Параметры.Заполнить (ВыборкаДетали) ;

    Если ВыборкаДетали.СчетДт = Счет Тогда

        ОбластьДетали.Параметры.СуммаДт = ВыборкаДетали.Сумма ;

        ОбластьДетали.Параметры.СуммаКт = 0 ;

        ОборотДт = ОборотДт + ВыборкаДетали.Сумма ;

    КонецЕсли ;

```

Если .Вывести (ОбластьДетали) ;
Выборк КонецЦикла;
аДетал
и.Счет ОбластьОбороты =
Кт Макет.ПолучитьОбласть ("Обороты"); Область
Счет Обороты.Параметры.ОборотДт = ОборотДт;
Тогда ОбластьОбороты.Параметры.ОборотКт = ОборотКт;
Област ТабДок.Вывести (ОбластьОбороты) ;
ьДетал
и.Пара
метры. ТабДок.Вывести (ОбластьПодвалТаблицы)
СуммаД

```

Облас  
тьДет  
али.П  
араме  
тры.С  
уммаК  
т =  
Выбор  
каДет  
али.С  
умма,  
Оборо  
тКт =  
Оборо  
тКт +  
Выбор  
каДет  
али.С  
умма;  
КонецЕс  
ли;  
Т  
а  
б  
л  
о  
к

**Обратите внимание:** в условном операторе мы проверяем равенство поля таблицы и ссылки из диалога... Т.е. мы оставляем необработанной ситуацию, когда отчет строится по счету-группе. Можно попытаться искать (метод Найти()) код счета выборки в коде счета, выбранного в диалоге, но тогда мы «завязываемся» на коды счетов, которые, откровенно говоря, могут или вообще отсутствовать или не иметь ничего общего с иерархией. Как поступить?

Например, можно воспользоваться оператором запроса ВЫБОР:

```

"ВЫБРАТЬ
| ОсновнойДвиженияССубконто.Период КАК Период,
| ОсновнойДвиженияССубконто.Регистратор,
|
ОсновнойДвиженияССубконто.Регистратор.Представление,
ОсновнойДвиженияССубконто.СчетДт КАК
СчетДт, | ОсновнойДвиженияССубконто.СчетКт
КАК СчетКт, | ВЫБОР КОГДА
ОсновнойДвиженияССубконто.СчетДт
В ИЕРАРХИИ (&Счет) ТОГДА ОсновнойДвиженияССубконто.Сумма ИНАЧЕ 0
КОНЕЦ КАК СуммаДт,
Т
а
б
л
о
к
"ВЫБОР КОГДА ОсновнойДвиженияССубконто.СчетКт В
ИЕРАРХИИ (&Счет) ТОГДА ОсновнойДвиженияССубконто.Сумма ИНАЧЕ 0
КОНЕЦ КАК
СуммаКт
|ИЗ

```

```
| РегистрБухгалтерии.Основной.ДвиженияССубконто (&Дата1,
&Дата2, СчетДт В ИЕРАРХИИ (&Счет) ИЛИ СчетКт В ИЕРАРХИИ (&Счет))
КАК ОсновнойДвиженияССубконто
```

```
| ИТОГИ СУММА (СуммаДт), СУММА (СуммаКт) ПО
ОБЩИЕ";
```

В этом случае, в результате выполнения запроса мы получим таблицу, уже содержащую нужные нам поля. Останется лишь перебрать результаты.

Как Вы, наверное, заметили, мы потихоньку приближаемся к Карточке счета - стандартному отчету, который присутствует в любой версии «1С:Бухгалтерии». Не хватает:

1. Остатка на начало периода (дебетового или кредитового)
2. Текущего остатка по каждой строке
3. Остатка на конец периода (который можно рассчитать как остаток на начало +/- оборот)

Как получить остаток?

### Остатки по счетам - таблица «Остатки»

«Остатки» - виртуальная таблица, которая позволяет получить остатки в разрезе счетов, измерений и субконто (если есть аналитический учет по субконто). С помощью этой таблицы можно построить, например, баланс на указанную дату.

Давайте построим отчет, который покажет нам остатки на начало периода и текущий остаток по счету.

```
"ВЫБРАТЬ
| ОсновнойОстатки.Счет,
ОсновнойОстатки.Счет.Представление,
ОсновнойОстатки.СуммаОстаток,
ОсновнойОстатки.СуммаОстатокДт,
ОсновнойОстатки.СуммаОстатокКт | ИЗ
| РегистрБухгалтерии.Основной.Остатки (&ВыбДата, Счет В
ИЕРАРХИИ (&Счет) , , ) КАК ОсновнойОстатки";
```

В данном случае нас интересуют поля: ОстатокДт, ОстатокКт и поле Остаток, которое считается как ОстатокДт - ОстатокКт.

### Отчет «Сводные проводки» - Таблица «Обороты Дт Кт»

Таблица «Обороты Дт Кт» позволяет получать обороты между счетами между заданным счетом дебета и кредита. Ее можно использовать, например, для получения отчетов типа «Шахматка» и «Сводные проводки».

```
"ВЫБРАТЬ
ОсновнойОборотыДтКт.СчетДт,
ОсновнойОборотыДтКт.СчетДт.Представление,
ОсновнойОборотыДтКт.СчетКт,
ОсновнойОборотыДтКт.СчетКт.Представление,
ОсновнойОборотыДтКт.СуммаОборот ИЗ
```





РегистрБухгалтерии.Основной.ОборотыДтКт{&Дата1,  
&Дата2) КАК ОсновнойОборотыДтКт";

Запрос, который позволяет получить стандартный для  
«1С:Бухгалтерии» отчет — «Сводные проводки»

### Таблица «Обороты Дт Кт» - детальный отчет

Значительная часть виртуальных таблиц, среди которых и ОборотыДтКт, имеет среди параметров «Периодичность». Этот параметр позволяет получить дополнительную группировку запроса по периодам. Периоды, которые могут выступать в качестве группировки данных виртуальной таблицы можно увидеть, например, в диалоге заполнения параметров таблицы конструктора выходных форм.

Параметр может принимать значения: Период, Запись, Регистратор, День, Неделя, Декада, Месяц, Квартал, Полугодие, Год.

Периодичность «Период», или незаполненный параметр таблицы, говорит о том, что дополнительной группировки не будет и периодичностью будет период отчета.

Периодичность «Запись», позволяет получить отчет с детализацией до проводки. Таким образом, отчет, который был нами построен ранее (отчет по проводкам) на основании таблицы ДвиженияССубконто, при желании получить и на основе других таблиц.

Давайте построим отчет по проводкам (т.е. с детализацией до записи регистра) на основе таблицы ОборотыДтКт.

Для этого заполним параметры виртуальной таблицы, как показано на рисунке.

Из таблицы будет отображены поля:

Период
СчетДт
СчетКт
СуммаО

На  
закла  
дке  
«Итог  
и»  
укаже  
м, что  
нужн  
о  
подво  
дить  
общ  
е  
итоги  
по  
полю  
Сумм  
аОбор  
от.

На закладке «Выходная форма» назначим соответствующие типы для параметров. Отчет готов.

#### Отчет по проводкам (таблица ОборотыДтКт)

Период	Регистратор	СуммаОборот
28.10.2003 11:26:42	Приходная накладная 1 от 28.10.2003 11:26	100
28.10.2003 11:26:42	Приходная накладная 1 от 28.10.2003 11:26	15
28.10.2003 11:26:42	Приходная накладная 1 от 28.10.2003 11:26	15
28.10.2003 12:18:41	Расходная накладная A0001 от 28.10.2003	50
28.10.2003 12:18:41	Расходная накладная A0001 от 28.10.2003	10
28.10.2003 13:10:00	Приходная накладная 2 от 28.10.2003 13:10	40
28.10.2003 13:10:00	Приходная накладная 2 от 28.10.2003 13:10	25
28.10.2003 13:10:00	Приходная накладная 2 от 28.10.2003 13:10	15

## Практикум № 2

Используя полученные в последних двух темах (таблицы ДвиженияСсубконто и Остатки) знания (и наработки), разработайте отчет «Карточка счета». Пользователь может выбрать в диалоге: интервал дат, счет. В таблице результата пользователь должен увидеть: остатки на начало (дебетовый или кредитовый), по каждой строке: сумму проводки (дебет или кредит счета) и текущее сальдо по строке (Остаток на начало дебетовый - остаток на начало кредитовый + сумма по дебету - сумма по кредиту), обороты за период и остаток на конец периода. Остаток на конец можно получить арифметически или еще раз выполнив запрос по остаткам.

#### Карточка счета

Период	Регистратор	Дебет	Кредит	Остаток
<b>Остаток</b>				
29.10.2003 14:02:17	Операция 1 от 29.10.2003 14:02:17	2 900		2 900
30.10.2003 11:14:40	Операция 2 от 30.10.2003 11:14:40	1 000		3 900
30.10.2003 11:14:40	Операция 2 от 30.10.2003 11:14:40	71 000		74 900
30.10.2003 11:14:40	Операция 2 от 30.10.2003 11:14:40	9 657		84 557
30.10.2003 11:15:23	Операция 3 от 30.10.2003 11:15:23	3 550		88 107
30.10.2003 11:15:23	Операция 3 от 30.10.2003 11:15:23		1 420	86 687
<b>Обороты</b>		<b>88 107</b>	<b>1 420</b>	
<b>Остаток</b>		<b>86 687</b>		



## Аналитический учет

Под аналитическим учетом будем понимать учет на счетах бухгалтерского учета в денежном выражении в разрезе аналитических признаков, которые соответствуют в системе «1С:Предприятие» понятию «Субконто». Группа однородных объектов аналитического учета (субконто) называются «Вид субконто». Для ведения аналитического учета нам необходимо произвести настройку объектов Планы видов характеристик, Планы счетов, Регистры бухгалтерии.

### План видов характеристик и виды субконто

План видов характеристик имеет несколько предназначений. При автоматизации бухгалтерского учета основное из них - хранение списка видов субконто (аналитических счетов). В общем случае, этот объект предназначен для описания множеств однотипных объектов аналитического учета, т.е. может содержать несколько таких множеств. Часть этих «множеств» (их название, наполнение) могут быть определены на этапе конфигурирования (предопределенные характеристики). Часть - в пользовательском режиме.

Каждое из этих множеств - вид субконто, т.е. аналитический счет. А его наполнение (элементы справочника, значения перечисления, документы и т.д.) - элементы аналитического учета, субконто.

Во-первых, нам необходимо предусмотреть все виды субконто, которые мы сделаем «жесткими» и которые пользователь изменять не будет. Для этого надо решить, какие аналитические разрезы нам могут понадобиться при проведении документов. Реализуются «жесткие» (т.е. не изменяемые пользователем) виды субконто с помощью предопределенных значений характеристик и служат для организации аналитического учета на тех счетах, которые будут задействованы в алгоритмах проведения документов. Как и любой объект конфигурации, предопределенная характеристика имеет «имя», которое позволяет обращаться к ней «из языка».

Скорее всего, расчеты с покупателями и поставщиками, мы захотим детализировать по контрагентам (как минимум, а может и по договорам, при необходимости учета взаиморасчетов в разрезе договоров). Товары - по товарам. Взаиморасчеты с сотрудниками в разрезе сотрудников. Надо отметить, что подобная упрощенная организация аналитического учета возможна только в учебных целях. Организация аналитического учета в реальной бухгалтерии будет определяться тем набором показателей в отчетах, которые необходимо получать конечному пользователю - бухгалтеру. Другими словами, желание бухгалтера видеть отчет о взаиморасчетах с контрагентами в

разре  
зе  
контр  
агент  
ов  
прив  
едет  
к  
необ  
ходи  
мост  
и  
веден  
ия

аналитического учета на счете расчетов в разрезе вида субконто «Контрагенты». В случае необходимости получения отчета о товарных запасах в разрезе номенклатуры товаров и складов, потребуется «привязать» к счету учета товаров два вида субконто - «Номенклатура» и «Склады».

Создаем новый план видов характеристик **«ВидыСубконто»**. Тип значения характеристик указываем «составной» и отмечаем типы: СправочникСсылка.Контрагенты, Номенклатура, ФизическиеЛица.

Свойство «Дополнительные значения характеристик» позволяет пользователю вводить новые характеристики, которые будут наполняться значениями из выбранного справочника. Этот справочник должен быть подчинен плану видов характеристик.

Для этих целей создадим новый справочник, иерархический, без дополнительных реквизитов, имя «Субконто» и сделаем его подчиненным плану видов характеристик «ВидыСубконто». После чего выберем его в качестве дополнительных значений характеристик. Эта связка - элемент плана вида характеристик и подчиненные ему элементы справочника позволят создать неограниченное количество «множеств» объектов аналитического учета.

Внимание: справочник должен быть обязательно включен в составной тип данных «Тип значения характеристик».

Закладку «Иерархия» в данном случае оставляем нетронутой (нам нужен линейный список видов субконто).

Переходим на закладку «Прочие» и создаем нужные предопределенные виды субконто.



Когда список видов субконто готов, можно вернуться в план счетов «Основной» и продолжить его настройку.

## Настройка плана счетов для ведения аналитического учета

На закладке «Субконто» плана счетов «Основной» выбираем в свойстве «виды субконто» созданный план видов характеристик «ВидыСубконто».





Необходимо выполнить сохранение конфигурации и обновление конфигурации базы данных до того, как будут выполнены последующие действия.

и устанавливаем максимальное количество субконто на счете (для решения нашей учебной задачи нам вполне хватит одного вида субконто на счете, но в учебных же целях устанавливаем «два» - на вырост). Признаки учета по субконто пока не создаем, они позволят организовать только количественный, или только валютный (как пример) для отдельных видов субконто отдельных счетов. Механизм признаков учета по субконто будет рассмотрен нами отдельно, пока же можно заметить, что существует один признак учета по субконто «Только обороты», который позволяет отказаться от хранения в регистре бухгалтерии остатков по указанному счету в разрезе объектов аналитического учета

Следующая задача - подключить аналитику к соответствующим счетам плана счетов. Выполняется это в списке predeterminedенных счетов (закладка «Прочие»)

Первый уровень иерархии счетов	Второй уровень иерархии счетов	Первый субконто	вид	Только обороты
<b>Активы</b>				
	Касса			
	Дебиторка	Контрагенты		Нет
	Товары	Номенклатура		Нет
<b>Обязательства</b>				
	Сотрудники	Сотрудники		Нет
	Поставщики	Контрагенты		Нет
<b>Капитал</b>				Нет

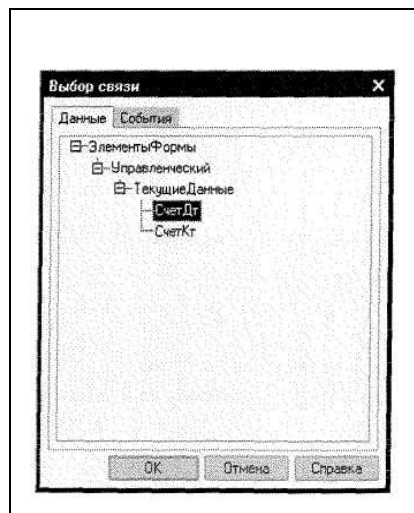
## Ввод данных по аналитике в регистр

В самом регистре менять ничего не придется, он сам будет хранить данные по счетам, к которым прикреплен аналитика в разрезе объектов аналитики. Осталось изменить документы, которые используются для ввода проводок.

### Документ Операция

В документе «Операция», используя помощник размещения данных, необходимо добавить в табличное поле регистра, размещенное в диалоге формы документа, четыре колонки: два субконто дебета и два субконто кредита. После чего навести «красоту».

Колонки можно менять местами: хватаем колонку за заголовок и «тащим» мышкой в нужное место. Кроме того, для колонок можно задавать положение (раздел





«Характеристики» свойств колонки).

Чтобы пользователю не задавалось лишних вопросов при выборе субконто, нужно для всех субконто (поля ввода СубконтоДт1, СубконтоДт2, СубконтоКт1, СубконтоКт2) в свойствах Поля ввода в разделе Использование «Связь по типу» установить связь. Для субконто дебета - СчетДт, для субконто кредита - Счет Кт.

Там же (в окне свойств) в поле «Выбор групп и элементов» имеет смысл поставить «Выбор элементов», чтобы запретить пользователю делать проводки по группам справочников (если, конечно, мы хотим запретить это пользователю, программа позволяет анализировать как итоги по счету, так и итоги по всем его субсчетам).

## Документ «Приходная накладная»

Изменить модуль объекта «Приходная накладная» для корректного формирования движений по регистру бухгалтерии по субконто позволит уже знакомый нам конструктор.

В результате процедура «Обработка проведения» изменится. В цикл формирования движений по регистру бухгалтерии добавится пара новых строк.

```
Движение.СубконтоДт [ПланыВидовХарактеристик.ВидыСубконто.
Номенклатура] = ТекСтрокаТовары.Номенклатура;
Движение.СубконтоКт [ПланыВидовХарактеристик.ВидыСубконто.
Контрагенты] = Контрагент;
```

## Отчеты по аналитике

В качестве примера отчета с группировкой по аналитике может служить отчет «движение товаров», предназначенный для анализа остатков и оборотов по счету товаров в разрезе видов номенклатуры.



## Движение товаров

Для построения отчета воспользуемся уже знакомым нам конструктором выходных форм. Идем по закладкам.

### Таблицы ж поля

Выбираем таблицу ОстаткиИОбороты, для которой задаем параметры. Ниже в таблице представлены задаваемые параметры (колонка «Движение товаров») в сравнении с параметрами ранее сделанного нами отчета «Оборотка»

Параметр	Оборотка	Движение товаров
НачалоПериода	&ДатаНачала	&ДатаНачала
КонецПериода	&ДатаОкончания	&ДатаОкончания
Периодичность		
МетодДополнения		
УсловиеСчега		Счет = &Счет
Субконго		
Условие		Субконто1 В ИЕРАРХИИ (&Субконто)

Как видим, по сравнению с отчетом «Оборотно-сальдовая ведомость» добавились два параметра: мы указываем условие по счету и условие на субконто этого счета, по которому отбирать. Где Субконто 1 в слева от знака равенства - поле таблицы, а &Субконто - параметр запроса, значение которого выбирается пользователем в поле ввода типа СправочникСсылка.Номенклатура в диалоге формы..

В списке поля указываем:

- Счет (с представлением) - чтобы увидеть итог по счету в целом
- Субконто1 (с представлением)
- СуммаНачальныйОстатокДт
- СуммаОборотДт
- СуммаОборотКт
- СуммаКонечныйОстатокДт

До закладки Итоги можем все пропустить

## Итоги

На закладке «Итоги» указываем группировочные поля Счет (что позволит получить итогов по счету в целом) и Субконто 1 с типом итогов «Элементы и иерархия» (в результате запроса будут добавлены итогов по элементам и группам справочника «Номенклатура»). Суммируемые поля - остатки и обороты.

Выходная форма

Указываем, что параметры «ДатаНачала» и «ДатаОкончания» должны быть вставлены в диалог и иметь тип «Дата», Субконто - тип СправочникСсылка.Номенклатура А параметр «Счет» в диалог вставлять не нужно - определим в модуле.

Изменения в модуле объекта

После того, как конструктор построит отчет, нам останется лишь изменить передачу параметров в запрос

```
Запрос.УстановитьПараметр("ДатаНачала", ДатаНачала);
Запрос.УстановитьПараметр("ДатаОкончания", ДатаОкончания);
Запрос.УстановитьПараметр("Счет", ПланыСчетов.Основной.Товары);
```

Из макета отчета можно удалить колонку 2 (где показывается представление счета), а из модуля - детальную выборку. Отчет готов.

Движение товаров				
Субконто1	СуммаНачальныйОстатокДт	СуммаОборотДт	СуммаОборотКт	СуммаКонечныйОстатокДт
	657	1352	519,88	1 489,12
Авторучки	630	1 300	500	1 430
Паркер	630	1 300	500	1 430
Карандаши	9	27	9	27
Цветной	9	27	9	27
Чернила	18	55	10,88	62,12
Радуга	18	55	10,88	62,12

Практикум №3

Построить отчет о задолженности поставщикам. Отчет аналогичен отчету ДвижениеТоваров, но выполняется по счету Поставщики. У пользователя должна быть возможность выбора в



диалоге интервала дат и поставщика. Предусмотрите в запросе итог по счету (верхний итог) и по поставщикам (с иерархией).

## Выручка в кассу - таблица «Обороты»

Таблица «Обороты» - виртуальная таблица, которая позволяет получить обороты между счетом и корсчетом в разрезе измерений и периодов. Пример использования этой таблицы - отчет, который должен показать, сколько какой покупатель заплатил нам за указанный период. В данном случае нас интересует оборот в дебет счета Касса с кредита счета Дебиторка в разрезе субконто счета Дебиторка.

Отчет должен содержать две колонки: покупатель и сумма выручки.

```
Процедура ВыручкаВКассу(ТабДок, Дата!, Дата2, КорСчет,
Субконто, Счет) Экспорт
    Макет = ПолучитьМакет{"Отчет"};
    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
        |      ОсновнойОбороты.Субконто1 КАК Субконто1,
        |      ОсновнойОбороты.Субконто1.Представление,
        |      ОсновнойОбороты.СуммаОборотКт КАК СуммаОборотКт
        |ИЗ
        |      РегистрБухгалтерии.Основной.Обороты(&Дата1, &Дата2,
        |      , Счет В ИЕРАРХИИ (&Счет), , Субконто! В ИЕРАРХИИ (&Субконто),
        |      КорСчет В ИЕРАРХИИ (&КорСчет), ) КАК ОсновнойОбороты

        ИТОГИ СУММА (СуммаОборотКт) ПО
            ОБЩИЕ, |
        Субконто1 ИЕРАРХИЯ";

    Запрос.УстановитьПараметр("Дата!", Дата!);
    Запрос.УстановитьПараметр("Дата2", Дата2);
    Запрос.УстановитьПараметр("КорСчет",
        ПланыСчетов.Основной.Касса);
    Запрос.УстановитьПараметр("Субконто", Субконто);
    Запрос.УстановитьПараметр("Счет",
        ПланыСчетов.Основной.Дебиторка);

    Результат = Запрос.Выполнить();

    ОбластьЗаголовков = Макет.ПолучитьОбласть("Заголовков");
    ОбластьПодвал = Макет.ПолучитьОбласть{"Подвал"};
    ОбластьШапкаТаблицы =
        Макет.ПолучитьОбласть("ШапкаТаблицы");
    ОбластьПодвалТаблицы =
        Макет.ПолучитьОбласть("ПодвалТаблицы");
    ;
    ОбластьОбщийИтог = Макет.ПолучитьОбласть("ОбщиеИтоги");
    ОбластьСубконто1Иерархия =
        Макет.ПолучитьОбласть("Субконто1Иерархия");
    ;
    ОбластьСубконто! = Макет.ПолучитьОбласть("Субконто!");
    ОбластьДетали = Макет.ПолучитьОбласть("Детали");
```



Т  
а  
б  
л  
и  
ц  
а  
.  
В  
ы  
в  
е  
с  
т  
и  
(  
О  
б  
л  
а  
с  
т  
ь  
З  
а  
г  
о  
л  
о  
в  
о  
к  
)  
;

ТабДок.Вывести (ОбластьШапкаТаблицы)  
;

```

        ВыборкаОбщийИтог =
        Результат.Выбрать (ОбходРезультатаЗапроса.ПоГруппировкам) ;

        ВыборкаОбщийИтог.Следующий() ;           // Общий итог
        ОбластьОбщийИтог.Параметры.Заполнить (ВыборкаОбщийИтог) ;
        ТабДок.Вывести (ОбластьОбщийИтог) ;

        ВыборкаСубконто! =
        ВыборкаОбщийИтог.Выбрать (ОбходРезультатаЗапроса.ПоГруппировкам) ;

        Пока ВыборкаСубконто1.Следующий() Цикл
            Если ВыборкаСубконто1.ТипЗаписи{} =
            ТипЗаписиЗапроса.ИтогПоИерархии Тогда
                Область = ОбластьСубконто1Иерархия;
            Иначе
                Область = ОбластьСубконто!;
            КонецЕсли;

            Область.Параметры.Заполнить (ВыборкаСубконто1) ;
            ТабДок.Вывести (Область) ;

            ВыборкаДетали = ВыборкаСубконто1.Выбрать() ;

        КонецЦикла;

        ТабДок.Вывести (ОбластьПодвалТаблицы) ;
        ТабДок.Вывести (ОбластьПодвал) ;

        //}} КОНСТРУКТОР_ВЫХОДНЫХ_ФОРМ
    КонецПроцедуры

```

Отчет можно полностью собрать конструктором и только слегка изменить - описать параметры запроса (счет и корсчет), убрать цикл по детальной выборке.

### **Практикум №4**

*Этот отчет был придуман с целью анализа прибыльности работы с клиентом. Однако, по нашему мнению, он не в полной мере для этого подходит, т.к. не учитывает ситуации, когда мы возвращаем деньги клиенту обратно. Измените отчет так, чтобы он содержал колонки Клиент, Выручка (уже есть), Возвраты (сколько мы вернули клиенту денег, т.е. Дт. Дебиторка, Кт Касса) и Результат (разница между Выручкой и Возвратом)*

### **Практикум ;5**

*Разработайте отчет «Анализ счета». Отчет должен показывать остаток на начало (см. отчет «Карточка счета»), обороты за период общие по счету и детальные по корсчетам (таблица «Обороты») и остаток на конец.*



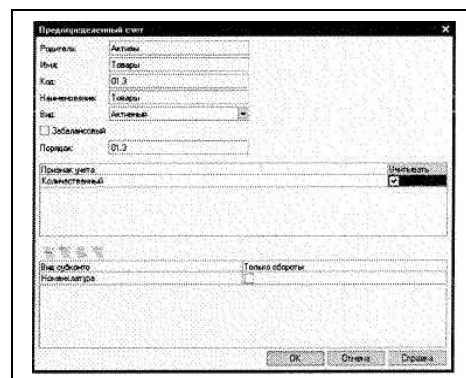
## Количественный учет

Под количественным учетом будем понимать учет материальных ценностей в количественном (натуральном) выражении. Таким образом, к нашему учету в денежном выражении добавляется еще один вид учета. Он имеет смысл не на всех счетах, а только на счетах материальных ценностей и только на тех из них, на которых ведется учет аналитический. Какие изменения нужно сделать в конфигурации.

### Изменения в плане счетов

Во-первых, необходимо отличать счета, по которым будет вестись количественный учет, от тех, где это не имеет смысла. Для этого в объекте «ПланыСчетов.Основной» на закладке «Данные» добавляем новый признак учета - «Количественный»

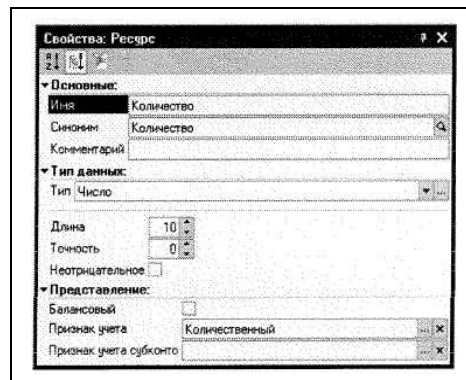
Там же (ПланыСчетов.Основной) на закладке «Прочие» вносим изменения в настройку предопределенных счетов. Для счета «Товары» устанавливаем признак количественного учета.



### Изменение в регистре

В регистре бухгалтерии добавляем новый ресурс «Количество», не балансовый, который будет задействован для хранения данных в натуральном выражении по счетам с признаком «Количественный».

Отличие «Балансовых» ресурсов от «Небалансовых» состоит в том, что балансовый ресурс в одной записи - один и, стало быть, баланс по нему обязателен (одна и та же сумма была списана с одного счета и зачислена на другой). Негалансовых ресурсов и измерений на одну запись существует два - для дебета и кредита, т.е. при желании можно с одного счета списать 10 штук, на другой положить 9 штук, потеряв 1 штуку по дороге. Для количества это в принципе возможно - когда мы списываем с одного счета материалы или полуфабрикаты, а на другой приходим готовое изделие.



## Изменения в документах

В документе «Операция» необходимо добавить две колонки в табличное поле набора записей регистра «Основной», расположенного в диалоге формы, КоличествоДт и КоличествоКт. Делается это так же, как в прошлый раз - с помощью «Размещения данных».

Документ «Приходная накладная» также требует корректировки, которую можно выполнить или «вручную», дописав одну строку в процедуру «Обработка проведения» модуля объекта, или с помощью конструктора движений.

```
Движение.КоличествоДт = ТекСтрокаТовары.Количество;
```

## Обращение к результатам

Если мы поставим себе задачу показать пользователю остатки товаров в расходной накладной при ее заполнении, единственный способ это сделать - выполнить запрос к регистру. Для этого нам потребуется

- Во-первых, добавить в табличное поле Товары новую колонку (имя Остаток).
- Во-вторых, написать обработчик события, которое возникает при выводе строки в табличное поле.

```
Процедура ТоварыПриВыводеСтроки(Элемент, ОформлениеСтроки,
ДанныеСтроки) //1
```

```
ОформлениеСтроки.Ячейки.Остаток.
```

```
УстановитьТекст (ПоказатьОстаток (ДанныеСтроки.Номенклатура) ); //2
```

```
КонецПроцедуры
```

Событие ПриВыводеСтроки возникает, когда строка выводится в табличное поле. В качестве параметра процедура получает объект типа «ОформлениеСтроки», имеющий свойства «шрифт, цвет текста, цвет фона» и коллекцию оформлений ячейки. Через этот параметр мы получаем доступ к коллекции ячеек, в том числе и к ячейке «Остаток».

- В-третьих, написать функцию, которая и будет выполнять запрос к регистру

```
Функция ПоказатьОстаток(Товар)
    Запрос = Новый Запрос; //1
    Запрос.Текст =
        "ВЫБРАТЬ
        | ОсновнойОстатки.КоличествоОстаток
        |ИЗ
        | РегистрБухгалтерии.Основной.Остатки (&МоментВремени,
Счет = &Счет, ,Субконто1 = &Субконто) КАК ОсновнойОстатки";

    Запрос.УстановитьПараметр ("МоментВремени", МоментВремени() ); //2
    Запрос.УстановитьПараметр ("Счет", ПланыСчетов.Основной.Товары);
    Запрос.УстановитьПараметр ("Субконто", Товар); Остатки =
    Запрос.Выполнить().Выбрать{}; //3 Остатки.Следующий();
```



```

Ост = Остатки.КоличествоОстаток; //4
Если Ост = NULL Тогда //5
    Возврат 0;
КонецЕсли;

Возврат Ост //6

КонецФункции

```

Запрос можно собрать конструктором, который вызывается по пункту контекстного меню "Конструктор запросов" Комментарии к модулю:

//1 Текст запроса построен конструктором. В качестве таблицы Мы выбрали таблицу Остатки, в которую в качестве параметров передаем МоментВремени документа, условие по счету и по субконто.

//2 устанавливаем значения параметров запроса

//3 выполняем запрос, открываем выборку результатов и позиционируемся на первой (и единственной в нашем случае) строке.

//4 обращаемся к полю таблицы по имени ресурса регистра (Количество) + «Остаток». Виртуальная таблица «Остатки» нашего регистра имеет поля КоличествоОстаток, КоличествоОстатокДт, КоличествоОстатокКт. Имя поля состоит из имени ресурса (как оно определено в свойствах регистра) и типа остатка. Мы могли воспользоваться или полем КоличествоОстатокДт (остаток дебетовый, имеет смысл для активного счета) или полем КоличествоОстаток (которое для любого счета показывает разницу между остатком дебетовым ж остатком кредитовым)

//5 если остатка нет, нам будет возвращено значение типа NULL.

//6 показываем остаток пользователю

Если мы хотим ускорить работу механизма, это можно сделать за счет вынесения вычисляемого поля Остаток табличного поля в шапку документа. В этом случае у нас будет задействовано другое событие табличной части «При активизации строки»

Где  
//1 -  
есть  
эleme  
нт  
форм  
ы  
Остат  
ок1  
(имя  
Надп  
иси)  
//2 -  
есть  
рекви  
зит  
форм  
ы с  
имене  
м  
Остат  
ок1  
(имя  
надпи  
си  
значе  
ния  
не  
имеет  
)



**Практикум №6-**

Расположите в диалоге формы (в шапке) документа надпись, в которой пользователь сможет видеть себестоимость товара, рассчитанную как среднюю (остаток суммовой / остаток количественный)

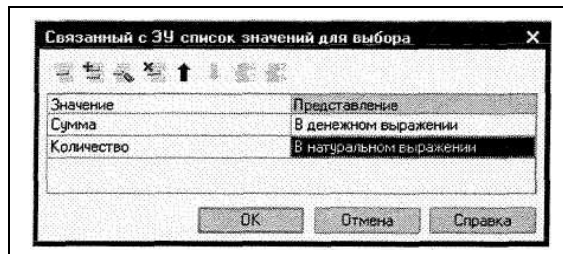
**Практикум №7**

Изменить существующий отчет «Движение товаров» таким образом, чтобы предоставить пользователю возможность выбора режима формирования отчета - в суммовом или количественном выражении. Решить задачу можно, например, разместив в диалоге формы отчета «поле выбора» для выбора пользователем вида учета. Чтобы добавить в поле выбора виды учета можно воспользоваться двумя способами:

Во-первых, можно разместить в теле модуля или в процедуре обработчике события формы «ПриОткрытии» следующий текст:

```
ЭлементыФормы.ВидУчета.СписокВыбора.Очистить ();
ЭлементыФормы.ВидУчета.СписокВыбора.Добавить ("Сумма", "В денежном
выражении");
ЭлементыФормы.ВидУчета.СписокВыбора.Добавить ("Количество", "В
натуральном выражении");
ЭлементыФормы.ВидУчета.Значение = "Сумма";
```

Во-вторых, можно ввести значения прямо в поле выбора. Для этого в окне свойств поля выбора, в разделе «Использование» щелкаем гиперссылку «Строки списка выбора» и заносим в список две строки.



Текст запроса придется изменить следующим образом:

```
ОсновнойОстаткиИОбороты."+ВидУчета+"НачальныйОстатокДт
КАК СуммаНачальныйОстатокДт,
```

Где ВидУчета - переменная, содержащая строку «Количество» или «Сумма», в зависимости от того, какой отчет сейчас хочет сформировать пользователь. Обратите внимание на псевдонимы (после КАК). Если они не заданы, то запрос будет называть поля «СуммаНачальныйОстатокДт» или «КоличествоНачальныйОстатокДт» в зависимости от вида отчета. А в макете таблицы остался параметр «СуммаНачальныйОстатокДт».



## Валютный учет

Для организации валютного учета необходимо, во-первых, определить на каких счетах его вести. Во-вторых, хранить в регистре по этим счетам две характеристики: сумму в валюте проводки (документа, сделки) и сумму в валюте учета (как эквивалент суммы в валюте проводки пересчитанный по курсу). В-третьих, разделить суммы в валюте сделки по валютам, чтобы иметь возможность получить отдельно сумму по каждой валюте.

В нашем примере будем считать, что валюта учета (валюта составления баланса) и национальная валюта (базовая валюта, относительно которой заводятся курсы в справочник валют, валюта, курс которой равен единице) - совпадают. Т.е. курс валюты учета равен единице.

## Изменения в плане счетов

В плане счетов на закладке «Данные» добавляем новый признак учета «Валютный», по которому будем отличать счета, требующие ведения учета в разных валютах.

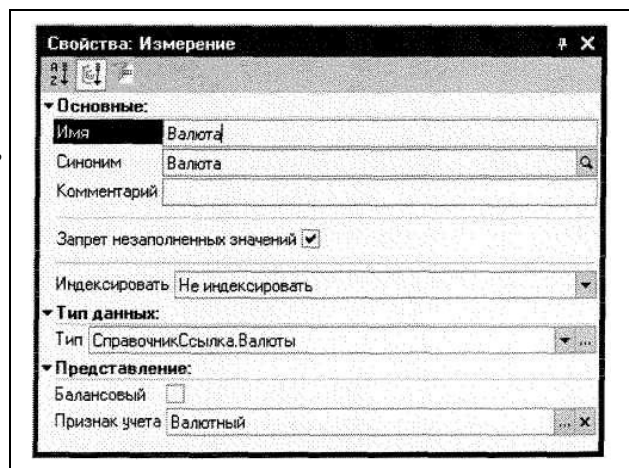
На закладке «Прочие» в списке предопределенных счетов устанавливаем для счета «Касса» признак учета «Валютный».

## Изменения в регистре бухгалтерии

В регистр бухгалтерии добавляем измерение «Валюта». Типа СправочникСсылка.Валют ы, небалансовое, для счетов с признаком учета «Валютный».

Небалансовое измерение позволит нам, например, в случае ведения взаиморасчетов в валюте проводить отгрузки

(закупки) в одной валюте, а оплаты - в другой. Небалансовое измерение создаст в таблице регистра два поля: для дебета и для кредита проводки, что позволит в одной проводки указать различные значения валют для дебета и кредита.



Кроме измерения в регистр добавляется новый ресурс «ВалютнаяСумма», также небалансовый, с признаком учета «Валютный» (признак учета субконто не заполняем).

## Изменения в документах

Внесения изменений потребует документ «Операция», где в форме, в табличном поле размещаются (размещение данных) колонки ВалютаДт, ВалютаКт, ВалютнаяСуммаДт, ВалютнаяСуммаКт.

Кроме того, необходимо организовать расчет суммы проводки, как эквивалента суммы в валюте сделки. В «Общем» модуле, создаем новую функцию:

```
Функция РассчитатьЭквивалент(ВалютнаяСумма, Валюта, Дата) Экспорт
Курс = РегистрыСведений.Валюты.ПолучитьПоследнее(Дата, Новый
Структура("Валюта", Валюта)).Курс; Возврат ВалютнаяСумма*Курс;
КонецФункции
```

В этой функции мы создаем «Структуру», в которую включаем один элемент Валюта со значением переданной валюты. Используя ее для отбора в регистре сведений, получаем структуру, содержащую ресурсы регистра сведений (в нашем случае структура содержит только одну запись) с последним на переданную дату значением курса по указанной валюте.

Эту функцию мы будем вызывать из модуля формы документа «Операция» для событий «При Изменении» полей ВалютаДт, ВалютнаяСуммаДт и ВалютаКт, ВалютнаяСуммаКт:

```
Процедура ОсновнойВалютаДтПриИзменении(Элемент) // из валюты и
валсуммы дт
```

```
    ТД = ЭлементыФормы.Основной.ТекущиеДанные;
```

```
    ТД.Сумма = РассчитатьЭквивалент(ТД.ВалютнаяСуммаДт,
ТД.ВалютаДт, Дата); КонецПроцедуры
```

```
Процедура ОсновнойВалютаКтПриИзменении(Элемент) // из валюты и
валсуммы ко?
```

```
    ТД = ЭлементыФормы.Основной.ТекущиеДанные;
```

```
    Если НЕ ТД.СчетДт.Валютный Тогда
```

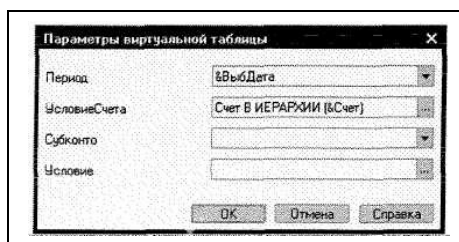
```
        ТД.Сумма = РассчитатьЭквивалент(ТД.ВалютнаяСуммаКт,
ТД.ВалютаКт, Дата);
```

```
        КонецЕсли;
```

```
КонецПроцедуры
```

## Получение результатов из регистров бухгалтерии

В качестве примера сформируем отчет, который покажет нам остатки в кассе по каждой валюте и эквивалент в валюте учета. Отчет



строится конструктором выходных форм.

Для формирования подобного отчета нам потребуется таблица остатков регистра бухгалтерии для которой необходимо определить параметры как на картинке.

Из таблицы отбираются поля: Валюта, ВалютнаяСуммаОстатокДт, СуммаОстатокДт. На закладке «Итоги» подводим Общий итог по полю СуммаОстатокДт.

### **Практикум №8**

*Сделать новый документ «РасчетКурсовыхРазниц», который в момент проведения должен анализировать валютный остаток по каждой валюте по счету касса и остаток в валюте учета. В случае, если результат от умножения валютной суммы на курс валюты (на дату документа) больше чем эквивалент этой валюты в валюте учета, необходимо отразить в учете положительную курсовую разницу*

*Дт. Касса / Кт. Капитал, по валюте на рублевую сумму (рассчитанный эквивалент минус рублевый остаток со счета)*

*В обратном случае — отрицательную курсовую разницу*

*Дт. Капитал /Кт. Касса*

*Для получения курса валюты имеет смысл воспользоваться виртуальной таблицей регистра сведений «СрезПоследних».*

## Консолидированный учет

Под консолидированным учетом понимается:

- Ведение учета холдинга в разрезе входящих в него юридических лиц, филиалов, дочерних компаний с получением отдельного баланса по каждой структурной единице и сводного в целом;
- ведение учета компании в разрезе центров учета и ответственности (при условии, что центры учета являются центрами инвестиций - самостоятельными бизнес единицами) с получением отчетности как в разрезе центров, так и сводного по компании в целом.

Если задача - получать баланс по каждому элементу консолидированного учета, значит - измерение должно быть «балансовым». В нашем случае, необходимо вести учет деятельности компании в разрезе фирм, по каждой из которых нужно составлять баланс и, кроме того, получать баланс по компании в целом. Для этого создаем новый справочник **«Фирмы»** иерархический.

### Изменения в регистре бухгалтерии

Добавляем в регистр бухгалтерии новое измерение «Фирма», типа СправочникСсылка. Фирмы, балансовое, без указания признака учета -т.е. данное измерение будет задействовано для всех счетов (в отличие от Валюты, которая ведется только по счетам с признаком учета Валютный).

### Изменения в документах

Изменений потребует и документ Операция и документ Приходная накладная.

#### Приходная накладная

В документы, которые формируют движения по регистру бухгалтерии, необходимо добавить новый реквизит «Фирма» типа СправочникСсылка. Фирмы. В модуль объекта, в процедуру ОбработкаПроведения добавляется одна строка:

```
Движение.Фирма = Фирма;
```

#### Документ Операция

При изменении документа «Операция» мы будем исходить из предпосылки, что в одной операции не может быть проводок по разным фирмам. Данное допущение вводится с целью упростить пример, хотя в реальной практике учета наличие в одной хозяйственной операции проводок по разным фирмам возможно. Как следствие, мы добавим в документ реквизит «Фирма», типа



СправочшшСсылка.Фирмы. Поместим его в диалоге. А в модуле объекта опишем обработчик события ПередЗаписью, которое позволит нам записать в движения регистра значение реквизита документа:

```
Процедура ПередЗаписью(Отказ, РежимЗаписи,  
РежимПроведения) Для каждого Проводка Из  
Движения.Основной Цикл Проводка.Фирма = Фирма;  
Проводка.Период = Дата; КонецЦикла; КонецПроцедуры
```

## Обращение к результатам

Обращение к итогам регистра бухгалтерии по Фирме осуществляется с помощью предложения запроса ГДЕ или с помощью условия, указываемого в параметре «Условие» виртуальных таблиц регистра. Так, например, чтобы отчет оборотно-сальдовая ведомость предоставил пользователю возможность получать данные и по фирме и сводно необходимо внести следующие изменения в отчет.

Во-первых, добавить в диалог формы новое поле ввода Фирма (типа СправочникСсылка.Фирмы).

Во-вторых, изменить процедуру обработчик нажатия кнопки (добавить в передаваемые параметры новый — Фирма).

В-третьих, изменить текст запроса

```
Запрос.Текст =  
    "ВЫБРАТЬ ...  
    | ИЗ  
    |     РегистрБухгалтерии.Основной.ОстаткиИОбороты (...)  
  
    | ГДЕ  
    |     | (ОсновнойОстаткиИОбороты.Счет.Забалансовый = ЛОЖЬ)  
    | И | (Фирма В Иерархии(&Фирма)) //1  
  
    ...";  
  
Запрос.УстановитьПараметр("Фирма", Фирма); //2
```

// 1. Новое условие запроса

// 2. новый параметр запроса «Фирма»

Альтернативный вариант - указать условие отбора в параметре виртуальной таблицы.

```
| из  
|  
РегистрВухгалтерии.Основной.ОстаткиИОбороты(&ДатаНачала  
, &ДатаОкончания, , , , Фирма В ИЕРАРХИИ{&Фирма})
```



## Практикум N&9

---

*Найдите все места в разработанной нами конфигурации, которые нуждаются в корректировке после добавления в наш учет разделения по фирмам. Внесите необходимые изменения в конфигурацию.*

Можно предположить, что использование условия в параметре виртуальной таблицы для отбора по значению измерения регистра является более рациональным, чем использование условия в предложении запроса «ГДЕ»



## Трехвалютный учет

В практике учета предприятий, входящих в холдинг или имеющих иностранных инвесторов может потребовать решения задача, когда баланс должен составляться не только в валюте учета, но и еще в одной валюте. Отличие от многовалютного учета состоит в том, что в данном случае в этой «третьей» (валюта учета, валюта сделки -первые две) должен составляться именно баланс, т.е. учет в этой валюте ведется на всех счетах наравне с валютой учета. Рассмотрим изменено в конфигурации, которые потребуются для этого сделать.

## Регистр бухгалтерии

Добавляем новый ресурс «СуммаХолдинга». Ресурс будет иметь признак «Балансовый» и не будет зависеть от признаков учета и признаков учета субконто.

## Изменения в справочниках и алгоритмах расчета

Необходимо каким-либо образом отметить валюту, в которой будет вестись баланс холдинга, и валюту, которая сейчас является валютой учета. Как это сделать?

Во-первых, можно сделать predetermined элементы справочника: ВалютаУчета и ВалютаХолдинга. Такое решение подходит при условии, что, во-первых, в конфигурации еще не заполнен справочник валют, и, во-вторых, эти две валюты не будут изменяться во времени. Т.е. не получится так, что в этом году мы ведем учет в рублях и долларах, а в следующем - в рублях и евро.

Воспользоваться константами в данном случае нельзя. Это бы решило проблему с уже заполненным справочником валют, но проблема смены валюты учета (валюты холдинга) в «живой» базе осталась бы: изменить значение константы можно, но что будет при перепроведении документа задним числом?

Остается вариант - сделать периодический регистр сведений для хранения значений валюты учета и валюты холдинга. Тогда пользователь сможет их выбрать из справочника валют и при изменении учетной политики предприятия - изменить без опасения, что могут «поплыть» данных прошлых периодов.

В методических целях мы разберем самую простую ситуацию -справочник валют еще не заполнен, и валюта учета и холдинга изменяться *же* могут.

В справочнике «Валюты» заводим новый predetermined элемент с именем «ВалютаУчета» и еще один «ВалютаХолдинга».

Обратите внимание на код нового предопределенного элемента. В справочнике уже есть элементы, и если мы заведем новый предопределенный элемент с кодом, который уже «занят» не предопределенным элементом, программа об этом предупредит, разрешит сохранить изменения, но в справочнике будет отключен признак уникальности кода.

Сумма в валюте холдинга по своей сути является эквивалентом суммы в валюте учета, пересчитанным по курсу валюты холдинга. Если валюта учета и национальная (базовая) валюта совпадают (т.е. курс валюты учета равен единице, а у нас именно так), то достаточно поделить сумму в валюте учета на курс валюты холдинга. Если задача сложнее и это три разные валюты (валюта учета = доллар, национальная валюта = рубль, валюта холдинга = евро), то формула будет чуть сложнее. И, кроме того, она будет включать в себя и первую ситуацию, т.е. будет более универсальной. Ее и реализуем.

Создадим новую экспортную процедуру в общем модуле:

```

Функция РассчитатьЭквивалентХолдинга(СуммаУчета, Дата) Экспорт
    ВалютаУчета = Справочники.Валюты.ВалютаУчета;
    ВалютаХолдинга = Справочники.Валюты.ВалютаХолдинга;

    ОтборПоВалютеУчета = Новый Структура("Валюта", ВалютаУчета);
    КурсВалютыУчета =
РегистрыСведений.Валюты.ПолучитьПоследнее(Дата,
ОтборПоВалютеУчета).Курс;

    ОтборПоВалютеХолдинга = Новый Структура("Валюта",
ВалютаХолдинга);
    КурсВалютыХолдинга = РегистрыСведений.
Валюты.ПолучитьПоследнее(Дата, ОтборПоВалютеХолдинга).Курс;

    Если КурсВалютыХолдинга = 0 Тогда
        Сообщить("Курс валюты холдинга не указан");
        Возврат 0;
    КонецЕсли;

    СуммаХолдинга =
СуммаУчета*КурсВалютыУчета/КурсВалютыХолдинга;

    Возврат СуммаХолдинга;

КонецФункции

```

Процедура не содержит новых методов и объектов и сильно напоминает расчет суммы учета от суммы в валюте сделки, рассмотренный в теме валютный учет.

## Изменения в документах - ввод данных в третьей валюте

Можно предположить, что сумма в валюте холдинга не должна просматриваться и редактироваться пользователем. Скорее всего, она



должна быть получена расчетным путем, записана в регистр и потом проанализирована в отчетах.

Можно предложить разные варианты, мы предложим два пути. В любом случае нужно будет перезаписать (перепровести) документы или пересчитать записи регистра обработкой.

### Документ Операция

Документ «Операция» содержит в модуле объекта процедуру обработчик события «ПередЗаписью». Процедура будет дополнена одной строкой:

```
Процедура ПередЗаписью(Отказ, РежимЗаписи,  
РежимПроведения) Для каждого Проводка Из  
Движения.Основной Цикл
```

```
Проводка.Фирма = Фирма;  
Проводка.Период = Дата;  
Проводка.СуммаХолдинга =  
РассчитатьЭквивалентХолдинга(Проводка.Сумма, Дата);
```

```
КонецЦикла;  
КонецПроце  
дуры
```

### Прочие документы

Можно пойти другим путем и передать всю коллекцию наборов записей документа из модуля объекта в общий модуль, где в каждое движение добавить заполнение нового ресурса. Тогда в процедуре ОбработкаПроведения каждого документа перед записью движений добавляется вызов новой экспортной процедуры общего модуля

```
ЗаполнитьСуммыХолдинга(Движения.Основно  
й); // записываем движения регистров  
Движения.Основной.Записать();
```

Процедура общего модуля:

```
Процедура ЗаполнитьСуммыХолдинга(Проводки) Экспорт  
Для каждого Проводка Из Проводки Цикл  
Движение.СуммаХолдинга =  
РассчитатьЭквивалентХолдинга(Проводка.Сумма, Проводка.Период)  
КонецЦикла;  
КонецПроцедуры
```

### Обращение к результатам

Изменим для примера только один отчет — Оборотно-сальдовую ведомость. Начнем с того, что предоставим пользователю возможность выбора валюты составления баланса - в валюте учета или в валюте холдинга. Для этого добавим в диалог формы новое поле

выбора . Имя элемента формы - «СуммаБаланса».

В модуле формы опишем новый обработчик события «ПриОткрытии», и прикрепим его к свойству формы «При

откр  
ытии  
»

```

Процедура ПриОткрытии()
Для Каждого Ресурс Из
Метаданные.РегистрыБухгалтерии.Основной.Ресурсы Цикл
Если Не Ресурс.Балансовый Тогда
    Продолжить;
КонецЕсли;
ЭлементыФормы.СуммаБаланса.СписокВыбора.Добавить (Ресурс.Имя,
Ресурс.Синоним); КонецЦикла;
ЭлементыФормы.СуммаБаланса.Значение =
ЭлементыФормы.СуммаБаланса.СписокВыбора[0].Значение;
КонецПроцедуры

```

Процедура, используя свойство глобального контекста «Метаданные» позволит нам получить все балансовые ресурсы регистра бухгалтерии, их имена и синонимы. Синонимы будут выведены пользователю в список выбора в качестве представлений.

А имена потребуются нам для формирования текста запроса (вспомним, что имена полей виртуальных таблиц формируются как ИмяРесурсы+Остаток/ОстатокДт/Оборот... и т.д.).

Там же в модуле формы изменим вызов процедуры формирования отчета, передав туда еще один параметр.

```

ОтчетОбъект.СформироватьОборотку(ТабДок, Дата1, Дата2, Фирма,
СуммаБаланса);

```

В саму процедуру формирования запроса также придется внести изменения.

```

Процедура СформироватьОборотку(ТабДок, Дата1, Дата2, Фирма,
СуммаБаланса) Экспорт //1

```

```

Макет = ПолучитьМакет("Оборотка");
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
|     ОсновнойОстаткиИОбороты.Счет КАК Счет,
|     ОсновнойОстаткиИОбороты.Счет.Представление,
|     ОсновнойОстаткиИОбороты."+СуммаВаланса+"НачальныйОстатокДт

КАК СуммаНачальныйОстатокДт, //2
|     ОсновнойОстаткиИОбороты."+СуммаБаланса+"НачальныйОстатокКт

КАК СуммаНачальныйОстатокКт,
|     ОсновнойОстаткиИОбороты."+СуммаБаланса+"ОборотДт КАК

СуммаОборотДт,
|     ОсновнойОстаткиИОбороты."+СуммаБаланса+"ОборотКт КАК

СуммаОборотКт,
|     ОсновнойОстаткиИОбороты."+СуммаБаланса+"КонечныйОстатокДт

КАК СуммаКонечныйОстатокДт,
|     ОсновнойОстаткиИОбороты."+СуммаБаланса+"КонечныйОстатокКт

КАК СуммаКонечныйОстатокКт

```

ИЗ





```
| РегистрВухгалтерии. Основной.ОстаткиИОбороты(&Дата1, &Дата2
, , , , Фирма В ИЕРАРХИИ(&Фирма)) КАК ОсновнойОстаткиИОбороты
// остальная часть модуля не изменяется
```

Комментарии:

// 1 - новый параметр, куда мы получим строку - имя ресурса, по которому строить баланс

//2 в описании полей таблицы в тексте запроса добавляется указание на ресурс

Отчет готов.

Оборотно-сальдовая ведомость						
Счет	Нач ост дт	Нач ост кт	Оборот дт	Оборот кт	Кон ост дт	Кон ост кт
Активы			576,51	576,51		
Товары			576,51	465,71	110,8	
Дебиторка			19,37			19,37
Касса			28,57	437,14	-408,57	
Обязательства			528,57	28,57	500	
Поставщики				19,37		19,37
Капитал				91,43		91,43

68

### Практикум №10

Измените отчеты «Анализ счета» и «Карточка счета» с целью предоставления пользователю возможности формирования отчетов как в валюте учета, так и в валюте холдинга.

## Расчет себестоимости

### Проведение документа с анализом данных регистра

Документ приходная накладная в нашей конфигурации не требует анализа состояния учета в момент проведения - он просто увеличивает нашу задолженность перед поставщиком и товары на счете товаров. С расходной - другая ситуация...

Документ должен проверить наличие товара (есть ли он вообще, и если есть - хватает ли), рассчитать себестоимость (мы будем списывать товар по средней на момент записи документа) и списать товар со счета товаров по рассчитанной себестоимости.

### Проведение документов в различных версиях «1С:Бухгалтерии»

Чтобы лучше понять принципы работы системы «1С:Предприятие 8.0» проанализируем реализацию алгоритма расчета себестоимости в более ранних версиях.

#### **1С:Бухгалтерия 6.0**

«Шестерка» и более ранние системы позволяли нам проанализировать наличие товара на счете в суммовом и количественном выражении на конец месяца (обычно поступали именно так, без использования временного расчета). Алгоритм проведения включал в себя цикл по строкам документа, внутри которого мы обращались к итогам на конец месяца, получали остатки (количественный и суммовой), рассчитывали себестоимость и делали проводки. В конце месяца документы перепроводили и добивались, таким образом, правильной себестоимости товаров за месяц.

#### **1С:Предприятие 7.5**

Сохранив возможность использования основного расчета (на конец месяца), новая платформа предоставила новую возможность - выполнения запроса к таблицам бухгалтерских итогов. Временный расчет мы не рассматриваем, т.к. его обычно не использовали в модуле проведения - слишком медленно, ведь в нем не было возможности указать по каким товарам нужно выполнить расчет итогов на дату документа, стало быть, расчет выполнялся по всем (а у нас их может быть несколько десятков тысяч...). Теперь мы могли списать товар по средней себестоимости на момент проведения документа.

Алгоритм проведения включал в себя цикл по строкам документа, внутри которого (внимание!) выполнялся запрос к бухгалтерским



итогам по тому товару, строка которого обрабатывается циклом. Таким образом, если в накладной было 100 строк, запрос к базе данных выполнялся 100раз.... Недостаток очевиден.

### 1С:Предприятие 7.7

Среди новых возможностей система «1С:Предприятие 7.7» предоставило возможность выполнить запрос не по одному товару (внутри цикла), и не по всем (нескольким тысячам), а по нужному массиву товаров. Этот массив мы могли получить, выгрузив колонку табличной части накладной и передав в запрос. Далее выполнить запрос по этим товарам. В результате мы получали в памяти машины табличку, содержащую остатки товаров и остатки товаров, что-то вроде:

Субkonto	ОстКол	ОстСум
Школьная	5	50
Паркер	3	300

Плюс к этому мы имели табличную часть самой накладной

Товар	Количество	Сумма
Школьная	2	30
Паркер	1	150

70

Алгоритм проведения состоял из цикла по строкам накладной (вторая таблица, содержащая все товары, даже те, остатки по которым отсутствовали в итогах) и «заглядывания» в первую таблицу (итоги) чтобы получить остаток количественный (ОстКол) для проверки наличия и остаток суммовый (ОстСум) для расчета себестоимости.

### 1С:Предприятие 8.0

Теперь у нас есть возможность объединить две таблицы (документа и итогов регистра) в одну используя запрос. Таким образом, мы будем «бегать» же по двум, а по одной таблице, содержащей все необходимые данные для формирования проводок.

Таким образом, наша цель получить в результате выполнения запроса таблицу:

Товар	Количество	Сумма	ОстКол	ОстСум
Школьная	2	30	5	50
Паркер	1	150	3	300

### Алгоритм проведения в «1С:Предприятие 8.0»

Начнем с написания запроса. Это, наверное, самый сложный участок, поэтому будем решать задачу «по частям».

## Запрос по таблице документа

Первым делом напишем запрос, который позволит нам получить таблицу с данными документа (товар, количество, сумма продажи)

```

ВЫБРАТЬ
Док.Номенклатура,
Док.Количество,
Док.Сумма
ИЗ
Документ.Расходная.Товары Как Док
ГДЕ
Док.Ссылка = &Ссылка //1

```

/// в качестве параметра передаем ссылку на документ.

Мы выполняем запрос по таблице Товаров Расходной накладной выбранной в диалоге, и «вынимаем» из нее товар, количество и сумму.

Товар	Количество	Сумма
Школьная	2	30
Паркер	1	150
Пустой	1	10

## Запрос по таблице бухгалтерского регистра

Из всех таблиц бух регистра нам лучше всего подходит таблица Остатки. Запрос по таблице Остатки будет выглядеть так:

```

ВЫБРАТЬ
Ост.Субконто1,
Ост.КоличествоОстатокДт Как ОстКол,
Ост.СуммаОстатокДт Как ОстСум
ИЗ
РегистрБухгалтерии.Основной.Остатки(&Момент,Счет в
Иерархии(&Счет),,Субконто! в (&Субконто) И Фирма = йФирма) Как Ост

```

Где, счет - счет учета товаров, фирма - значение из справочника «фирмы», а Субконто - массив, содержащий все товары накладной.

Результат - таблица:

Субконто1	ОстКол	ОстСум
Школьная	5	50
Паркер	3	300

## Соединяем две таблицы в одну

```

ВЫБРАТЬ
Док.Номенклатура,
Док.Количество,
Док.Сумма,
Ост.КоличествоОстатокДт Как ОстКол,
Ост.СуммаОстатокДт Как ОстСум
ИЗ
Документ.Расходная.Товары Как Док
ЛЕВОЕ ВНЕШНЕЕ СОЕДИНЕНИЕ
РегистрБухгалтерии.Основной.Остатки(&Момент,Счет в
Иерархии(&Счет),,Субконто1 в (&Субконто) И Фирма = &Фирма) Как Ост
ПО
Док.Товар = Ост.Субконто1

```



ГДЕ

(Док.Ссылка = &Ссылка)

Комментарии:

Мы используем Левое внешнее соединение и добавляем в таблицу документа (которая содержит все строки), таблицу итогов (которая будет содержать только те субконто, по которым есть остатки). Если для какого-то товара остатков нет, поля ОстКол и ОстСум будет содержать NULL и объединяем их по полям Товар=Субконто 1.

Результат:

Товар	Количество	Сумма	ОстКол	ОстСум
Школьная	2	30	5	50
Паркер	1	150	3	300
Пустой	1	10	NULL	NULL

## Обработка проведения

Осталось скопировать запрос в модуль объекта Документ.Расходная и дописать код для записи новых движений в регистр:

Процедура ОбработкаПроведения(Отказ, РежимПроведения)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ До к .

Номенклатура, |

Док.Количество, |

Док.Сумма,

|Ост.КоличествоОстатокДт Как

ОстКол, | Ост.СуммаОстатокДт Как

ОстСум |ИЗ

| Документ.Расходная.Товары Как Док

ЛЕВОЕ ВНЕШНЕЕ СОЕДИНЕНИЕ

| РегистрБухгалтерии.Основной.Остатки&Момент, Счет в

Иерархии(&Счет),,Субконто1 в (&Субконто) ИФирма = &Фирма) Как

Ост |ПО

|Док.Товар =

Ост.Субконто1 |ГДЕ |

(Док.Ссылка = &Ссылка)";

РезультатЗапроса = Запрос.Выполнить();

Запрос.УстановитьПараметр("Субконто"

, Товары.ВыгрузитьКолонку("Номенклатура"));

Запрос.УстановитьПараметр("Момент", МоментВремени());

Запрос.УстановитьПараметр("Счет",

ПланыСчетов.Основной.Товары);

Запрос.УстановитьПараметр("Ссылка", Ссылка);

Запрос.УстановитьПараметр("Фирма", Фирма);

СтрокаТаблицы = РезультатЗапроса.Выбрать(); //1

Пока СтрокаТаблицы.Следующий() Цикл //2

Товар = СтрокаТаблицы.Номенклатура; //3

КолВо = СтрокаТаблицы.Количество; //4

Сумма = СтрокаТаблицы.Сумма; // 5

ОстСум = СтрокаТаблицы.ОстСум; // 6



```

ОстКол = СтрокаТаблицы.ОстКол; // 7

Если КолВо <= 0 Тогда //8
    Продолжить;
КонецЕсли;

Если ОстКол = Null Или ОстКол < 0 Тогда //9
Сообщить ("Товара "+Товар.Наименование*" нет в наличии",
СтатусСообщения.ОченьВажное);
    Отказ = Истина;
    Возврат;

ИначеЕсли ОстКол < КолВо Тогда //10
Сообщить ("Товара "+Товар.Наименование+" не достаточно",
СтатусСообщения.ОченьВажное);
    Отказ = Истина;
    Возврат;

ИначеЕсли ОстКол = КолВо Тогда //11
    СуммаПроводки = ОстСум;

Иначе //12
    Себестоимость = ОстСум / ОстКол;
    СуммаПроводки = Себестоимость*КолВо
КонецЕсли;

Движение = Движения.Основной.Добавить(); //13
Движение.СчетДт = ПланыСчетов.Основной.Капитал;
Движение.СчетКт = ПланыСчетов.Основной.Товары;
Движение.Период = Дата; Движение.Регистратор =
Ссылка; Движение.Фирма = Фирма; Движение.Сумма =
СуммаПроводки; Движение.КоличествоКт = КолВо;

Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.
Номенклатура] = Товар;
Движение.Содержание = "Списана себестоимость";

КонецЦикла;

Движение = Движения.Основной.Добавить(); //14
Движение.Период = Дата; Движение.Регистратор =
Ссылка;

Движение.СчетДт = ПланыСчетов.Основной.Дебиторка;
Движение.СубконтоДт[ПланыВидовХарактеристик.ВидыСубконто.Кон
трагенты] = Покупатель;
Движение.СчетКт = ПланыСчетов.Основной.Капитал;
Движение.Сумма = Товары.Итог("Сумма");
Движение.Фирма = Фирма; Движение.Содержание =
"Продажа";

Движения.Основной.Записать ();
КонецПроцедуры

```

Комментарии к модулю:

Запрос остался тот же, разве что изменились параметры (теперь мы находимся в объекте ДокументОбъект, а не в отчете).

//1 получили выборку из результата запроса



//2 вошли в цикл по этой выборке  
//3 товар из табличной части документа  
//4 продаваемое количество товара из документа  
//5 сумма, за которую мы продаем это количество товара из табличной части документа (на самом деле, в нашем примере - не нужна, понадобится, если нужно будет делать проводку по реализации товара по каждой строке документа)  
//6 суммовой остаток из регистра  
//7 количественный остаток из регистра  
//8 в строке документа не указано количество, пропускаем  
//9 ОстКол может быть равен NULL, если этого товара вообще нет в регистре, или меньше нуля, если есть, но перерасходован. В этом случае присваиваем параметру процедуры ОбработкаПроведения Отказ значение Истина - документ проведен не будет.  
//10 так же поступаем, если товара не хватает.  
//11 если осталось (количественный учет) столько, сколько нам нужно - в сумму проводки пойдет весь суммовой остаток из регистра  
//12 если осталось больше - надо считать среднюю  
//13 добавляем новое движение в регистр и заполняем его  
//14 кроме того делаем движение на сумму продажи товаров (одна на всю накладную)

## Альтернативный вариант обработки проведения документа

На момент написания методики отсутствовали типовые решения, реализующие задачу автоматизации бухгалтерского учета, поэтому в рамках этого курса мы лишь предлагаем возможные подходы к решению задач. В продолжение темы применения запроса для расчета себестоимости, можно предложить следующий вариант алгоритмы проведения:

```
Процедура ОбработкаПроведения(Отказ,  
    РежимПроведения) // запрос по шапке документа  
    ЗапросПоШапке = Новый Запрос;  
    ЗапросПоШапке.Текст = "ВЫБРАТЬ  
        Расходная.Дата,  
        Расходная.Контрагент Как Покупатель,  
        Расходная.СуммаДокумента,  
        Расходная.Ссылка,  
        Расходная.Фи  
    рма ИЗ
```

Докуме  
нт. Рас  
ходная  
КАК  
Расход  
ная

---

```

ГДЕ
    Расходная.Ссылка = &Ссылка";

ЗапросПоШапке.УстановитьПараметр{"Ссылка", Ссылка);
Шапка = ЗапросПоШапке.Выполнить().Выбрать();
Шапка.Следующий();

// текст вложенного запроса, который вернет неповторяющиеся
товары из таблицы документа
СписокРазличныхТоваров =
"ВЫБРАТЬ РАЗЛИЧНЫЕ
|      РасходнаяТовары.Номенклатура Как Товар
|ИЗ
|      Документ.Расходная.Товары КАК РасходнаяТовары

ГДЕ
    РасходнаяТовары.Ссылка = &Ссылка";

// запрос по таблице документа и регистру бухгалтерии
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
|      РасходнаяТовары.Номенклатура КАК Товар,
|      РасходнаяТовары.Количество КАК Количество,
|      РасходнаяТовары.Сумма КАК Сумма,
|      ОсновнойОстатки.КоличествоОстаток КАК Остаток,
|      ВЫБОР КОГДА ОсновнойОстатки.КоличествоОстаток = 0
ТОГДА 0
|      КОГДА ОсновнойОстатки.КоличествоОстаток =
РасходнаяТовары.Количество ТОГДА ОсновнойОстатки.СуммаОстаток
|      ИНАЧЕ ОсновнойОстатки.СуммаОстаток /
ОсновнойОстатки.КоличествоОстаток * РасходнаяТовары.Количество
|      КОНЕЦ КАК Стоимость
|ИЗ
|      Документ.Расходная.Товары КАК РасходнаяТовары
|      ЛЕВОЕ СОЕДИНЕНИЕ
РегистрВухгалтерии.Основной.Остатки(&МоментВремени, Счет = &Счет,
, Субконто1 В ("СписокРазличныхТоваров+") И Фирма = &Фирма) КАК
ОсновнойОстатки
|      ПО РасходнаяТовары.Товар =
ОсновнойОстатки.Субконто1

ГДЕ
    РасходнаяТовары.Ссылка = &Ссылка";

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());
Запрос.УстановитьПараметр("Счет", ПланыСчетов.Основной.Товары);
Запрос.УстановитьПараметр("Ссылка", Ссылка);
Запрос.УстановитьПараметр("Фирма", Фирма);

Строка = Запрос.Выполнить().Выбрать();
Пока Строка.Следующий() Цикл

    Если Строка.Остаток = Null Или Строка.Остаток < 0
Тогда
        Сообщить ("Товара "+Строка.Товар.Наименование*"
нет в наличии", СтатусСообщения.ОченьВажное);
        Отказ = Истина;
        Возврат;

```



ИначеЕсли Строка.Остаток < Строка.Количество Тогда  
Сообщить ("Товара "+Строка.Товар.Наименование\*" не достаточно",  
СтатусСообщения.ОченьВажное); Отказ = Истина; Возврат;

КонецЕсли;

Движение = Движения.Основной.Добавить ();  
Движение.СчетДт =  
ПланыСчетов.Основной.Капитал;  
Движение.СчетКт = ПланыСчетов.Основной.Товары;  
Движение.Период = Шапка.Дата;  
Движение.Регистратор =  
Шапка.Ссылка; Движение.Фирма =  
Шапка.Фирма;  
Движение.Сумма = Строка.Стоимость;  
Движение.КоличествоКт =  
Строка.Количество;

Движение.СубконтоКт [ПланыВидовХарактеристик.ВидыСубконто.  
Ном енклатура] = Строка.Товар;

Движение.Содержание = "Списана себестоимость";

КонецЦикла;

Движение = Движения.Основной.Добавить  
(); Движение.Период = Шапка.Дата;  
Движение.Регистратор = Шапка.Ссылка;

Движение.СчетДт = ПланыСчетов.Основной.Дебиторка;  
Движение.СубконтоДт [ПланыВидовХарактеристик.ВидыСубконто.Кон  
трагенты] = Шапка.Покупатель;

Движение.СчетКт =  
ПланыСчетов.Основной.Капитал; Движение.Сумма  
= Шапка.СуммаДокумента; Движение.Фирма =  
Шапка.Фирма; Движение.Содержание = "Продажа";

Движения.Основной.Записать  
(); КонецПроцедуры

Все тексты запросов написаны с помощью конструктора запросов. Основное отличие от предыдущего варианта - полный отказ от использования объектной модели (ссылок) для формирования движежий документа и выполнение расчета в тексте запроса (а не при обработке его результатов)

## Формирование движений разными датами

В учете встречаются ситуации, когда одно событие в хозяйственной жизни предприятия должно порождать несколько проводок, причем разными датами. Пример: когда дата отгрузки продукции (оказания услуг) отличается от даты реализации этой продукции или услуг. Так, например, проводки по отгрузке продукции должны быть сформированы в момент реального отпуска продукции со склада (тогда же и оформляется расходная накладная), а реализация (т.е.

доходы) возникает, например, когда оплачены деньги, или когда истекает гарантийный срок и товар уже не может быть возвращен. Для услуг в турагентской деятельности это могут быть дата продажи путевки и дата завершения тура (или выселения из гостиницы...). Как решить эту задачу? Или двумя документами, но тогда одна операция (продажа товара) «распадается» на две. Или «научить» документ делать проводки разными датами. «1С:Предприятие 8.0» позволяет это организовать.

Для этого в расходной накладной добавляем новый реквизит -«ДатаРеализации» (тип «Дата», Дата+Время), размещаем его в форме, и изменяем модуль объекта, процедура ОбработкаПроведения, в том месте, где формируется проводка по реализации товаров:

```
// закончился цикл по строкам документа
    Если ДатаРеализации = '0001-01-01'- Тогда
        Возврат;
    КонецЕсли;

Движение = Движения.Основной.Добавить(); //14
Движение.Период = ДатаРеализации; Движение.Регистратор
= Ссылка; // дальше - как было
```

Проверив, заполнена ли ДатаРеализации, мы формируем проводку, где указываем, что свойство Период записи регистра должно заполняться не датой документа, а датой реализации.

## Оперативное и неоперативное проведение

Документы, как правило, проводятся или в реальном времени, или задним числом. Если документ проводится без нарушения нормальной хронологической последовательности документов (т.е. за предшествующим), такое проведение считается оперативным. И при таком проведении, например, документа «расходная накладная» обязательно надо проверить, хватает ли товаров на складе, какая задолженность и проч. Если документ, который на настоящий момент является последним, перезаписывается, возможны варианты (можно провести документ или в оперативном или в неоперативном режиме). Если документ вводится задним числом, то можно предположить, что он фиксирует уже состоявшийся факт, и раз он состоялся, значит, лишние проверки можно исключить.

При оперативном (в реальном времени) проведении документа можно предположить ситуацию, когда при одновременной работе нескольких пользователей товар может быть списан дважды, потому что в момент проведения документа одного пользователя, второй успел провести свой, списывающий этот же товар.

Чтобы избежать такой ситуации, имеет смысл добавить в запрос обработки проведения документа предложение «ДЛЯ ИЗМЕНЕНИЯ»,



которое позволит заблокировать от чтения для других пользователей таблицы регистра на момент проведения документа.

В документе «Расходная накладная» на закладке свойств «Движения» устанавливаем свойство «Оперативное проведение» в значение «Разрешить» и изменяем запрос:

```
Если РежимПроведения = РежимПроведенияДокумента.Оперативный  
Тогда  
Запрос.Текст = Запрос.Текст +  
"ДЛЯ ИЗМЕНЕНИЯ РегистраБухгалтерии.Основной.Остатки";  
КонецЕсли;
```

## Признаки учета субконто

В программе предусмотрена возможность отключать ведение отдельных видов учета по отдельным срезам аналитики. Осуществляется это с помощью признаков учета субконто. Изначально предусмотрен один признак - «Только обороты», который позволяет отключить ведение остатков в разрезе значений субконто в регистре.

### Только оборотное субконто

Предусмотрим ведение учета на счета Касса в разрезе статей денежных потоков. Использовать для этого «обычно нельзя - по е» статьям мы субконто одним будем получать деньги в кассу, по другим - списывать.

Признак учета	Учитывать
Количественный	<input type="checkbox"/>
Валютный	<input checked="" type="checkbox"/>

новый

Создадим справочник «ДенежныеПотоки».

В объекте ПланыВидовХарактеристики ВидыСубконто на закладке «Основные» добавим тип данных (СправочникСсылка.ДенежныеПотоки) в составной тип данных плана видов характеристик.

На закладке «Прочие» в окне редактирования предопределенных значений добавим еще одну предопределенную характеристику ДенежныеПотоки типа СправочникСсылка.ДенежныеПотоки.

В плане счетов «Основной» в списке предопределенных счетов изменим настройки счета Касса: добавим аналитический срез ДенежныеПотоки и установим для него признак «Только обороты»

Осталось ввести несколько проводок по счету «Касса» с указанием в качестве субконто элемента справочника «ДенежныеПотоки» и можно посмотреть результаты.

Убедиться в том, что программа не хранит остатки в разрезе субконто, можно построив с помощью конструктора отчет по таблице ОстаткиИОбороты. По счету в целом можно будет увидеть и остатки и обороты, по статьям - только обороты.

## Практикум МП

80

Постройте отчет о движении денежных средств, который покажет пользователю остаток на начало (синтетический по счету

в целом), все поступления в разрезе статей и итог по ним, все списания в разрезе статей и итог по ним, остаток денежных средств на конец периода.

### Дополнительные признаки учета субконто

Поставим себе задачу - нам необходимо хранить обороты по статьям движения денежных средств только в валюте учета, т.е. не требуется хранить данные в различных валютах.. А мультивалютный учет на счете нам нужен только для учета синтетического. Каким образом отключить ведение валютного учета по субконто, оставив его по счету в целом?

В плане счетов «Основной» на закладке «Субконто» заводим новые признаки учета субконто: Суммовой и Валютный.

В плане счетов «Основной» на закладке «Прочие» изменяем настройку предопределенного счета «Касса» - снимаем флаг «Валютный» для субконто «Денежные потоки», оставив флаг «Валютный» (признак учета) для счета в целом.

В регистре «Основной» для ресурса «Сумма» устанавливаем признак учета субконто «Суммовой», а для ресурса «ВалютнаяСумма» - «Валютный». Таким образом, мы установили правило хранения итогов по этим ресурсам для различных субконто.

Общ. порядок:

- 1) Заводим признак учета
- 2) Привязываем признак к плану счетов и к регистру учета
- 3) Определяем признак для нужных счетов.

Теперь, мы сможем видеть остатки и обороты по счету в целом, как в валюте, так и в валюте учета. А обороты по статьям - только в валюте учета.

## Табличный документ - усовершенствуем отчеты

Отчет - средство получения и предоставления пользователю информации в удобном виде. Как средство получения информации, они нам уже знакомы. Обратим теперь внимание на то, чтобы эта информация была представлена в «удобном виде»


### Расположение табличного документа в конфигурации

Табличный документ в конфигурации может быть расположен в списке макетов объекта, среди общих макетов (ветка «Общие» дерева объектов конфигурации), как внешний файл (Главное меню - Файл -Новый - Табличный документ) или прямо в диалоге формы.

Все бухгалтерские отчеты, которые были нами разработаны, создавались конструктором, а он «предпочитает» создавать табличный документ и открывать его в отдельном окне.

Первая наша задача - расположить табличный документ в диалоге формы отчета.

Среди отчетов, которые мы разработали, есть оборотка (оборотно-сальдовая ведомость), изменим отчет так, чтобы табличный документ располагался в диалоге формы. Для этого необходимо:

1. Добавить в форму элемент управления «Поле табличного документа»  Поле табличного документа. И задать ему имя. Чтобы это сделать нужно, выделить весь табличный документ (а не одну ячейку). Поступаем так: выделяем ячейку и нажимаем «Esc». Можно оставить имя по умолчанию - ПолеТабличногоДокумента1.
2. Изменить процедуру формирования отчета так, чтобы не создавать новый табличный документ, а использовать элемент формы, который надо предварительно очистить.

```
//ТабДок = Новый ТабличныйДокумент;// как было
ТабДок = ЭлементыФормы.ПолеТабличногоДокумента1;
ТабДок.Очистить();
```

### Практикум №12

*Измените также (разместив табличный документ в диалоге формы) отчет «Анализ счета»*

### Сводная таблица для построения кросс-отчетов

Кросс-отчеты большинству бухгалтеров знакомы «под именем» шахматных отчетов. Значительная часть бухгалтерских отчетов имеют шахматную форму. Это, конечно, сама «шахматка» (или шахматный



баланс), журнальный ордер, ведомость по счету, мемориальный ордер и многие другие. Мы не настаиваем на использовании того или другого термина (кросс-отчет или шахматный отчет) и оставляем выбор терминологии на усмотрение специалиста, выполняющего обследование и беседующего с постановщиком задачи.

Сводная таблица предоставляет пользователю возможность интерактивного анализа многомерных данных. Эти данные получаются как результат выполнения запроса. Располагается сводная таблица в табличном документе. Разработаем новый отчет, аналогичный отчету «Сводные проводки», который позволит анализировать обороты между счетами в сводной таблице.

Отчет будет выполнять запрос по таблице ОборотыДтКт. В параметрах указываем интервал дат. В отчет включаем поля СчетДт, СчетКт и СуммаОборот. На закладке итоги подводим итоги по полям СчетДт и СчетКт, суммируя значение поля СуммаОборот. В выходную форму добавляем поля типа «Дата» для параметров Дата1 и Дата2.

В макет отчета добавляем сводную таблицу. Для этого выделяем ячейку, которая станет верхним левым углом этой таблицы, и через меню «Таблица - Встроенные таблицы - Вставить сводную таблицу», размещаем ее в табличном документе.

Чтобы установить имя новой сводной таблицы, необходимо установить курсор на любую ее ячейку сводной таблицы и в палитре свойств, в верхнем поле выбора вместо «Ячейка» указать СводнаяТаблица1 (именно так называется таблица по умолчанию). Можно изменить имя и настроить свойства таблицы. Поля будут отображаться по умолчанию, авто-фиксация - позволит зафиксировать колонки и строки этих полей.

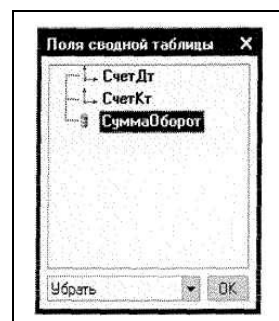
Выделяем группу строк, в которых располагается сводная таблица и создаем именованную область. Имя - «Сводная»

Чтобы заполнить таблицу данными запроса, нужно изменить модуль следующим образом. Все строки модуля, которые идут после выполнения заменяются строками:

```
Результат = Запрос.Выполнить();
ОбластьСводной = Макет.ПолучитьОбласть("Сводная");
ТабДок.Вывести(ОбластьСводной);
ТабДок.ВстроенныеТаблицы.СводнаяТаблица1.Источн
икДанных = Результат;
```

Результат выполнения модуля — сводная таблица в отчете. Правда, пока незаполненная - без строк и колонок, это мы оставили пользователю. Он с помощью окна «Поля сводной таблицы», которое, при нашем запросе будет как на картинке справа.

Чтобы пользователь сразу видел, например, счета



кредита, как колонки таблицы, а счета дебета, как строки (а потом уже поменял бы, если захотел), нужно дописать еще несколько строк в модуль.

```
Результат = Запрос.Выполнить ();
```

```
ОбластьСводной = Макет.ПолучитьОбласть ("Сводная");
```

```
ТабДок.Вывести (ОбластьСводной);
```

```
СводнаяТаблица = ТабДок.ВстроенныеТаблицы.СводнаяТаблица1;
```

```
СводнаяТаблица.ИсточникДанных = Результат;
```

```
СводнаяТаблица.Обновление = Ложь;
```

```
СводнаяТаблица.Колонки.Добавить ("СчетКт");
```

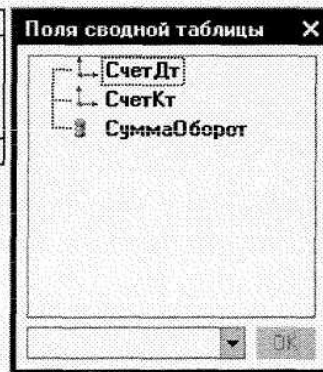
```
СводнаяТаблица.Строки.Добавить {"СчетДт"};
```

```
СводнаяТаблица.Данные.Добавить ("СуммаОборот");
```

```
СводнаяТаблица.Обновление = Истина;
```

В данном фрагменте программы мы, после вывода таблицы в отчет, определяем, что увидит пользователь в качестве строк, что в качестве колонок. До того, как проводить эти операции с таблицей имеет смысл установить свойство Обновление =Ложь. Это ускорит операции с

СчетДт	Товары	Поставщики	Касса	Капитал	Дебиторка	Итог
Товары		5 854				5 854
Касса				3 333	85 207	88 540
Капитал	60		1 030			1 090
Дебиторка			1 420	190		1 610
Итог	60	5 854	2 450	3 523	85 207	97 094



таблицей.

## Построитель отчета

Построитель отчета - объект, предназначенный для разработки сложных, настраиваемых пользователем отчетом. Причем, разработки с минимальными усилиями для разработчика.

В качестве примера использования построителя давайте разработаем отчет «Шахматка», который позволит просматривать обороты между счетами в таблице шахматной формы (строки - счета дебета, колонки - счета кредита). Т.е. решим задачу, ранее сделанную с помощью сводной таблицы другим способом.

Для этого создаем новый отчет. В список реквизитов отчета добавляем один реквизит «Построитель» типа «ПостроительОтчета».





Создаем диалог.

В диалоге формы располагаем три табличных поля, соответствующих типов (Свойство «данные» табличного поля):

Построитель.ИзмеренияСтроки

Построитель.ИзмеренияКолонки

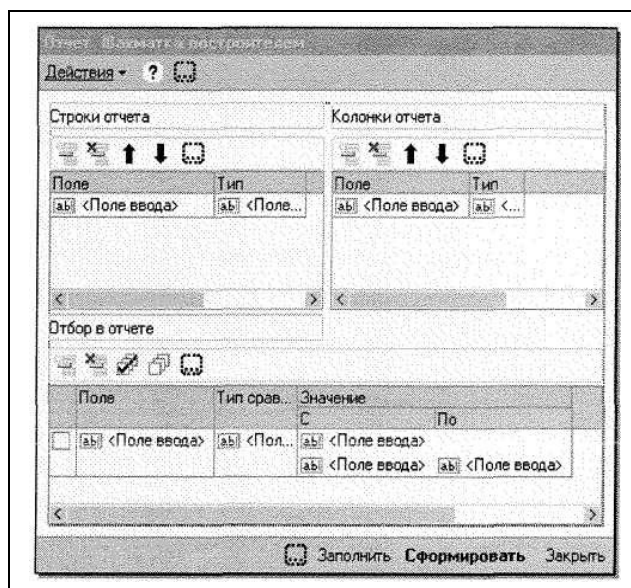
Построитель.Отбор

В этих полях, пользователь сможет, соответственно, указать строки отчета, колонки отчета, условия отбора.

Добавляем на нижнюю командную панель кнопку

«Заполнить», для которой описываем процедуру обработчик. Альтернативный вариант решения - выполнить заполнение настроек построителя в теле модуля или процедуре обработчике события «При открытии».

Вместе с процедурой обработчиком события нажатия кнопки Сформировать, получается модуль:



П  
о  
с  
с  
т  
р  
о  
и  
т  
е  
л  
ь  
.  
В  
ы  
в  
е  
с  
т  
и  
(  
);  
/  
/  
3

```
екст = "ВЫБРАТЬ //5 |
ОсновнойОборотыДтКт.СчетДт, |
ОсновнойОборотыДтКт.СчетКт,
        ОсновнойОборотыДтКт.СуммаОборот КАК СуммаОборот
|ИЗ
        РегистрБухгалтерии.Основной.ОборотыДтКт КАК
ОсновнойОборотыДтКт
|ГДЕ
        (ОсновнойОборотыДтКт.СчетДт.Забалансовый =
        ЛОЖЬ) И
        (ОсновнойОборотыДтКт.СчетДт.Забалансовый =
        ЛОЖЬ)

| {ИТОГИ ПО
ОсновнойОборотыДтКт.СчетДт,ОсновнойОборотыДтКт.СчетКт} //6

ИТОГИ СУММА (СуммаОборот)
        ПО ОВНЕ " ;
        Построитель.ЗаполнитьНастройки(); //
7 КонецПроцедуры
```

Текст запроса можно написать с помощью конструктора запросов, который вызывается в тексте модуля по пункту контекстного меню

К  
о  
н  
е  
ц  
П  
р  
о  
ц  
е  
д  
у  
р  
ы

Проце  
дура  
Основ  
ныеДе  
йстви  
яФорм  
ыЗапо  
лнить  
1 (Кно  
пка)  
//4

П  
о  
с  
т  
р  
о  
и  
т  
е  
л  
ь  
.  
Т

"Конструктор запросов" , вызываемого по правой кнопке мыши.

Кое-что в тексте придется поправить руками. Комментарии:

//1 процедура-обработчик события «Нажатие» кнопки Сформировать

//2 выполняем запрос с помощью построителя

//3 выводим результаты в табличный документ

//4 процедура кнопки Заполнить. Анализируя текст запроса, который передается в построитель, подготавливает данные для заполнения настроек построителя

//5 записываем в свойство построителя текст запроса. Запрос по таблице ОборотыДтКт без параметров. Параметры в виртуальную таблицу передавать не нужно, это возьмет на себя построитель - они будут доступны среди условий отбора.

//6 предложение в фигурных скобках {} дописано «руками». Это расширение языка запроса для построителя отчета. Оно будет игнорироваться объектом «Запрос», а объект ПостроительОтчета «поймет» конструкцию {ИТОГИ ПО...} как ресурсы для построителя отчета. Именно благодаря этой конструкции у пользователя будет возможность указать в полях строк и колонок отчета счета дебета и кредита. Если бы потребовалось вывести в список условий отбора, например, дополнительные параметры, то их тоже нужно было бы указать в фигурных скобках.

//7 после этого метода, пользователь получит возможность выбрать из возможных вариантов настроек отчета нужные в диалоге формы.

Проверяем. Открываем отчет в пользовательском режиме. Нажимаем «Заполнить». Настраиваем диалог как на картинке.

Теперь, по кнопке Сформировать получим отчет «Шахматка»





та, а только их часть, относящуюся к группировке одного уровня.

Чтобы предоставить пользователю такую возможность можно воспользоваться методами, которые рассматриваются ниже.

Группировку строк (колонок) в отчете можно устанавливать или самостоятельно (решать, где начинается и заканчивается группа), или поручить это программе. Для второго варианта потребуется внести следующие изменения в модуль.

До того, как начался первый уровень (группировка строк) отчета, начинаем автогруппировку строк.

```
ТабДок.НачатьАвтогруппировкуСтрок();
```

При выводе в результирующую таблицу строк, указываем к какому уровню они относятся. Если отчет формируется запросом (а обычно именно так и бывает), можно воспользоваться методом `Уровень()`, чтобы определить уровень строки.

```
ТабДок.Вывести(Область, Выборка.Уровень{});
```

И после окончания группировки

```
ТабДок.ЗакончитьАвтогруппировкуСтрок();
```

Гру  
ппи  
ров  
ка

Груп  
пиро  
вка  
позв  
оляе  
т  
пока  
зать  
поль  
зова  
тел  
ю не  
все  
стро  
ки  
(кол  
онки  
)  
отче

Если необходимо предоставить пользователю возможность управлять строками группировки (сворачивать / разворачивать все группировки одного уровня), то в модуле формы для таблицы необходимо установить свойство «Отображать заголовки».

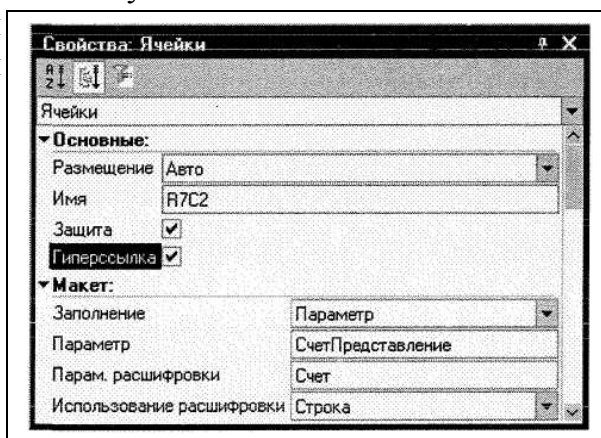
ТабДок.ОтображатьЗаголовки = Истина;

Этим свойством сможет управлять и пользователь через «Меню - Таблица - Вид - Отображать заголовки».

## Расшифровка

Расшифровка — возможность детализации показателей отчета. В отличие от Группировки, при которой мы детализируем отчет строчками того же типа (в нашем случае - тот же состав колонок и наполнение данными), при расшифровке мы можем детализировать отчет с помощью совершенно другого по структуре и наполнению отчета.

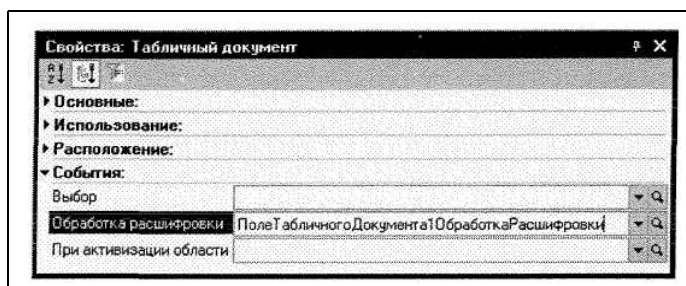
Основывается этот механизм на том, что каждая прямоугольная область табличного документа, кроме того значения,



которое показывается пользователю, может «за кадром» хранить еще одно - значение любого типа. Пользователь его не видит, но догадывается о его наличии (если для таблицы установлено свойство ТолькоПросмотр=Истина) - курсор приобретает над такой ячейкой вид перекрестия с лупой (или, если для ячейки установлено свойство гиперссылка, как на картинке справа, «пальчика или ручки»).

Если больше ничего не делать, т.е. ограничиться тем, что записать в ячейку таблицы параметр расшифровки (Счет), установить свойство ТолькоПросмотр для табличного документа, то, по двойному щелчку мыши (или, если это гиперссылка - одинарный щелчок) программа обработает значение «стандартно» - если это объект, откроет форму этого объекта. Если установить свойство «Использование расшифровки» в значение «Строка», то указанное действие будет происходить по щелчку на любой ячейке строки, расположенной от настраиваемой вправо.

Если же мы хотим, чтобы было выполнено какое-то действие, это действие надо описать. Отвечает за это событие «Обработка





расшифровки» элемента формы типа табличный документ. Существует это событие ТОЛЬКО для табличного документа, расположенного в форме.

Чтобы описать его, необходимо щелкнуть мышью по полю табличного документа (выделится ячейка табличного документа), нажать Esc (выделится весь табличный документ), и создать процедуру-обработчик для события Обработка расшифровки.

Если среди свойств отсутствуют события, видимо причина в кнопке



, которая позволяет спрятать часть свойств и которая должна быть «отжата».

## Расшифровка отчета Оборотка

Оборотку мы разрабатывали с помощью конструктора выходных форм, и параметр расшифровки он уже прописал для ячейки второго столбца области «Счет». Нам осталось установить свойство Использование расшифровки в значение «Строка», и для ячеек этой строки установить флажок «Гиперссылка».

Мы будем расшифровывать Оборотку с помощью отчета Анализ счета. Оборотка показывает пользователю обороты по счету в целом, а АнализСчета - в разрезе корреспондирующих счетов. Там же в отчете «Оборотка», в свойствах табличного поля назначаем обработчик события ОбработкаРасшифровки.

Сам механизм расшифровки, таким образом, реализуется следующим образом:

1. Записать в параметр расшифровки макета отчета, который будем детализировать (Оборотка) значение, по которому и будем строить подробный отчет (в нашем случае - счет)
2. Описать обработчик события Обработка расшифровки

Чем отличается событие табличного документа (как элемента формы) Выбор от ОбработкиРасшифровки? Выбор «срабатывает» при нажатии на любой ячейке, даже если там не указан параметр расшифровки.

## Первый вариант решения

В форме отчета «Анализ счета» располагается процедура СформироватьНажатие(Элемент), которая является обработчиком события Нажатие и запускает формирование отчета. Делаем ее «экспортной», чтобы иметь возможность обращаться к ней из модуля.

Описываем в модуле формы отчета Оборотка процедуру-обработчик события ОбработкаРасшифровки

Определяем в свойствах табличного документа, расположенного в диалоге формы отчета событие - обработка расшифровки.

Процедура ПолеТабличногоДокументаОбработкаРасшифровки (Элемент,  
Расшифровка, СтандартнаяОбработка) //1 СтандартнаяОбработка =  
Ложь; //2

ФормаПодробного = Отчеты.АнализСчета.ПолучитьФорму(); //3  
ФормаПодробного.Дата1 = Дата1; //4 ФормаПодробного.Дата2  
= Дата2; //5 ФормаПодробного.Счет = Расшифровка; //6  
ФормаПодробного.Фирма = Фирма; 1/1

ФормаПодробного.Открыть(); //8  
ФормаПодробного.ЭлементыФормы.СуммаВаланса.Значение =  
ЭлементыФормы.СуммаВаланса.Значение; //9  
ФормаПодробного.СформироватьНажатие (Неопределено); //10

КонецПроцедуры

**Комментарии к модулю:**

//1 процедура обработчик события расшифровки

//2 не открывать диалог счета

//3 получаем форму отчета, который будет детализировать Оборотку,  
т.е. Анализа

//4, 5,7 - устанавливаем значения реквизитов формы

//6 в качестве счета - передаем полученное на параметр Расшифровка  
значение счета

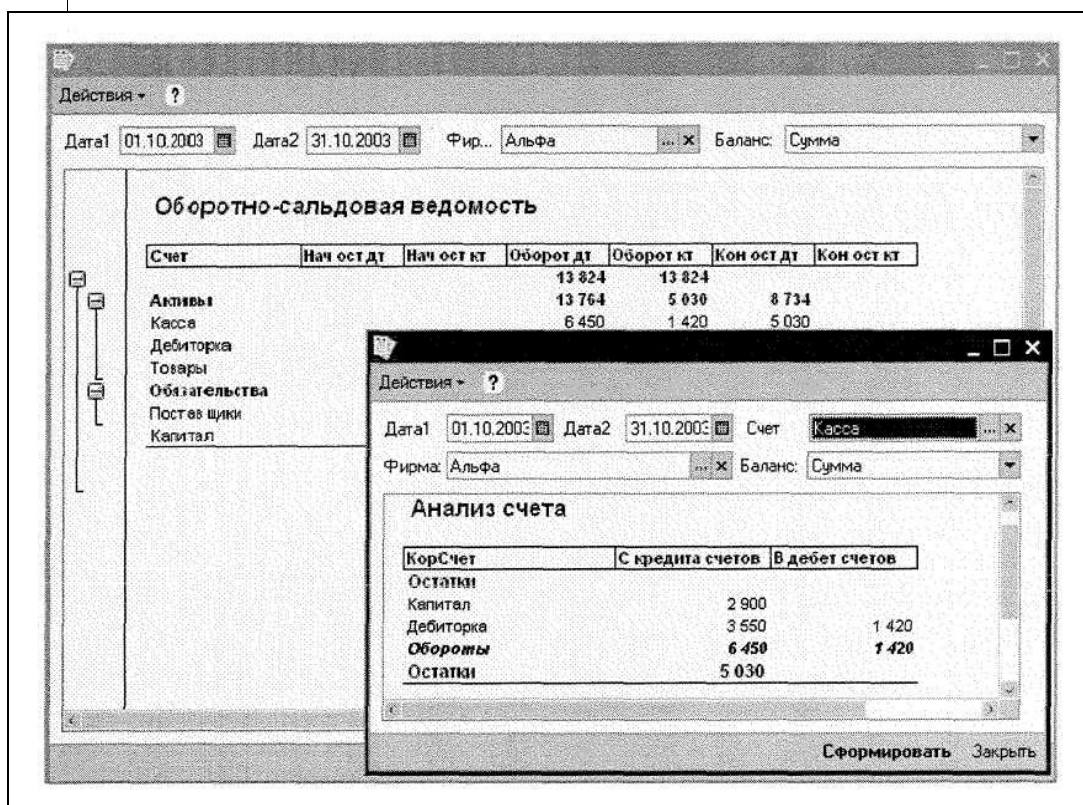
//8 Т.к. табличный документ расположен в диалоге формы, открываем  
форму отчета «Анализ счета»

//9 указываем в качестве выбранного значения списка «сумма баланса»  
то же значение, что выбрано сейчас в диалоге формы Оборотки

//10 запускаем процедуру СформироватьНажатие



## Второй вариант решения



90

Если в отчете Анализ счета табличный документ не был расположен в диалоге формы, можно было не возиться с заполнением реквизитов формы, заполнением списка поля выбора и проч. Ведь процедура формирования отчета располагается в модуле объекта и объявлена как экспортная. Можно передать параметры сразу в нее. Анализ счета внесения изменений не требует. Модуль обработки расшифровки Оборотки:

Процедура

```
ПолеТабличногоДокумента1ОбработкаРасшифровки (Элемент,
Расшифровка, СтандартнаяОбработка)
    СтандартнаяОбработка = Ложь;
    ТабДок = Новый ТабличныйДокумент; //1
    ПодробныйОтчет = Отчеты.АнализСчета.Создать(); //2
    ПодробныйОтчет.Отчет(ТабДок, Дата1, Дата2,
Расшифровка, Фирма, СуммаБаланса); //3
    ТабДок.ТолькоПросмотр = Истина; ///4
    ТабДок.ОтображатьСетку = Ложь;
    ТабДок.ОтображатьЗаголовки = Ложь;
    ТабДок.Показать();
//5 КонецПроцедуры
```

Эта процедура откроет таблицу с Анализом счета в отдельном окне. Комментарии к модулю:

//1 создаем новый табличный документ, в который выведем пользователю подробный отчет.

//2 получаем в переменную подробный отчет - объект ОтчетОбъект, именно его и возвращает метод ОтчетМенеджера.Создать()

//3 используя этот объект запускаем экспортную процедуру, расположенную в модуле объекта, передавая ей в качестве параметров требуемые значения

//4 устанавливаем свойства табличного документа //5

показываем его пользователю

Готово. Однако, при таком решении - детализировать отчет Анализ с помощью другого отчета уже не удастся - ведь у него нет формы. Весь отчет представлен лишь табличным документом.

## Комментарии

Если есть необходимость различные ячейки одного отчета расшифровывать различными отчетами, то определить, какую ячейку выбрал пользователь, может помочь свойство табличного документа ТекущаяОбласть . Через

ЭлементыФормы.ИмяТабличногоДокумента.ТекущаяОбласть.Имя — можно получить имя активной области, которое задано в формате **R1C1** - номер строки, номер колонки. А далее по номеру колонки определить -что именно нужно детализировать (дебетовый оборот или кредитовый). Или получить то же самое иначе: Номер колонки = ЭлементыФормы.<Имятаблицы>.ТекущаяОбласть.Лево/Право

## Практикум №13

*Продолжить цепочку расшифровки и детализировать отчет Анализ счета с помощью отчетов Карточка счета и Отчет по проводкам (по таблице ОборотыДтКт).*



## Самостоятельная работа

Мы с Вами прошли основные механизмы, с помощью которых можно реализовать бухгалтерский учет в «1С:Предприятии 8.0». Для закрепления материала Вам предоставляется следующая задача. В предыдущих примерах мы разобрали схему учета собственного товара, но кроме него наша фирма торгует еще и товаром, взятым на реализацию. Право собственности на этот товар остается за комитентом (поставщиком), у нас он числится только на ответственном хранении. При продаже комиссионного товара дополнительных услуг мы не оказываем. Дата продажи совпадает с датой отгрузки.

Задача: Организовать учет торговли комиссионным товаром. **Используемые счета бухгалтерского учета.**

Счет 10 «Товары на комиссии» (новый счет первого уровня)

Товар в ценах покупки должен учитываться на забалансовом счете 10 «Товары на комиссии» (право собственности на товар принадлежит поставщику, значит, он не является активом нашего предприятия). На счете ведется количественный и аналитический учет. Аналитический учет ведется в разрезе товаров, принятых на комиссию и комитентов (организации - поставщиков), от которых этот товар получен.

При списании товара себестоимость рассчитывается как средняя на момент проведения документа отдельно по каждому комитенту, т.е. мы можем брать на реализацию один и тот же товар по разным ценам от разных комитентов и при списании должны будем указать, товар какого комитента мы сейчас списываем. Причем, в одной расходной накладной мы можем продавать товар разных комитентов.

Счет 02.3 «Комитенты» (новый счет в группе обязательств)

Расчеты с комитентом (поставщиком) ведутся на отдельном балансовом счете 02.3 «Расчеты с комитентами». На счете ведется аналитический учет в разрезе субконто «Организации»

Счет 01.2 «Дебиторка»

Учет расчетов с покупателями комиссионного товара ведется на счете 01.2 «Дебиторка» в разрезе организаций.

### Автоматизировать операции с помощью документов

Необходимо автоматизировать с помощью документов операции получения товара и его продажи.

### Разработать отчеты:

Движение комиссионного товара: оборотная ведомость, по строкам которой товар, по столбцам: Начальный остаток, Приход, Расход, Конечным остаток. Строки отчета показывают суммовые остатки и

обороты. Пользователь может выбрать следующие режимы: По одной фирме или по всем, Период отчета, Товар (или группу товаров),

Комитент (или группу комитентов). Макет отчета:

Товар	Ост. на начало	Приход	Расход	Ост. на конец
Всего:				

Прибыль, полученную от продажи комиссионного товара, по каждому комитенту. Пользователь может выбрать: Период отчета, фирму. Отчет должен показывать обороты в кредит счета Капитал с дебета счета Комитенты.

Комитет	(ЛГмма прибыли)
Наш главный комитент	1,500=00
Тоже комитент	30=00
Всего:	1,530=00

Схема проводок «самолетиками» по комиссионной торговле.

Взяли на реализацию товар по 15,000=00, продали покупателю за 20,000=00

Разделы (горизонтальные пунктирные линии): 1 - поступление товара на комиссию, 2 - реализация, 3 - оплата комитенту (ручная операция), 4 - оплата от покупателя (ручная операция)

Товары на комиссии	Капитал	Расчеты с комитентами	Дебиторка	Касса
15,000=00 Поступлен				
15,000=00 Списание		20,000=00 Задолженность		
	5,000=00 Ком.вознагр.			
	5,000=00			
	Касса			
	15,000=00 Опл. комитенту			
			20,000=00 Опл. покупат.	
	5,000=00			

При решении задачи на релизе 8.0.3.15 и более ранних, следует отказаться от формирования проводок по списанию товара с забалансового счета из-за особенностей работы платформы.



## Заключение.

Курс, который был Вами (и надеюсь — успешно!) пройден, является второй после курса «Введение в конфигурирование в системе «1С:Предприятие 8.0». Основные объекты» и далеко не последней ступенью в изучении конфигурирования и программирования в системе «1С:Предприятие 8.0». Что можно посоветовать начинающему внедренцу?

Во-первых, пройти этот курс еще раз самостоятельно, но уже не в группе, где Вы должны были «подстраиваться» под уровень Ваших коллег (для кого-то слишком быстрый, для кого-то наоборот слишком медленный), а самостоятельно. При этом очень желательно еще раз прорешать все практикумы или хотя бы те из них, которые не были Вами успешно выполнены при обучении в группе.

Во-вторых, изучать типовые конфигурации. Типовые конфигурации разрабатываются ведущими специалистами, поэтому вполне можно принимать решения в них реализованные как теоремы программирования в среде «1С:Предприятие».

В-третьих, если Вы хотите, что бы Ваши знания росли не только «вглубь», но и «вширь» мы рекомендуем Вам рассмотреть возможность продолжения обучения на наших курсах. Кроме «Задач бухгалтерского учета» существуют курсы для специалистов, занимающихся автоматизацией торгового и складского учета, учетом персонала и расчета заработной платы и др. (следите за обновлением программ и курсов на нашем сайте).

И самое главное - любые знания, полученные Вами на наших курсах и не закреплённые на практике, будут Вами потеряны... Чтобы этого не произошло - надо программировать, в чем мы и желаем Вам успехов!

## Успехов в работе!





Журнал-главная за XX месяц 20XX г. по предприятию Иванова И.И.

№	Операция	Наличные		Офис		Мебель		Обязательства к получению		Капитал		Обязательства к выдаче	
		Дебет	Кредит	Дебет	Кредит	Дебет	Кредит	Дебет	Кредит	Дебет	Кредит	Дебет	Кредит
Сальдо на начало:													
1	Инвестиция					,	-----.						
2	Купил офис					-----	-----						
	Мебель в долг												
	Оказал услуги												
	Услуги в долг		-----										
6	Зарплата сотруднику												
7	Отдал долг												
8	Сходил в кино												
9	Получил долг												
10	Продал офис							-----					
11	Изъял деньги												
Обороты за период:													
Сальдо на конец:													

Оборотный баланс за XX месяц 20XX г. по предприятию Иванова И.И.

Счет	Сальдо на начало		Обороты за период		Сальдо на конец	
	Дебет	Кредит	Дебет	Кредит	Дебет	Кредит
Наличные						
Офис						
Мебель						
Обязательства к получению						
Капитал						
Обязательства к выдаче						
Итого:						



## Дополнение к методике

Конфигурирование в системе «1С:Предприятие 8.0».  
Решение задач бухгалтерского учета.

### Доступ к субконто движения регистра бухгалтерии

Кроме синтаксиса, который используется конструктором при формировании движений документами в регистр бухгалтерии, появился альтернативный более простой и удобный вариант заполнения значений субконто по их именам.

Вариант, использовавшийся ранее:

Движение.СубконтоДт [ПланыВидовХарактеристик.ВидыСубконто.Номенклатура] =  
ТекСтрокаТовары.Номенклатура;

Новый вариант:

Движение.СубконтоДт.Номенклатура = ТекСтрокаТовары.Номенклатура;

Подобный вариант использовался в системе «1С:Предприятие 7.7», и удобен в первую очередь своей краткостью. Осталось дожидаться появления варианта заполнения субконто по его номеру (аналога Субконто(НомерСубконтоНаСчете.ЗначениеСубконто) версии 7.7)...

### Активность записей регистра

Кроме описанной в методике возможности управлять активностью записей регистра программно, появилась возможность делать это интерактивно в пользовательском режиме без написания дополнительного кода.

А...	Ном...	Счет Дт	СубконтоДт1	Количество Дт	Счет Кт	СубконтоКт1	Количество Кт	Сумма
			СубконтоДт2	Валюта... Валютная...		СубконтоКт2	Валют... Валютная...	Содержание
<input checked="" type="checkbox"/>	1	Касса		Доллар	Дебиторка	Купим все		3 050,00
<input checked="" type="checkbox"/>	2	Касса		руб.	Дебиторка	Купим всегда		2 000,00
<input checked="" type="checkbox"/>	3	Касса		Евро	Дебиторка	Купим все		30 000,00
<input checked="" type="checkbox"/>	4	Касса		Доллар	Дебиторка	Поставка		6 100,00

Новую возможность можно проверять, например, в документе «Операция». Достаточно с помощью помощника размещения данных добавить в табличное поле, отображающее набор записей регистра бухгалтерии, новую колонку «Активность» и у пользователя появится возможность ей управлять. Особенность, которую надо учитывать: управлять активностью каждой записи можно... но, в одной операции (записях по одному регистратору) не могут одновременно присутствовать и активные и неактивные записи - платформа это контролирует. Программ не разрешит записать «Операцию» где у части записей флажки активности сняты, у части - установлены.

## Методы объекта «Регистр бухгалтерии» для доступа к остаткам и оборотам

Если системе «1С:Предприятие 7.7» предоставляла четыре возможности работы с итогами:

- Основной расчет
- Временный расчет
- Бухгалтерский запрос
- Объект «Запрос» через проводки «Операции»

И первые три из них являлись возможностями объекта «Бухгалтерские Итоги», то до последнего релиза, система «1С:Предприятие 8.0» предоставляла всего одно (хотя и очень мощное и удобное!) средство доступа к остаткам и оборотам - использование запроса и табличной модели данных. Сильно упрощали ситуацию виртуальные таблицы регистра бухгалтерии, работа с которыми во многом напоминала работу с бухгалтерским запросом версии 7.7. Однако, не хватало простого синтаксиса и краткого написания методов основного расчета, знакомым многим внедренцам начиная с самых ранних версий «1С:Бухгалтерии». Теперь появились их аналоги. У объекта РегистрБухгалтерииМенеджер появилось три новых метода

### Обороты

РегистрБухгалтерииМенеджер.<Имя регистра бухгалтерии> (AccountingRegisterManager.<Имя регистра бухгалтерии>) Обороты (Turnovers)

Синтаксис:

Обороты(<Начало периода>, <Конец периода>, <Виды субконто>, <Виды кор. субконто>, <Отбор>, <Измерения>, <Ресурсы>)

Параметры:

<Начало периода> (необязательный)

Тип: Дата. МоментВремени. Граница. Момент времени, начиная с которого необходимо получить обороты. Если параметр не указан или установлен в Неопределено, то обороты будут получены начиная с момента ведения информационной базы. Если в качестве параметра передана Дата или МоментВремени, то обороты за указанную дату (момент времени) будут включены в результирующую таблицу значений.

Значение по умолчанию: Неопределено

<Конец периода> (необязательный)

Тип: Дата, МоментВремени, Граница. Момент времени, определяющий конец периода за который необходимо получить обороты. Если параметр не указан или установлен в Неопределено, то обороты будут получены до самого позднего движения регистра. Если в качестве параметра передана Дата или Момент Времени, то обороты за указанную дату (момент времени) будут включены в результирующую таблицу значений.

Значение по умолчанию: Неопределено

<Виды субконто> (необязательный)

Тип: ПланВидовХарактеристикСсылкаxИмя плана видов характеристика. Передается ссылка или массив ссылок на виды субконто. Виды субконто определяют, какие значения субконто будут доступны по именам "Субконто1", "Субконто2" и т.д. Кроме того, обороты будут посчитаны только по счетам, имеющим заданные виды субконто. Если виды субконто не заданы, то обращение к значениям субконто осуществляется по порядку (индексу), как они определены на соответствующем счете.

Значение по умолчанию: Неопределено

<Виды кор. субконто> (необязательный)

Тип: ПланВидовХарактеристикСсылкаxИмя плана видов характеристик. Передается ссылка или массив ссылок на виды субконто. Виды кор. субконто определяют, какие значения корреспондирующих субконто будут доступны по именам "КорСубконто1", "КорСубконто2" и т.д. Кроме того, обороты будут посчитаны только по кор. счетам, имеющим заданные виды кор. субконто. Если виды кор. субконто не заданы, то обращение к значениям кор. субконто осуществляется по порядку (индексу), как они определены на соответствующем счете.

Значение по умолчанию: Неопределено

<Отбор> (необязательный)

Тип: Структура. Структура, содержащая набор значений измерений регистра, по которым надо отбирать итоги.

Допустимые значения ключа: Счет, Субконто, <Имя измерения>, КорСчет, КорСубконто, <Имя измерения>Кор (только для не балансовых измерений).

Значение по умолчанию: Неопределено

<Измерения> (необязательный)

Тип: Строка. Список измерений, для которых надо получить обороты. Строка, содержащая имена измерений, разделенных запятыми. Допустимые имена измерений: Счет, Субконто, <Имя измерения>, КорСчет, КорСубконто, <Имя измерения>Кор.

Если параметр не указан или указана пустая строка, то обороты будут сформированы по всем измерениям.

Значение по умолчанию: пустая строка

<Ресурсы> (необязательный)

Тип: Строка. Список ресурсов, для которых надо получить обороты. Строка, содержащая имена ресурсов, разделенных запятыми.

Если параметр не указан или указана пустая строка, то обороты будут сформированы по всем ресурсам.

Значение по умолчанию: пустая строка

Возвращаемое значение:

Тип: ТаблицаЗначений.

Описание:

Получение оборотов по регистру бухгалтерии на заданный момент. Обороты получаются в разрезе заданных измерений и по заданным ресурсам. В измерения входят счет, субконто, измерения, кор. счет, кор. субконто, кор. измерения.

## ОборотыДтКт

РегистрБухгалтерииМенеджер<Имя регистра бухгалтерии> (AccountingRegisterManager.<Имя регистра бухгалтерии>)  
ОборотыДтКт (DrCrTurnovers)

Синтаксис:

ОборотыДтКт(<Начало периода>, <Конец периода>, <Виды субконто Дт>, <Виды субконто Кт>, <Отбор>, <Измерения>, <Ресурсы>)

Параметры:

<Начало периода> (необязательный)

Тип: Дата, МоментВремени, Граница. Момент времени, начиная с которого необходимо получить обороты. Если параметр не указан или установлен в Неопределено, то обороты будут получены начиная с момента ведения информационной базы. Если в качестве параметра передана Дата или МоментВремени, то обороты за указанную дату (момент времени) будут включены в результирующую таблицу значений.

Значение по умолчанию: Неопределено

<Конец периода> (необязательный)

Тип: Дата, МоментВремени, Граница. Момент времени, определяющий конец периода за который необходимо получить обороты. Если параметр не указан или установлен в Неопределено, то обороты будут получены до самого позднего движения регистра. Если в качестве параметра передана Дата или Момент Времени, то обороты за указанную дату (момент времени) будут включены в результирующую таблицу значений.

Значение по умолчанию: Неопределено

<Виды субконто Дт> (необязательный)

Тип: ПланВидовХарактеристикСсылка.<Имя плана видов характеристика> Передается ссылка или массив ссылок на виды субконто. Параметр определяет, какие значения субконтоДт будут доступны по именам "СубконтоДт1", "СубконтоДт2" и т.д. Кроме того, обороты будут посчитаны только по счетамДт, имеющим заданные виды субконтоДт. Если параметр не задан, то обращение к значениям субконтоДт осуществляется по порядку (индексу), как они определены на соответствующем счете.

Значение по умолчанию: Неопределено

<Виды субконто Кт> (необязательный)

Тип: ПланВидовХарактеристикСсылка.<Имя плана видов характеристика> Передается ссылка или массив ссылок на виды субконто. Виды субконто определяют, какие значения субконто Кт будут доступны по именам "СубконтоКт1", "СубконтоКт2" и т.д. Кроме того, обороты будут посчитаны только по счетам Кт, имеющим заданные виды субконтоКт. Если виды субконтоКт не заданы, то обращение к значениям субконтоКт осуществляется по порядку, как они определены на соответствующем счете.

Значение по умолчанию: Неопределено

<Отбор> (необязательный)

Тип: Структура. Структура, содержащая набор значений измерений регистра, по которым надо отбирать итоги.

Допустимые значения ключа: СчетДт, СубконтоДт, СчетКт, СубконтоКт, <Имя измерения> или <Имя измерения>Дт и <Имя измерения>Кт.

Значение по умолчанию: Неопределено

<Измерения> (необязательный)

Тип: Строка. Список измерений, для которых надо получить обороты. Строка, содержащая имена измерений, разделенных запятыми. Допустимые имена измерений: СчетДт, СубконтоДт, СчетКт, СубконтоКт, <Имя измерения> или <Имя измерения>Дт и <Имя измерения>Кт.

Если параметр не указан или указана пустая строка, то обороты будут сформированы по всем измерениям.

Значение по умолчанию: пустая строка

<Ресурсы> (необязательный)

Тип: Строка. Список ресурсов, для которых надо получить обороты. Строка, содержащая имена ресурсов, разделенных запятыми.

Если параметр не указан или указана пустая строка, то обороты будут сформированы по всем ресурсам.

Значение по умолчанию: пустая строка

Возвращаемое значение:

Тип: ТаблицаЗначений.

Описание:

Получение оборотов по регистру бухгалтерии на заданный момент. Обороты получаются в разрезе заданных измерений и по заданным ресурсам. В измерения входят счетДт, субконтоДт, счетКт, субконтоКт, измеренияКт, измерения (для балансовых) и измеренияДт, измеренияКт (для не балансовых).

## Остатки

РегистрБухгалтерииМенеджер<Имя регистра бухгалтерии> (AccountingRegisterManager.<Имя регистра бухгалтерии>)



Остатки (Balance)

Синтаксис:

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

Параметры:

<МоментВремени> (необязательный)

Тип: Дата, МоментВремени, Гранина.

<Виды субконто> (необязательный)

Тип: ПланВидовХарактеристикСсылка.<Имя плана видов характеристик>. Массив. Передается ссылка или массив ссылок на виды субконто. Виды субконто определяют, какие значения субконто будут доступны по именам "СубконтоГ", "Субконто2" и т.д. Кроме того, остатки будут посчитаны только по счетам, имеющим заданные виды субконто. Если виды субконто не заданы, то обращение к значениям субконто осуществляется по порядку (индексу), как они определены на соответствующем счете.

Значение по умолчанию: Неопределено

<Отбор> (необязательный)

Тип: Структура. Структура, содержащая набор значений измерений регистра, по которым надо отбирать итоги.

Допустимые значения ключа: Счет, Субконто, <Имя измерения>».

Значение по умолчанию: Неопределено

<Измерения> (необязательный)

Тип: Строка. Список измерений, для которых надо получить остатки. Строка, содержащая имена измерений, разделенные запятыми. Допустимые имена измерений: Счет, Субконто, <Имя измерения>

Если параметр не указан или указана пустая строка, то остатки будут сформированы по всем измерениям.

Значение по умолчанию: пустая строка

<Ресурсы> (необязательный)

Тип: Строка. Список ресурсов, для которых надо получить остатки. Строка, содержащая имена ресурсов, разделенные запятыми.

Если параметр не указан или указана пустая строка, то остатки будут сформированы по всем ресурсам.

Значение по умолчанию: пустая строка

Возвращаемое значение:

Тип: ТаблицаЗначений.

Описание:

Получение остатков по регистру бухгалтерии на заданный момент. Остатки получаются в разрезе заданных измерений и по заданным ресурсам. В измерения входят счет и субконто, зависящие от счета. Если параметр <Виды субконто> не задан, то обращение к субконто осуществляется по индексу в том порядке, как виды субконто определены на соответствующем счете. Если параметр <Виды субконто> задан, то остатки будут считаться только по счетам, у которых определены заданные виды субконто, обращение к значениям субконто осуществляется в порядке, заданном параметром <Виды субконто>.

## Комментарии:

Последний из них (Остатки) заставить работать так и не удалось... Однако, все остальные методы работают, и, видимо, работают «запросом», как и аналогичные методы регистра накопления. Т.е. за методом «прячется» запрос, таким образом, производительность метода объекта и запроса к таблице должна быть примерно одинакова. Проверить методы можно в отчете, в диалоге которого расположено табличное поле типа «Таблица значений» с именем Таб.

Таб = РегистрыБухгалтерии.Основной.ОборотыДтКтО;  
ЭлементыФормы.Таб.СоздатьКолонки{};

Данный код покажет обороты между счетами. То же самое можно получить, выгрузив в таблицу значений результат запроса к виртуальной таблице регистра