

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
**Федеральное государственное автономное образовательное учреждение
высшего образования**
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**Методические указания
по выполнению лабораторных работ по дисциплине**

«Программирование и конфигурирование в 1С»

для студентов направления 09.03.01 Информатика и вычислительная техника

СОДЕРЖАНИЕ

- Лабораторная работа №1. Основная конфигурация и конфигурация базы данных
- Лабораторная работа №2. Создание справочника «Клиенты»
- Лабораторная работа №3. Создание справочника «Сотрудники»
- Лабораторная работа №4. Создание справочника «Номенклатура»
- Лабораторная работа №5. Создание справочника «Склады»
- Лабораторная работа №6. Объект конфигурации «Документ»
- Лабораторная работа №7. Создание документа «Приходная Накладная»
- Лабораторная работа №8. Создание формы документа
- Лабораторная работа №9. Создание процедуры обработки события в модуле формы
- Лабораторная работа №10. Создание документа «Оказание услуги»
- Лабораторная работа №11. Назначение регистра накопления
- Лабораторная работа №12. Создание регистра накопления «Остатки Материалов»
- Лабораторная работа №13. Создание движений документа «Приходная Накладная»
- Лабораторная работа №14. Создание движений документа «Оказание Услуги»
- Лабораторная работа №15. Создание отчетов
- Лабораторная работа №16. Объект конфигурации Макет
- Лабораторная работа №17. Назначение периодического регистра сведений
- Лабораторная работа №18. Объект конфигурации Перечисление
- Лабораторная работа №19. Проведение документа по нескольким регистрам
- Лабораторная работа №20. Оборотный регистр накопления
- Лабораторная работа №21. Создание сложных отчетов

Лабораторная работа №1. Основная конфигурация и конфигурация базы данных

- Цель:** 1. Изучить механизм внесения изменений в конфигурацию.
2. Научиться использовать инструмент разработчика – *палитру свойств*.

С точки зрения пользователя программа 1С состоит из платформы и конфигурации. В каждом конкретном случае используется одна из множества возможных конфигураций. На самом деле это не совсем так.

Почему *не* так? Потому что в каждой информационной базе существуют как минимум две конфигурации.

Почему не *совсем* так? Потому что пользователь действительно работает всегда только с одной конфигурацией. Вторая конфигурация предназначена для разработчика или человека, который должен вносить изменения в конфигурацию (например, администратора базы данных). Для пользователя она «не видна».

Конфигурация, предназначенная для разработчика, называется *Основная конфигурация* (или просто *Конфигурация* – та, которую мы редактировали в Конфигураторе). Конфигурация, с которой работают пользователи, называется *Конфигурация базы данных*.

Основную конфигурацию можно редактировать. Конфигурацию базы данных редактировать нельзя, можно только произвести обновление конфигурации базы данных на основе основной конфигурации.

Такое внутреннее устройство позволяет вносить изменения в конфигурацию, не прерывая работы пользователей (поскольку изменения вносятся в основную конфигурацию). Затем, когда разработчик будет уверен в том, что все изменения, которые он внес, верны, можно будет быстро произвести обновление конфигурации базы данных, используя основную конфигурацию. Но для этого придется завершить работу всех пользователей.

Разработчик всегда может сравнить основную конфигурацию и конфигурацию базы данных, может вернуться к исходному состоянию основной конфигурации, используя конфигурацию базы данных (если, например, совсем запутался в своих изменениях).

Таким образом, взаимодействие двух конфигураций можно представить, как показано на рис. 1.

Когда разработчик работает с основной конфигурацией, система всегда подсказывает ему, отличается ли его вариант основной конфигурации от того, который сохранен, и отличается ли сохраненный вариант основной конфигурации от конфигурации базы данных.



Рис. 1. Взаимодействие двух конфигураций

Если разработчик редактирует основную конфигурацию и редактируемый вариант основной конфигурации отличается от того, который сохранен, в заголовке окна дерева конфигурации появляется признак модифицированности конфигурации (*) – рис. 2:

Конфигурация *

Рис. 2. Заголовок окна дерева конфигурации

Если сохраненный вариант основной конфигурации отличается от конфигурации базы данных, в заголовке окна дерева конфигурации появляется знак отличия конфигураций (<!>) – рис. 3:

Конфигурация <!>

Рис. 3. Заголовок окна дерева конфигурации

Для сохранения основной конфигурации следует воспользоваться командой *Конфигурация > Сохранить конфигурацию*, а для обновления конфигурации базы данных необходимо выполнить команду *Конфигурация > Обновить конфигурацию базы данных*. При выполнении команды *Отладка > Начать отладку* система сама сначала сохраняет основную конфигурацию, а затем производит ее сравнение с конфигурацией базы данных. В случае если конфигурации отличаются, выдается запрос на обновление конфигурации базы данных, который вы видели в предыдущих примерах.

При выполнении команды *Отладка > Продолжить* система, после описанных выше действий, предлагает еще и перезапустить приложение, чтобы прекратить текущую отладочную сессию.

Таким образом, система старается облегчить жизнь разработчика и автоматизировать часто выполняемые операции.

Важным фактом является то, что именно в момент обновления конфигурации базы данных система создает (модифицирует) в базе данных те структуры хранения данных, которые мы описали в виде объектов конфигурации.

Таким образом, обычные элементы справочника пользователь добавляет в ту структуру базы данных, которую создала система на основе объекта конфигурации Справочник, а предопределенные элементы этого справочника система добавляет в эту

структуру сама, на основе все того же описания этой структуры, которым является объект конфигурации Справочник.

Палитра свойств

Палитра свойств – это специальное служебное окно, которое позволяет редактировать все существующие свойства объекта конфигурации. Поскольку разные объекты конфигурации имеют самые разные свойства, содержимое этого окна будет меняться в зависимости от того, какой объект является текущим (на каком объекте конфигурации установлен курсор).

При некоторых действиях разработчика (например, создание табличной части) палитра свойств открывается автоматически. Но разработчик всегда может открыть палитру свойств объекта конфигурации самостоятельно, воспользовавшись пунктом Свойства контекстного меню правой кнопки мыши.

В этом случае палитра свойств откроется и будет закреплена на рабочей области конфигуратора. Однако есть удобная возможность «открепить» палитру свойств (используя символ кнопки в заголовке окна палитры свойств – рис. 4):

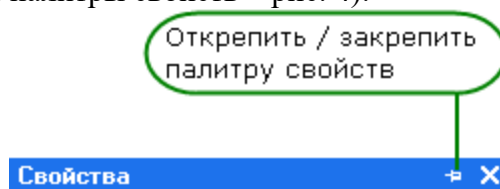


Рис. 4. «Открепим» палитру свойств...

В этом состоянии, при наведении курсора мыши на любое другое окно, палитра свойств будет сворачиваться на дополнительную панель в правой части экрана (по умолчанию) (рис. 5):



Рис. 5. Кнопка на дополнительной панели

А при наведении курсора мыши на символ свернутой палитры свойств она будет открываться.

Подобным поведением (возможностью быть прикрепленным, прячущимся и т.д.) обладает не только окно палитры свойств, но и другие окна конфигуратора (например, окно дерева конфигурации).

Объект конфигурации Справочник

Объект конфигурации Справочник является прикладным и предназначен для работы со списками данных. Объект конфигурации Справочник используется для того, чтобы на его основе платформа создала в базе данных информационную структуру, в которой будет храниться, например, список сотрудников, перечень товаров, список клиентов или поставщиков.

Справочник состоит из *элементов*. Характерной особенностью объекта конфигурации Справочник является то, что пользователь в процессе работы может самостоятельно добавлять новые элементы в справочник: например, добавить в справочник новых сотрудников, создать новый товар или внести нового клиента.

Каждый элемент справочника, как правило, содержит некоторую дополнительную информацию, которая подробнее описывает этот элемент. Например, все элементы справочника Товары могут содержать дополнительную информацию о производителе, сроке годности и др. Набор такой информации является одинаковым для всех элементов справочника, и для описания такого набора используются *реквизиты* объекта конфигурации Справочник, которые также, в свою очередь, являются объектами конфигурации. Поскольку эти объекты логически связаны с объектом Справочника, они называются *подчиненными*. Большинство реквизитов объекта конфигурации Справочник разработчик создает самостоятельно, однако у каждого объекта конфигурации Справочник существуют два поля «по умолчанию»: Код и Наименование.

Кроме этого, каждый элемент справочника может содержать некоторый набор информации, которая одинакова по своей структуре, но различна по количеству, предназначена для разных элементов справочника. Так, например, каждый элемент справочника Сотрудники может содержать информацию о составе семьи сотрудника. Для одного сотрудника это будет только супруга, а у другого семья может состоять из супруги, сына и дочери. Для описания подобной информации могут быть использованы *табличные части* объекта конфигурации Справочник, являющиеся подчиненными объектами конфигурации.

Для удобства использования элементы справочника могут быть сгруппированы пользователем по какому-либо принципу. Например, в справочнике Бытовая техника могут быть созданы следующие *группы*: Холодильники, Телевизоры, Стиральные машины и т.д. Возможность создания таких групп в справочнике задается свойством Иерархический объект конфигурации Справочник. В этом случае элемент справочника, представляющий собой группу, будет являться *родителем* для всех элементов и групп, входящих в эту группу. Такой вид иерархии называется *иерархией групп и элементов*.

Возможен и другой вид иерархии – *иерархия элементов*. В этом случае в качестве родителя выступает не группа элементов справочника, а непосредственно один из элементов справочника. Например, такой вид иерархии можно использовать при создании справочника Подразделения, когда одно подразделение является родителем для нескольких других, входящих в его состав.

Элементы одного справочника могут быть подчинены элементам или группам другого справочника. Например, справочник ЕдиницыИзмерения может быть подчинен справочнику Товары. Тогда для каждого элемента справочника Товары мы сможем указать единицы измерения, в которых этот товар поступает на склад. В системе 1С:Предприятие это достигается путем указания списка *владельцев* справочника для каждого объекта конфигурации Справочник.

Порой возникают ситуации, когда необходимо, чтобы в справочнике некоторые элементы существовали всегда, независимо от действий пользователя. Допустим, логика бизнес-процессов на предприятии такова, что все товары сначала поступают на основной склад, а затем по мере надобности перемещаются на другие склады. В этом случае в справочнике Склады всегда должен существовать склад Основной, иначе приходование товаров будет выполнено неправильно. Объект конфигурации Справочник позволяет описать любое количество таких элементов справочника. Они называются *предопределенными* элементами справочника.

В зависимости от того, какие действия мы хотим выполнять со справочником, нам требуется изображать справочник в «разном виде». Например, для того, чтобы выбрать некоторый элемент справочника, удобнее представить справочник в виде списка, а для того, чтобы изменить какой-то элемент справочника, удобнее представить все реквизиты этого элемента справочника на одной форме. Поэтому объект конфигурации Справочник может иметь произвольное количество форм, часть из которых можно назначить в качестве основных.

Следующая таблица поясняет названия этих форм, заданные в конфигураторе.

Таблица 1. Основные формы справочника

Объект конфигурации Справочник	Справочник
Основная форма объекта	Форма элемента справочника
Основная форма группы	Форма группы элементов справочника
Основная форма списка	Форма списка справочника
Основная форма для выбора	Форма для выбора элемента или группы справочника
Основная форма для выбора группы	Форма для выбора группы элементов справочника

Форма служит для «визуализации» данных, находящихся в базе. Она представляет эти данные в удобном для пользователя виде и позволяет описать алгоритмы, которые будут сопровождать работу пользователя с данными, показанными в форме.

Любая форма может быть описана в конфигураторе. Для создания такого описания существует подчиненный объект конфигурации Форма. Как правило, он подчинен одному из прикладных объектов, но может существовать и самостоятельно. На основании описания, содержащегося в объекте конфигурации Форма, в нужный момент работы пользователя платформа 1С:Предприятие создаст программный объект Форма, с которым и будет работать пользователь.

Лабораторная работа №2. Создание справочника «Клиенты»

ООО «На все руки мастер» оказывает услуги по ремонту бытовой техники, очевидно, что для ведения учета потребуется задать некоторую списочную информацию.

Для начала понадобится список сотрудников предприятия, которые будут оказывать услуги. Затем будет нужен список клиентов, с которыми работает наше ООО. После этого понадобится перечень услуг, которые может оказывать наше предприятие, и список материалов, которые могут быть израсходованы. Кроме этого потребуется список складов, на которых могут находиться материалы ООО «На все руки мастер».

Для создания списка сотрудников необходимо:

1. Открыть в конфигураторе учебную конфигурацию и создать новый объект конфигурации Справочник. Задача – создать справочник, в котором будут храниться наименования клиентов.

2. После того, как вы нажмете Добавить, система откроет *окно редактирования объекта конфигурации*.

Это средство создано в помощь разработчику. Оно предназначено специально для сложных объектов конфигурации и позволяет путем выполнения последовательных действий быстро создавать такие объекты. Для того чтобы придерживаться правильной последовательности действий, в нижней части окна имеются кнопки Далее и Назад. Кнопка Далее позволяет задавать свойства объекта в нужной последовательности, чтобы ничего не пропустить и не проскочить вперед, где потребуются данные, которые должны были быть введены ранее. Кнопка Назад позволяет вернуться на несколько шагов назад, если вы обнаружили, что ранее ввели не все или ошибочные данные.

3. Задайте *имя* справочника – Клиенты. Имя является основным свойством любого объекта конфигурации. При создании нового объекта система автоматически присваивает ему некоторое имя. Можно использовать имя, присвоенное системой, но лучше заменить его своим, понятным именем. Имя можно задавать любое, главное, чтобы оно начиналось с буквы и не содержало некоторых специальных символов (например, пробел). Для удобства чтения конфигурации принято составлять интуитивно понятные имена, и если они состоят из нескольких слов, удалять пробелы между словами и каждое слово начинать с большой буквы.

4. На основании имени платформа автоматически создаст *синоним* – Клиенты. Свойство Синоним также есть у любого объекта конфигурации. Оно предназначено для хранения «альтернативного» наименования объекта конфигурации, которое будет использовано в элементах интерфейса программы, то есть будет показано пользователю (рис. 6).

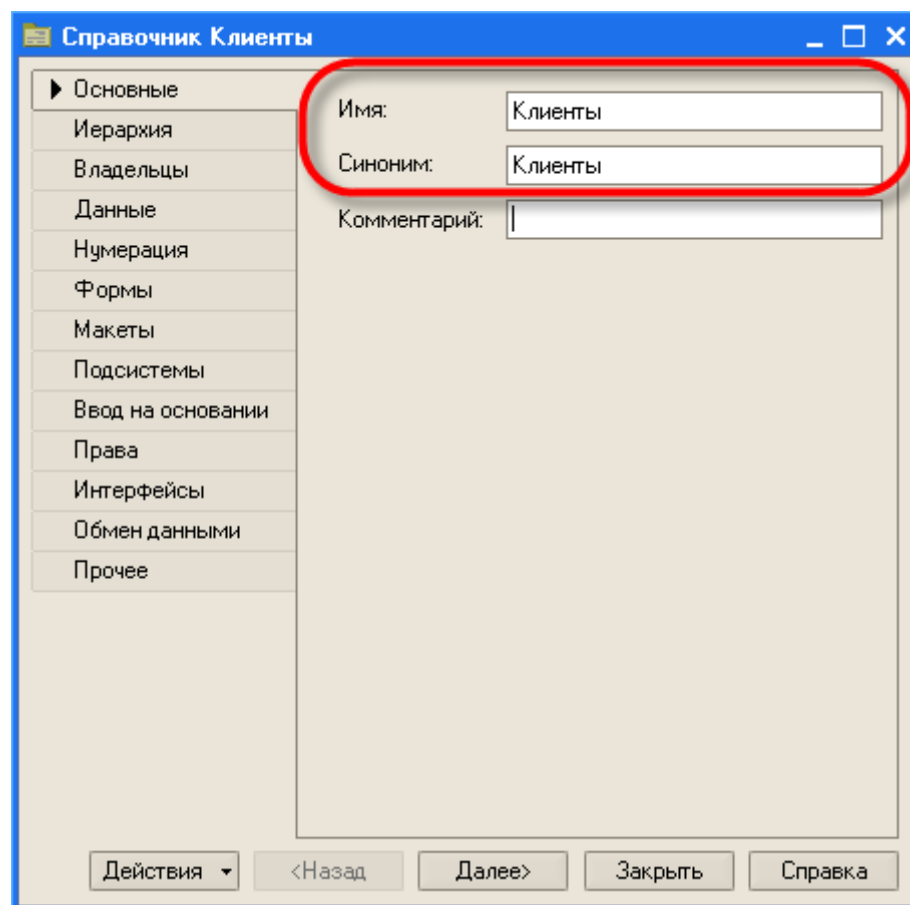


Рис. 6. Зададим имя и синоним справочника

Поэтому на синоним практически нет никаких ограничений, и его можно задавать в привычном для человека виде.

5. Все свойства объекта конфигурации Справочник настраивать нет необходимости, оставим те значения, которые система предлагает для них по умолчанию. Поэтому три раза нажмем Далее и окажемся на закладке Данные.

6. Здесь необходимо указать *длину кода* и *длину наименования*. Длина кода – важное свойство справочника. Как правило, код справочника используется для идентификации элементов справочника и содержит уникальные для каждого элемента справочника значения. Платформа может сама контролировать уникальность кодов и поддерживать автоматическую нумерацию элементов справочника.

Уникальность кодов платформа может отслеживать сама, поэтому от длины кода будет зависеть количество элементов, содержащихся в справочнике. Длина кода – 9 символов. В результате мы сможем использовать коды от 1 до 999999999 – этого вполне достаточно для нашего небольшого ООО «На все руки мастер».

Перейдем к длине наименования. 25 символов для нас явно мало, увеличим длину наименования до 50 (рис. 7).

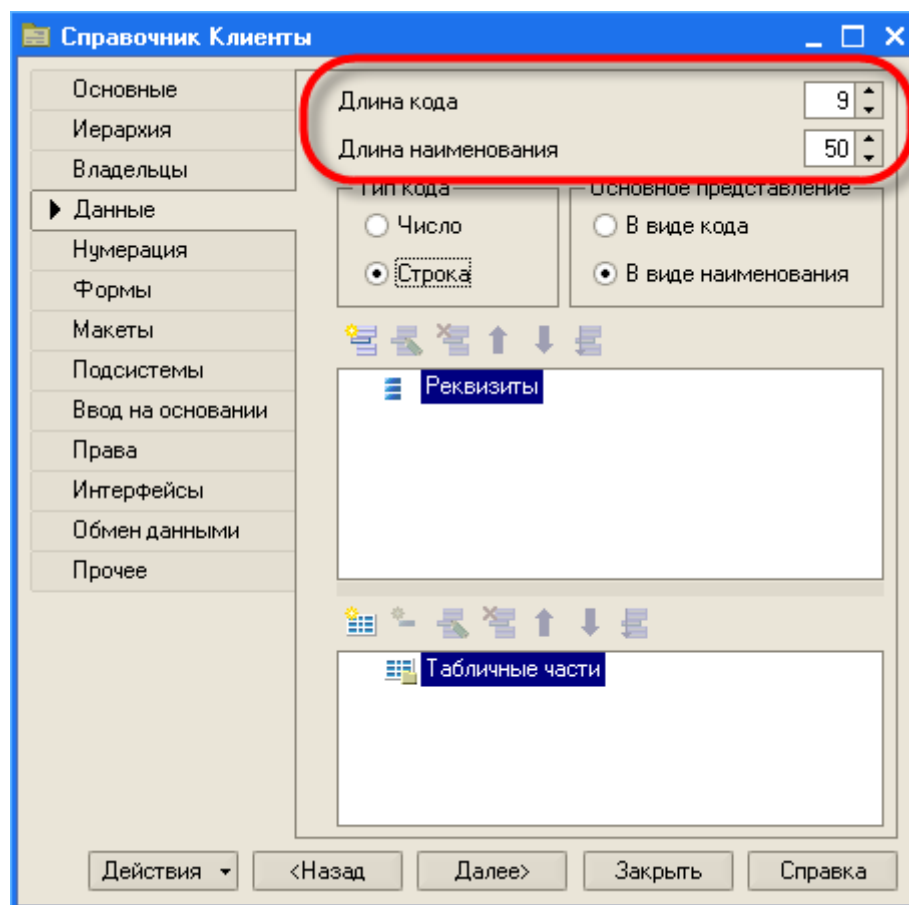


Рис. 7. Согласимся с длиной кода и зададим длину наименования

7. Все остальные свойства объекта конфигурации Справочник мы оставим такими, как их предлагает система по умолчанию, и нажмем Закреть.

Теперь давайте посмотрим, что создаст система на основе того объекта конфигурации Справочник, который мы добавили.

1. Выполним пункт меню Отладка > Начать отладку.
2. На появившийся вопрос (рис. 8):

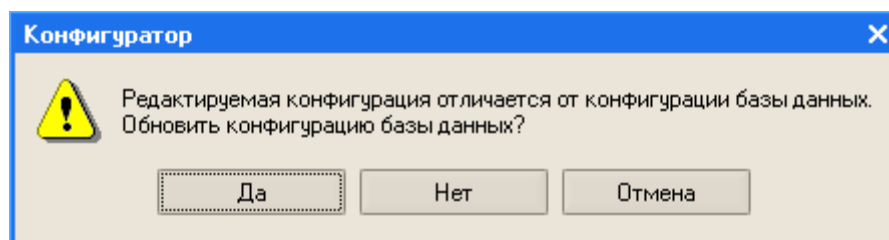


Рис. 8. Диалоговое окно с вопросом

ответим Да и в следующем окне нажмем Принять (рис. 9):

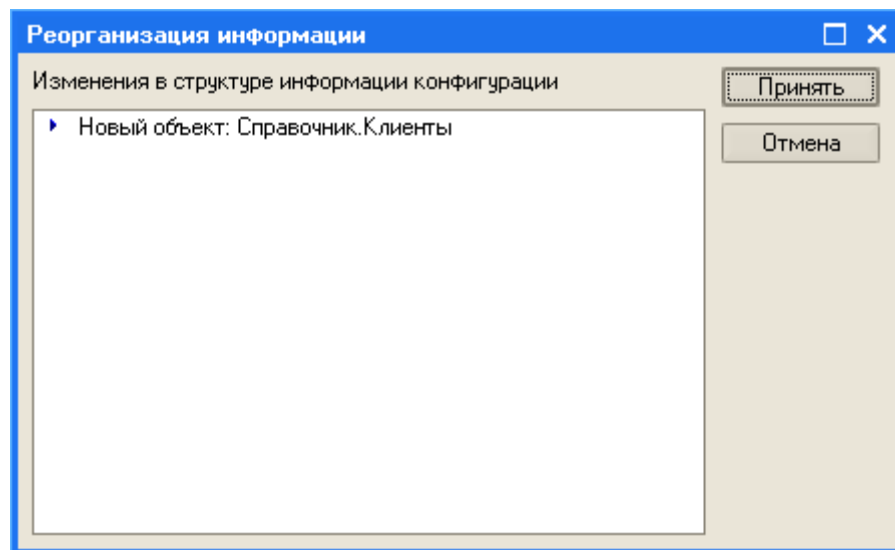


Рис. 9. Окно списка изменений структуры конфигурации

3. Перед нами откроется окно системы в режиме 1С:Предприятие. Поскольку мы не создавали никаких интерфейсов пользователей, для просмотра результатов нашей работы следует воспользоваться меню, которое создает система по умолчанию.

Выполним пункт меню **Операции > Справочник...** и выберем справочник **Клиенты** (рис. 10).

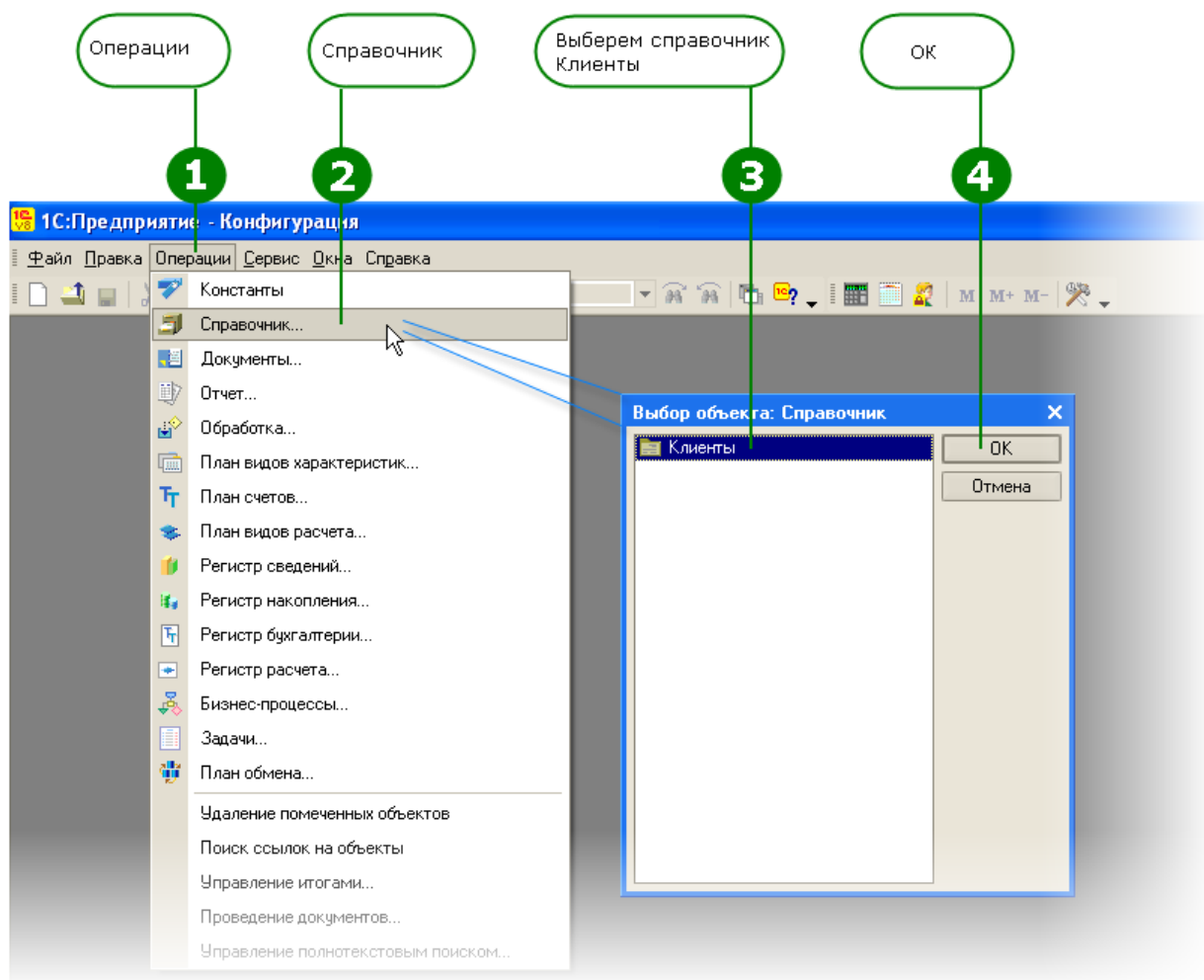


Рис. 10. Выполним Операции - Справочник... - Клиенты

4. Система откроет одну из основных форм справочника — *основную форму списка*.

Пока наш справочник Клиенты пуст, поэтому добавим в него несколько элементов (добавить новый элемент в справочник можно при помощи пункта меню Действия > Добавить, соответствующей иконки или клавишей Insert) (рис. 11).

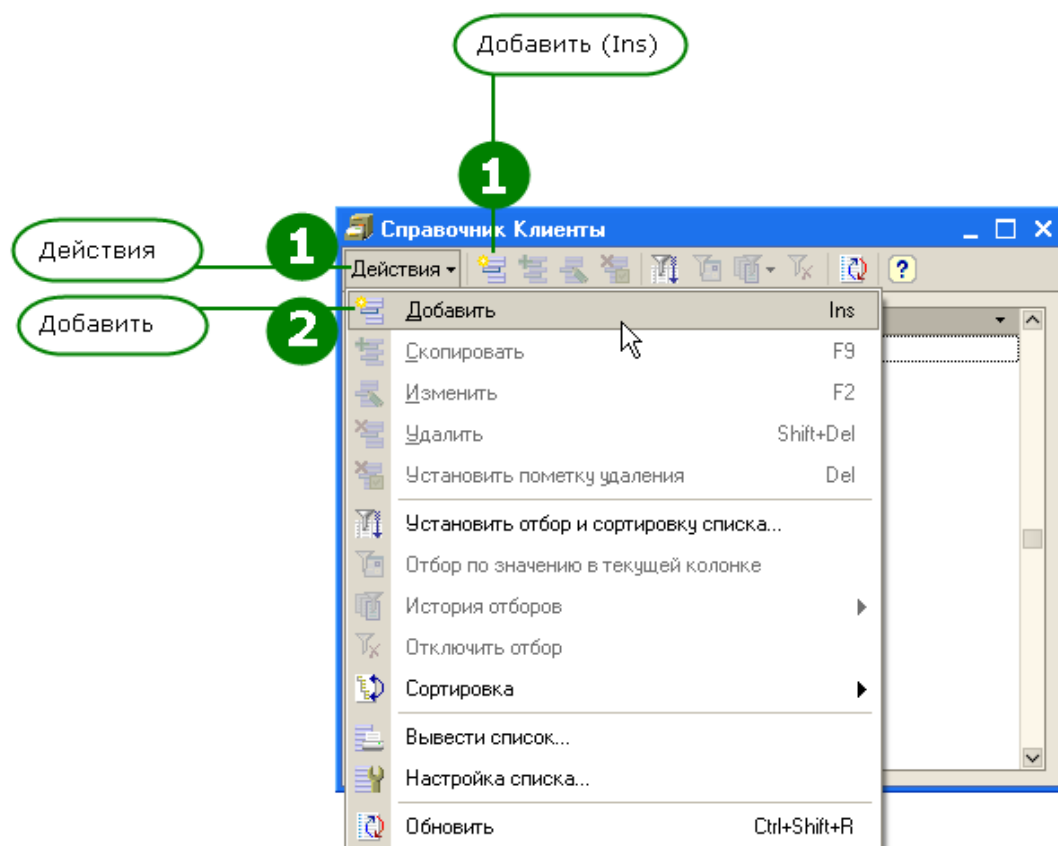


Рис. 11. Добавим новые элементы в справочник при помощи меню

После добавления элементов справочник будет выглядеть следующим образом (рис. 12):

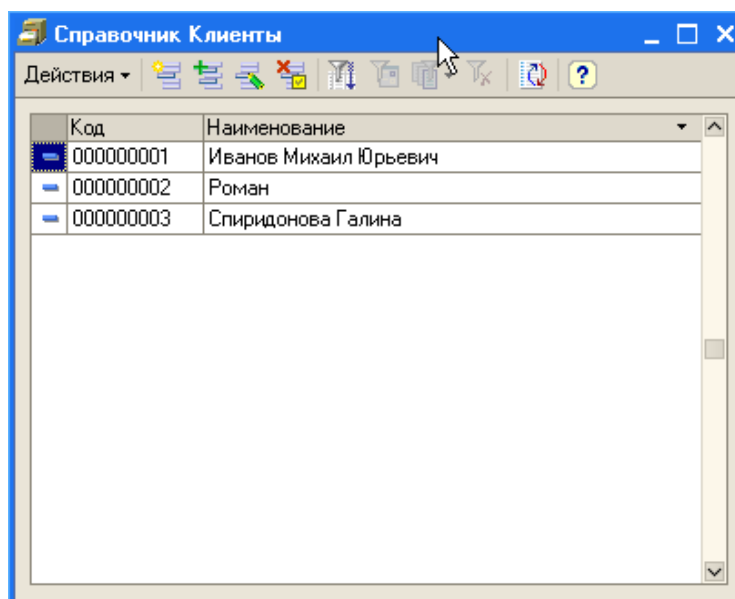


Рис. 12. Элементы справочника «Клиенты»

Теперь мы можем перейти к созданию второго справочника, который будет использоваться в нашей конфигурации, – справочника Сотрудники.

Лабораторная работа №3. Создание справочника «Сотрудники»

Справочник Сотрудники будет устроен несколько сложнее, чем справочник Клиенты. Дело в том, что в нем будет храниться не только фамилия, имя и отчество сотрудника, но и информация о его прошлой трудовой деятельности. Эта информация однородна по своей структуре (организация, начало, окончание работы, занимаемая должность), но количество предыдущих мест работы у разных сотрудников может быть различным. Поэтому для хранения такой информации необходимо использовать табличную часть справочника.

1. Вернитесь в конфигуратор и создайте новый объект конфигурации Справочник. Назовите его Сотрудники. Задача – создать справочник, имеющий табличную часть.

2. На закладке Данные задайте длину кода – 9, тип кода – Строка, длину наименования справочника равной 50 символам и добавьте в справочник новую табличную часть с именем ТрудоваяДеятельность (рис. 13):

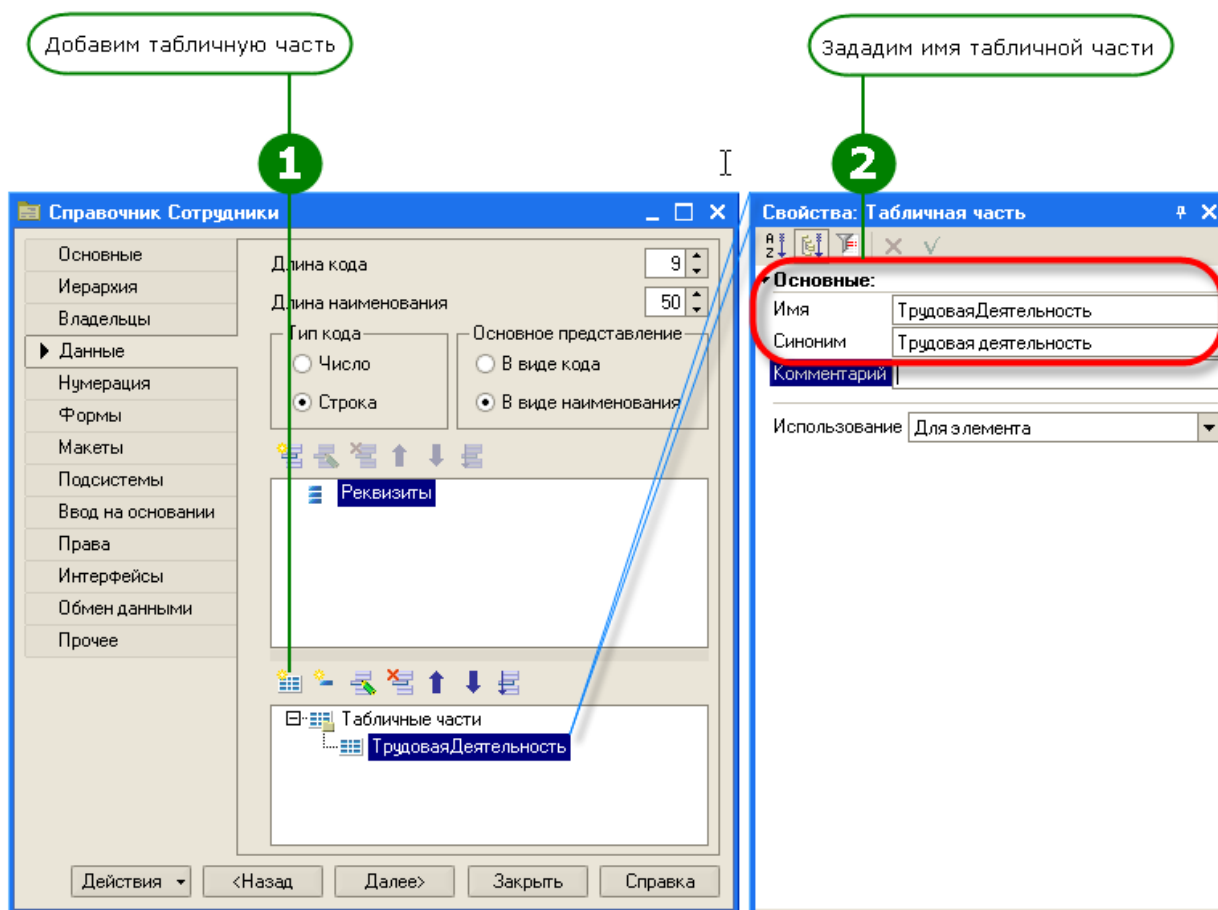


Рис. 13. Добавим в справочник «Сотрудники» новую табличную часть и зададим ее имя...

3. Создайте реквизиты табличной части ТрудоваяДеятельность (рис. 14):

- Организация – тип Строка, длина 100,
- НачалоРаботы – тип Дата, состав даты – Дата,
- ОкончаниеРаботы – тип Дата состав даты – Дата,
- Должность – тип Строка, длина 100:

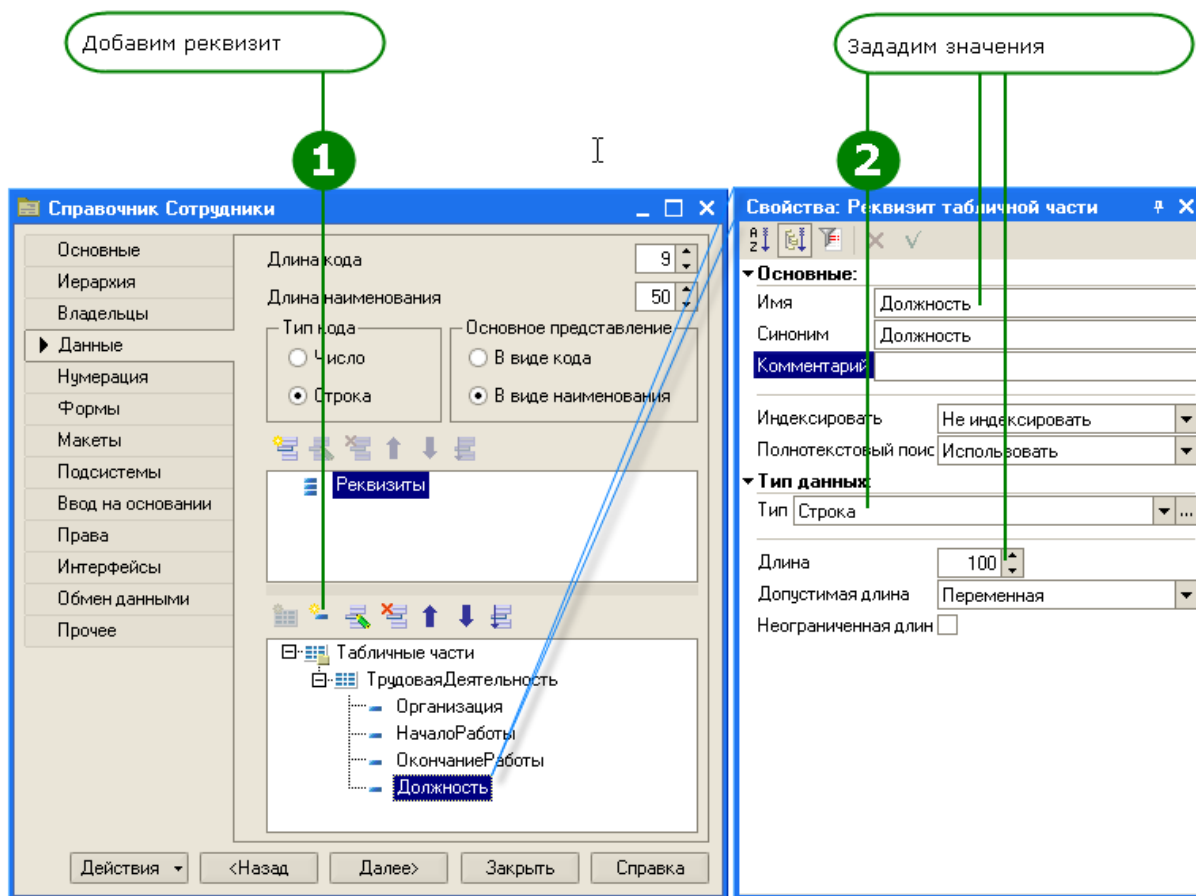


Рис. 14. Создадим реквизиты табличной части справочника...

Для реквизитов НачалоРаботы и ОкончаниеРаботы мы выбрали состав даты – Дата, поскольку в системе 1С:Предприятие 8 значения типа Дата содержат как дату, так и время. В данном случае время начала и окончания работы нам безразлично.

4. Теперь следует выбрать вариант редактирования справочника. Очевидно, что редактирование в списке уже не подойдет, поскольку в списке не возможно редактировать табличную часть справочника и вводить информацию о трудовой деятельности.

Поэтому в справочнике Сотрудники необходимо выбрать вариант редактирования справочника обоими способами – как в списке, так и в диалоге. Для этого перейдите на закладку Формы и установите соответствующий переключатель (рис. 15):

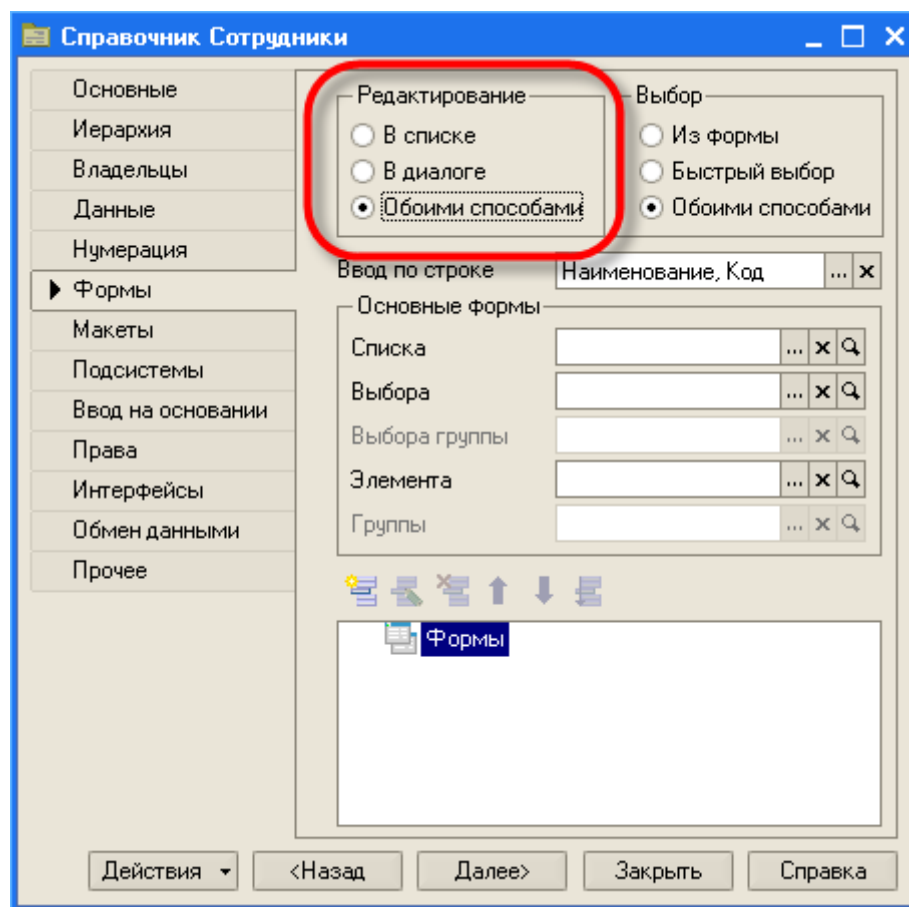


Рис. 15. Зададим вариант редактирования справочника...

5. Создание справочника Сотрудники завершено – теперь можно запустить 1С:Предприятие в режиме отладки и заполнить справочник.

Выполните пункт меню Отладка > Продолжить отладку, на вопрос системы (рис. 16):

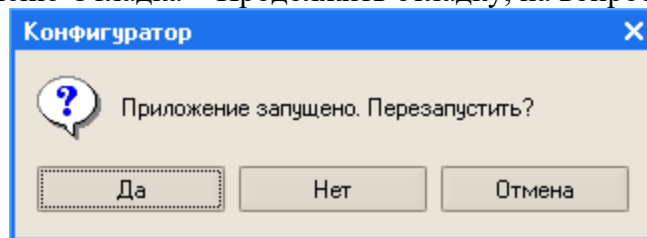


Рис. 16. Диалоговое окно с вопросом

ответьте Да, на последующие вопросы ответьте так же, как и ранее.

6. В появившемся окне программы откройте справочник Сотрудники и убедитесь, что установлен режим редактирования справочника в диалоге (меню Действия > Редактировать в диалоге или аналогичная иконка в командной панели).

7. Создайте сотрудников Гусакова Николая Дмитриевича, Делового Ивана Сергеевича и Симонова Валерия Михайловича, как показано на рисунках. Обратите внимание, что строки табличной части справочника можно сортировать по содержимому любой из колонок или располагать в произвольном порядке, используя иконки командной панели (рис. 17, 18, 19):

Сотрудники: Гусаков Николай Дмитриевич *

Действия ▾

Код: 000000001

Наименование: Гусаков Николай Дмитриевич

N	Организация	Начало работы	Окончание работы	Должность
1	ЗАО "НТЦ"	01.02.2000	16.04.2003	Ведущий специалист

OK Записать Закреть

Рис. 17. Заполнение элемента справочника «Сотрудники»

Порядок строк табличной части

Сотрудники: Новый *

Действия ▾

Код: 000000002

Наименование: Деловой Иван Сергеевич

N	Организация	Начало работы	Окончание работы	Должность
1	ООО "Автоматизация"	22.01.1996	31.12.2002	Инженер
2	НПО "СпецСвязь"	20.06.1986	21.01.1995	Начальник производства

OK Записать Закреть

Рис. 18. Можно управлять порядком расположения строк табличной части...

Сотрудники: Новый *

Действия ▾

Код: 000000003

Наименование: Симонов Валерий Михайлович

N	Организация	Начало работы	Окончание работы	Должность
1	000 "СтройМастер"	06.02.2001	03.04.2004	Прораб

OK Записать Закреть

Рис. 19. Заполнение элемента справочника «Сотрудники»
Далее следует создать следующий элемент справочника – Номенклатура.

Лабораторная работа №4. Создание справочника «Номенклатура»

Справочник Номенклатура будет содержать информацию об услугах, которые оказывает ООО «На все руки мастер», и о тех материалах, которые при этом могут быть использованы.

Этот справочник не будет сложным. Единственная особенность, которой он будет обладать, – это наличие иерархической структуры. Для того чтобы справочником было удобно пользоваться, мы сгруппируем услуги в одну группу, а материалы – в другую. Кроме этого, поскольку ООО «На все руки мастер» оказывает самые разные услуги, они также будут логически собраны в несколько групп. То же самое можно сказать и про материалы.

Задача будет состоять в создании иерархического справочника.

1. Создайте новый объект конфигурации Справочник и назовите его Номенклатура.
2. Перейдите на закладку Иерархия и установите флаг Иерархический справочник (рис. 20).
3. На закладке Данные задайте длину кода – 9, тип кода – Строка, длину наименования справочника равной 100 символам.
4. Теперь запустите 1С:Предприятие в режиме отладки и заполните справочник Номенклатура. В процессе заполнения справочника мы научимся создавать группы справочника и переносить элементы из одной группы в другую.

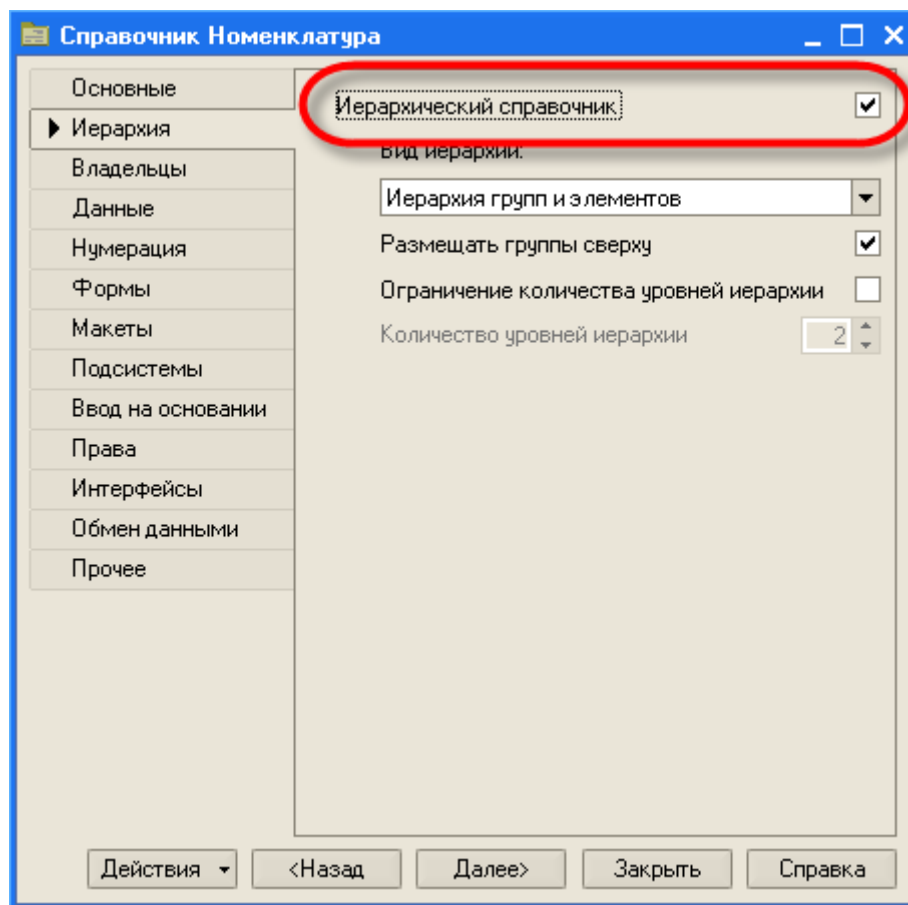


Рис. 20. Сделаем справочник иерархическим...

5. Создайте две группы в корне справочника: Материалы и Услуги (меню Действия > Новая группа) (рис. 21):

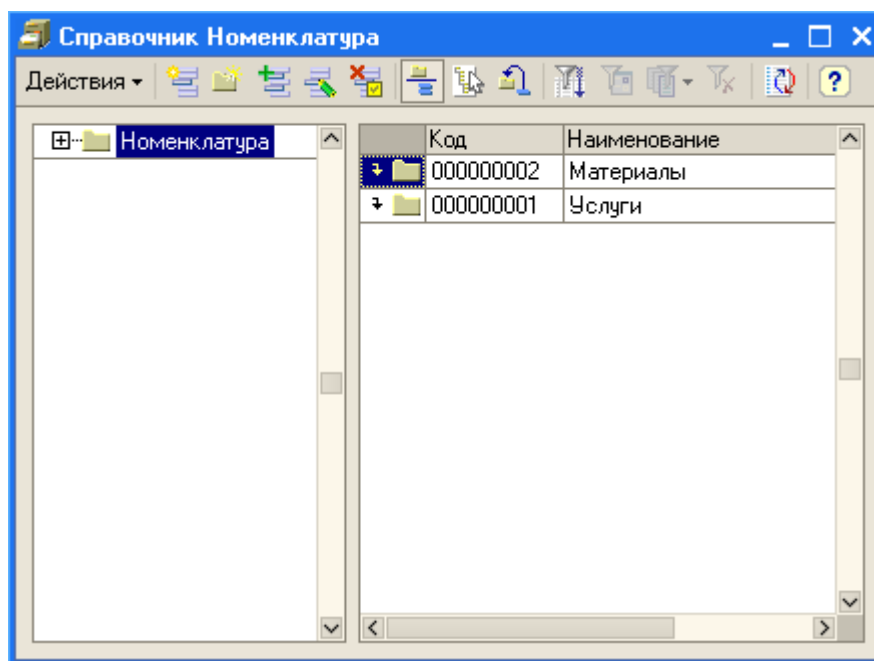


Рис. 21. Созданные группы в справочнике «Номенклатура»

6. В группе Материалы создайте пять элементов (рис. 22):

- Строчный трансформатор Samsung,
- Строчный трансформатор GoldStar,
- Транзистор Philips 2N2369,
- Шланг резиновый,
- Кабель электрический:
-

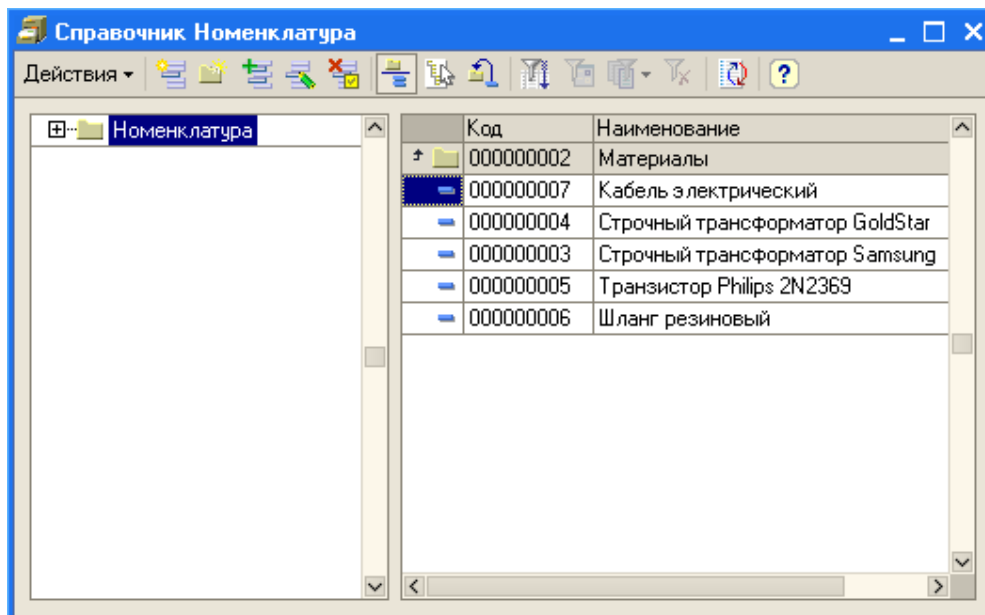


Рис. 22. Добавление элементов в группу «Материалы»

7. В группе Услуги создайте несколько элементов - услуги по ремонту телевизоров (рис. 23):

- Диагностика,
- Ремонт отечественного телевизора,
- Ремонт импортного телевизора.

И услуги по установке стиральных машин:

- Подключение воды,
- Подключение электричества:

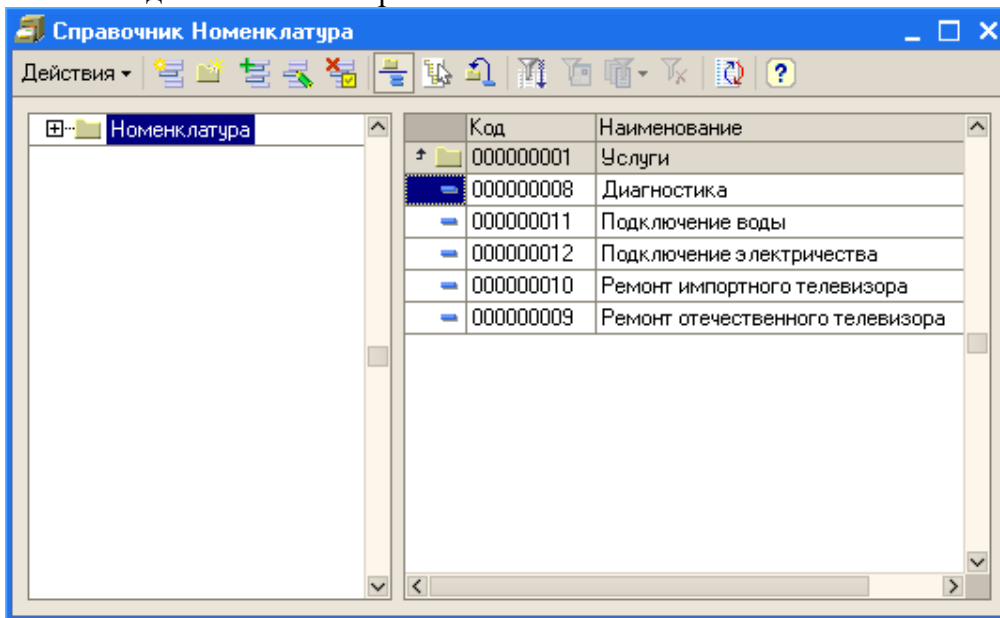


Рис. 23. Добавление элементов в группу «Услуги»

8. Теперь разнесите услуги по двум смысловым группам: услуги по ремонту телевизоров и услуги по установке стиральных машин. Для этого в группе Услуги создайте еще две группы: Телевизоры и Стиральные машины (рис. 24):

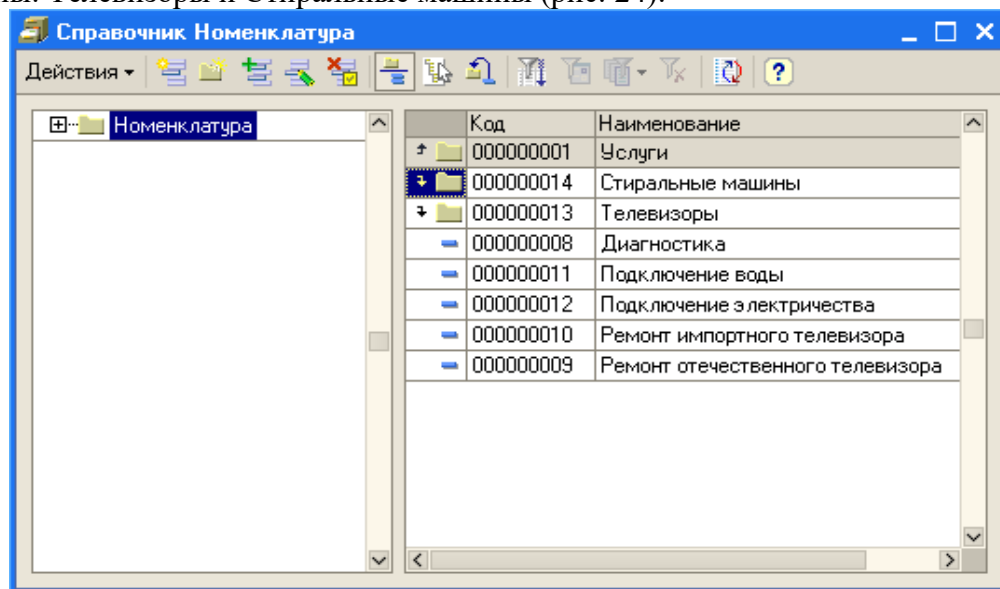


Рис. 24. Добавление групп «Телевизоры» и «Стиральные машины»

9. Для того чтобы переместить услуги в соответствующие группы, в окне списка установите курсор на ту услугу, которую нужно переместить, и выполните команду Действия > Переместить в группу. В открывшемся окне выберете новую группу (рис. 25):

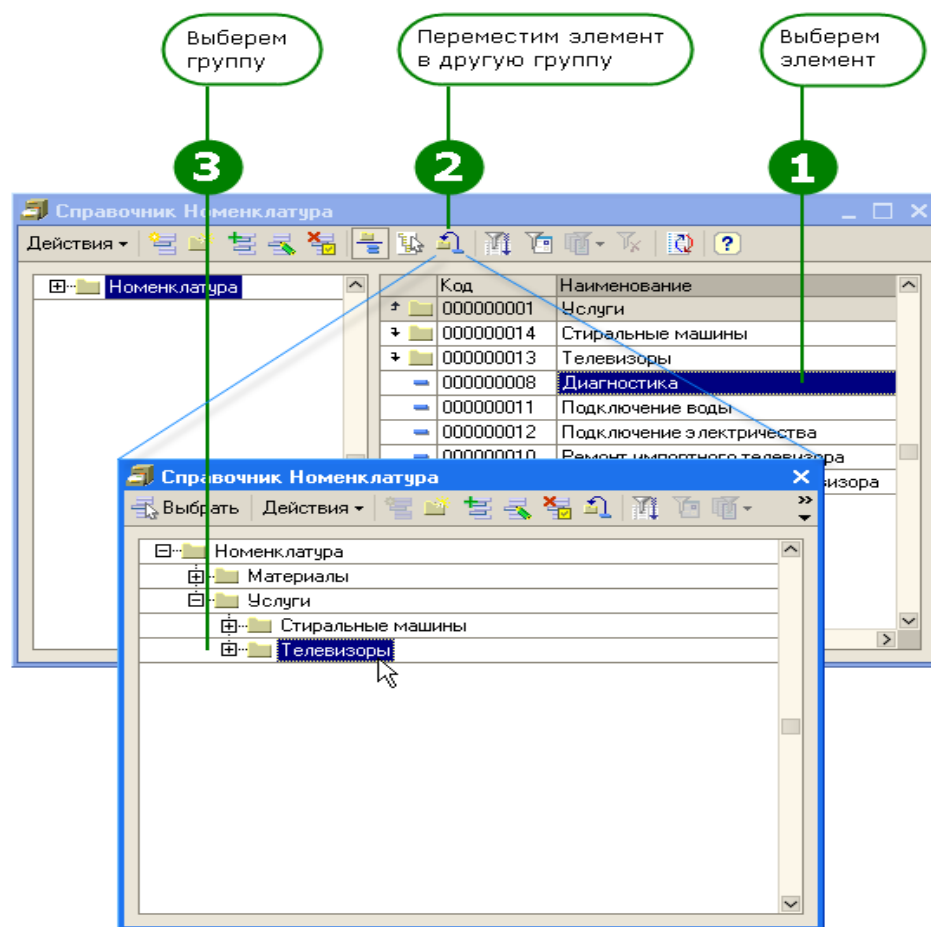


Рис. 25. Переместим услугу «Диагностика» в группу «Телевизоры»...

10. Аналогичным образом переместите в группу Телевизоры услуги Ремонт отечественного телевизора и Ремонт импортного телевизора. Услуги Подключение воды и Подключение электричества переместите в группу Стиральные машины.

11. Затем в группе Материалы создайте две группы: Радиодетали и Прочее. В группу Прочее поместите Кабель электрический и Шланг резиновый. Остальные материалы переместите в группу Радиодетали.

Лабораторная работа №5. Создание справочника «Склады»

В заключение мы создадим справочник Склады, который будет содержать информацию о складах, используемых ООО «На все руки мастер». Этот справочник будет содержать один predetermined элемент – склад Основной, на который будут поступать все материалы.

Задача – создать справочник, содержащий predetermined элементы.

1. Откройте конфигуратор и создайте новый объект конфигурации Справочник с именем Склады.
2. На закладке Данные установите длину кода – 9 и тип кода – Строка.
3. Перейдите на закладку Прочее и нажмите кнопку Предetermined.
4. Система откроет список predetermined элементов справочника. Сейчас он пуст, поэтому выполните команду Действия > Добавить и создайте predetermined элемент с именем Основной (рис. 26):

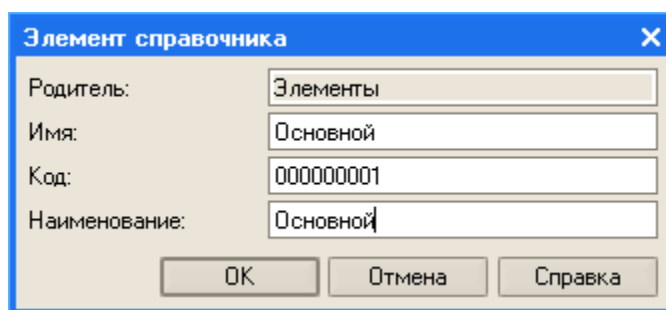


Рис. 26. Заполнение predetermined элемента справочника

Обратите внимание на то, что помимо наименования необходимо задать еще и *имя* predetermined элемента справочника. В дальнейшем, при использовании средства встроенного языка, будет возможность обратиться к этому элементу справочника, используя имя, которое присвоили ему в конфигураторе.

5. Запустите 1С:Предприятие в режиме отладки и откройте справочник Склады. Добавьте в справочник еще один склад с названием Розничный (рис. 27):

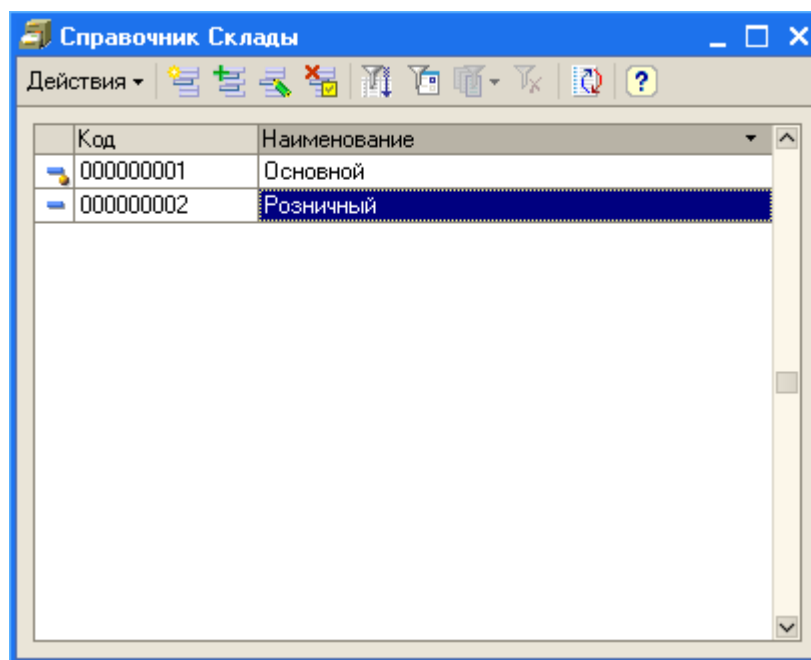


Рис. 27. Элементы справочника «Склады»

Обратите внимание, что система отмечает различными пиктограммами простой и предопределенный элементы справочника. Несмотря на то, что можно изменить код или наименование у обоих элементов, пометка на удаление (или удаление) возможна только для простых элементов справочника. При попытке пометить на удаление предопределенный элемент система выдаст предупреждение (рис. 28):

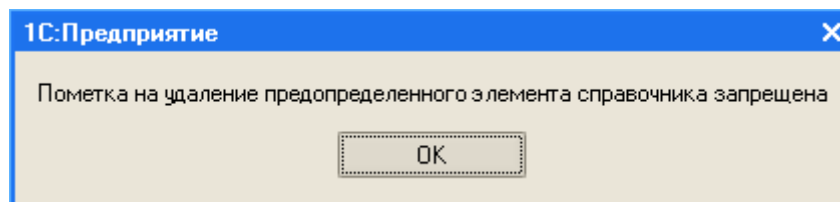


Рис. 28. Системное предупреждение

Таким образом, теперь можно выделить две характерные особенности предопределенных элементов:

- на предопределенные элементы могут опираться алгоритмы работы конфигурации (т.к. возможно обращение к ним из встроенного языка по имени),
- предопределенные элементы являются объектами базы данных, которые нельзя удалить в режиме 1С:Предприятия.

Из этого видно, в чем заключается принципиальная с точки зрения конфигурации разница между обычными и предопределенными элементами справочника.

Обычные элементы «непостоянны» для конфигурации. В процессе работы пользователя они могут появиться, исчезнуть. Поэтому конфигурация хоть и может отличить их друг от друга, но рассчитывать на них в выполнении каких-либо алгоритмов она не может в силу их «непостоянства».

Предопределенные элементы, напротив, «постоянны». В процессе работы пользователя они находятся всегда на своих местах, и исчезнуть не могут. Поэтому с ними конфигурация может работать вполне уверенно и опираться на них при отработке различных

алгоритмов. По этой причине каждый из predetermined элементов имеет уникальное имя для того, чтобы к нему можно было обратиться средствами встроенного языка.

КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1) для чего предназначен объект конфигурации Справочник?
- 2) каковы характерные особенности справочника?
- 3) для чего используются реквизиты и табличные части справочника?
- 4) зачем нужны иерархические справочники и что такое родитель?
- 5) зачем нужны подчиненные справочники и что такое владелец?
- 6) какие основные формы существуют у справочника?
- 7) что такое predetermined элементы?
- 8) чем, с точки зрения конфигурации, отличаются обычные элементы справочника от predetermined элементов?
- 9) что такое окно редактирования объекта конфигурации?
- 10) как создать объект конфигурации справочник и описать его структуру?
- 11) когда следует использовать редактирование справочника в списке, а когда – в диалоге?
- 12) как добавить новые элементы в справочник?
- 13) как создать группу справочника?
- 14) как пользователь может отличить обычные элементы справочника от predetermined элементов?
- 15) как переместить элементы из одной группы справочника в другую?
- 16) как запустить 1С:Предприятие в режиме отладки?
- 17) зачем нужна основная конфигурация и конфигурация базы данных?
- 18) как изменить конфигурацию базы данных?
- 19) как связаны объекты конфигурации и объекты базы данных?
- 20) зачем нужна палитра свойств?
- 21) что такое подчиненные объекты конфигурации?

Лабораторная работа №6. Объект конфигурации «Документ»

Цель:

1. Изучить область применения, структуру и свойства объекта конфигурации Документ.
2. Научиться создавать документы задавать собственные алгоритмы выполнения тех или иных действий, связанных с работой документа.
3. Изучить порядок создания *формы объекта* базы конфигурации, ознакомиться с некоторыми конструкциями *встроенного языка*.
4. Иметь представление о *типообразующих объектах* конфигурации.

Объект конфигурации Документ является прикладным и предназначен для описания информации о совершенных хозяйственных операциях или о событиях, произошедших в жизни организации вообще. На основе объекта конфигурации Документ платформа создает в базе данных информационную структуру, в которой будут храниться, например, такие документы, как приходные накладные, или приказы о приеме на работу, или платежные поручения, или счета и т. д.

Характерной особенностью объекта конфигурации Документ является то, что в процессе работы пользователь может самостоятельно создавать новые объекты этой структуры - новые документы.

Свойства документа отличается от свойств всех остальных объектов базы данных. Документ обладает способностью *проведения*. Факт проведения документа означает, что событие, которое он отражает, повлияло на состояние учета. До тех пор, пока документ не проведен, состояние учета неизменно, и документ – не более чем черновик, заготовка. Как только документ будет проведен – изменения, вносимые документом в учет, могут вступить в силу и состояние учета может быть изменено.

Поскольку документ вносит изменения в состояние учета, он всегда «привязан» к конкретному моменту времени. Это позволяет отражать в базе данных фактическую последовательность событий.

Следующим важным фактом, вытекающим из двух предыдущих, является то, что система 1С:Предприятие имеет механизмы, позволяющие отслеживать правильность состояния учета. Предположим, что мы изменили один из проведенных ранее документов и снова провели его «задним числом». В этом случае система 1С:Предприятие способна отследить, повлияют ли внесенные нами изменения на последующие проведенные документы, и если это так, система способна перепровести необходимые документы.

Каждый документ, как правило, содержит информацию, которая подробнее описывает этот документ. Например, каждый документ ПриходнаяНакладная может содержать информацию о поставщике товаров, складе, на который приходится товар, и т. д. Набор такой информации является одинаковым для всех документов одного вида, и для описания такого набора используются *реквизиты* объекта конфигурации Документ, являющиеся подчиненными объектами конфигурации. Большинство реквизитов объекта конфигурации Документ разработчик создает самостоятельно, однако у каждого объекта конфигурации Документ существуют два поля «по умолчанию»: Дата и Номер документа. Поскольку тип данных Дата содержит дату и время с точностью до секунды, этот реквизит и определяет в основном положение документа на оси времени.

Кроме этого, каждый документ содержит, как правило, некоторый набор информации, которая одинакова по своей структуре, но различна по количеству, предназначена для разных документов. Так, например, каждый документ ПриходнаяНакладная может содержать список

приходуемых товаров. Для описания подобной информации служат табличные части объекта конфигурации Документ.

Для «визуализации» документа существует несколько основных форм.

Таблица 1. Основные формы документа

Конфигуратор	База данных
Объект конфигурации Документ	Документ
Основная форма объекта	Форма документа
Основная форма списка	Форма списка документов одного вида
Основная форма для выбора	Форма для выбора из списка документов одного вида

Оперативное и неоперативное проведение

Факт проведения документа и необходимость поддержания актуальной последовательности документов на оси событий порождают два различных способа проведения документов: *оперативное* и *неоперативное* проведение.

С оперативным проведением документов связано понятие *оперативной отметки времени*. Оперативная отметка времени представляет собой значение типа Дата, которое формирует система. Оперативная отметка времени создается системой каждый раз при оперативном проведении документа. Ее значение формируется исходя из текущего времени и последней созданной оперативной отметки.

Если последняя оперативная отметка меньше текущего времени, в качестве новой оперативной отметки принимается текущее время.

Если последняя оперативная отметка равна или больше текущего времени, в качестве новой оперативной отметки принимается значение на одну секунду большее, чем старая оперативная отметка времени.

Таким образом, если у объекта конфигурации Документ установлено свойство оперативного проведения, последовательность действий системы будет следующей: при создании нового документа система будет устанавливать ему текущую дату и «нулевое» время.

При проведении такого документа (с текущей датой) система установит в качестве даты документа оперативную отметку времени. Если отменить проведение документа и затем провести его снова (не изменяя даты), система установит документу новую оперативную отметку времени.

Если же попытаться перепровести документ, то будет выдан запрос о виде проведения (оперативное или нет).

В случае оперативного проведения система установит новую оперативную отметку времени, а при неоперативном проведении время документа будет сохранено прежним.

При попытке проведения (или перепроведения) оперативно проводимого документа с датой, отличающейся от текущей, будет выдано сообщение о том, что оперативное проведение невозможно, и предложено провести документ неоперативно (то есть с сохранением существующей даты и времени документа).

Типообразующие объекты

Прежде чем мы приступим к практическому созданию документов, необходимо сделать отступление о том, какие типы данных могут использоваться в системе 1С:Предприятие.

В лабораторной работе, когда мы создавали реквизиты справочников или табличных частей, мы всегда указывали тип значения, которое может принимать этот реквизит. Это были

примитивные типы данных: Число, Строка, Дата и Булево. Примитивные типы данных изначально определены в системе и их набор ограничен.

Наряду с такими изначально определенными в любой конфигурации типами могут существовать типы данных, определяемые только конкретной конфигурацией. Такие типы сами образуют объекты конфигурации в момент их создания в конфигураторе.

Например, после того, как мы создали объект конфигурации Справочник Склады, сразу же появилось несколько новых типов данных, связанных с этим справочником. Среди них, например, СправочникСсылка.Склады. И если теперь мы укажем какому-либо реквизиту этот тип данных, то сможем хранить в нем ссылку на конкретный объект справочника Склады.

Объекты конфигурации, которые могут образовывать новые типы данных, называются *типообразующими*.

Лабораторная работа №7. Создание документа «ПриходнаяНакладная»

После того, как мы познакомились с объектом конфигурации Документ, создадим несколько таких объектов, чтобы иметь возможность фиксировать события, происходящие в ООО «На все руки мастер».

Одними из самых популярных услуг предприятия является ремонт телевизоров и установка стиральных машин. И в том, и в другом случае требуются некоторые материалы, которые расходуются в процессе оказания этих услуг. Поэтому двумя важнейшими событиями в хозяйственной жизни организации будут являться поступление материалов и оказание услуг. Для отражения этих событий в базе данных создадим два документа: Приходная накладная и Оказание услуги. Документ ПриходнаяНакладная будет фиксировать факт поступления в организацию необходимых материалов, а документ ОказаниеУслуги – фиксировать оказание услуг и расход материалов, которые используются при оказании этих услуг. Для этого:

1. Создайте новый объект конфигурации Документ.
2. Задайте имя документа – ПриходнаяНакладная.
3. Нажмите Далее.
4. Создайте реквизит документа с именем Склад и типом СправочникСсылка.Склады (рис. 1):

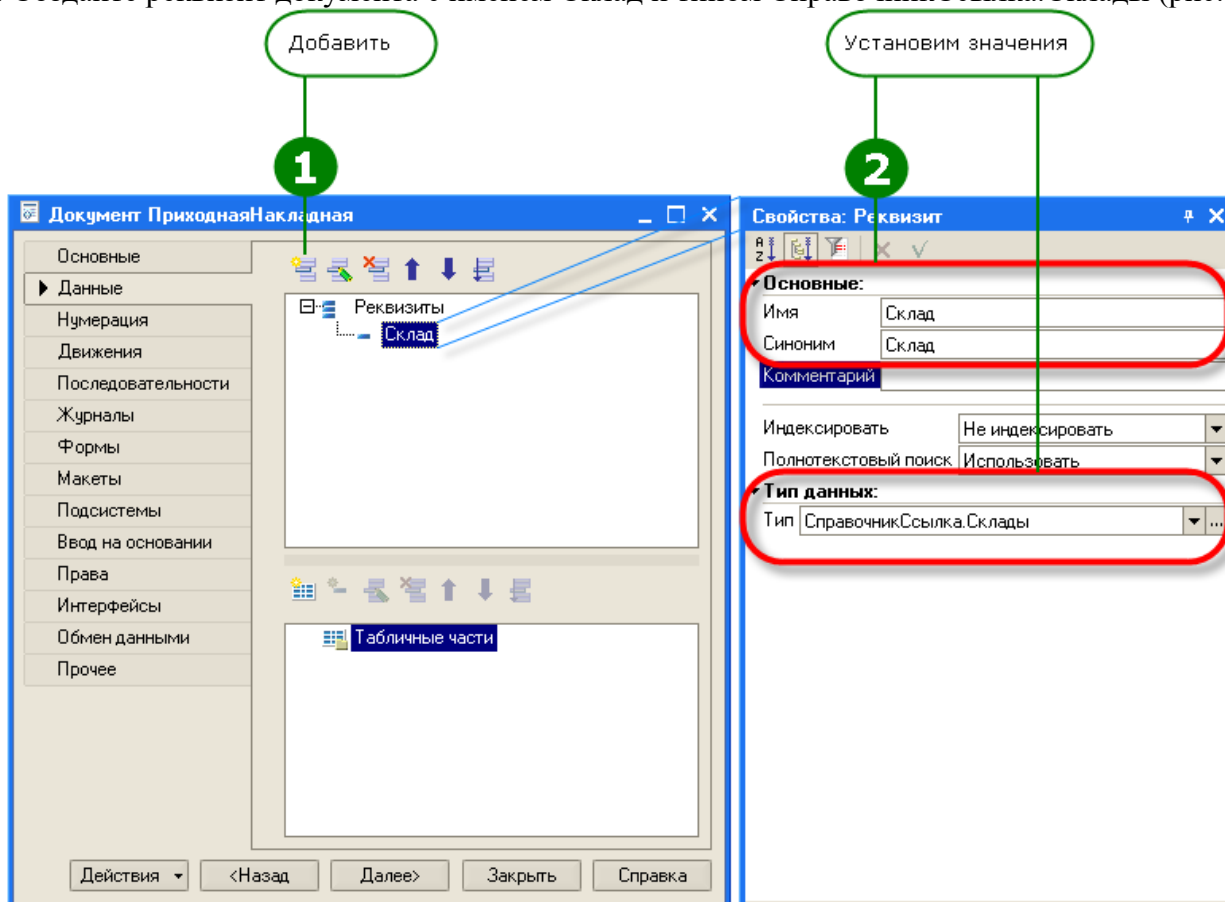


Рис. 1. Создание реквизита документа

5. Добавьте табличную часть с именем Материалы и создайте у нее четыре реквизита (рис. 2):
 - Материал с типом СправочникСсылка.Номенклатура;

- Количество с типом Число, длиной 15, точностью 3, неотрицательное;
- Цена с типом Число, длиной 15, точностью 2, неотрицательное;
- Сумма с типом Число, длиной 15, точностью 2, неотрицательное.

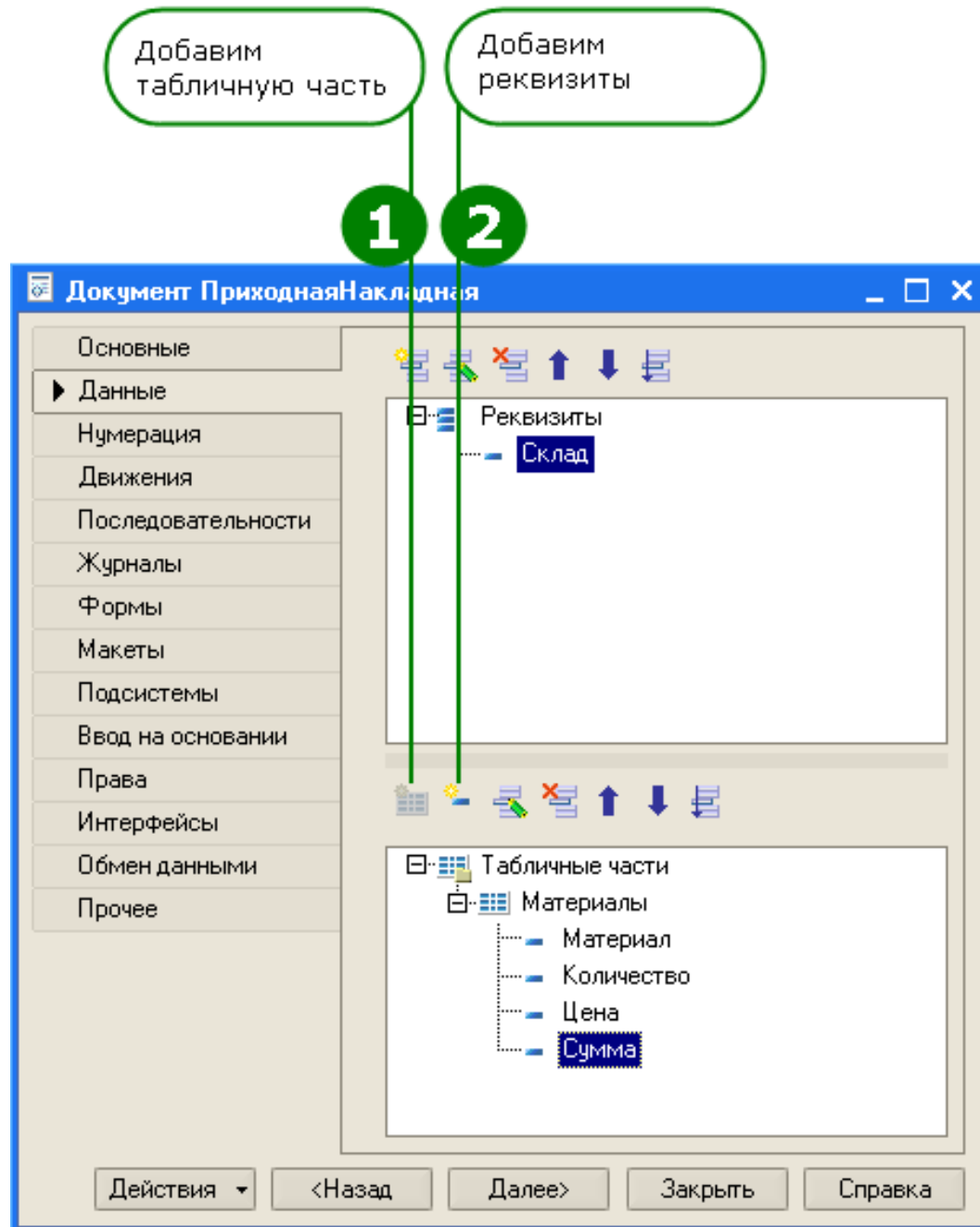


Рис. 2. Создадим табличную часть и опишем ее реквизиты

6. Перейдите на закладку Нумерация и задайте длину номера – 9 и тип номера – Строка.
7. Запустите 1С:Предприятие в режиме отладки и протестируйте получившийся результат: в отрывшемся окне программы выполните команду Операции > Документы..., выберете документ Приходная накладная. Система откроет одну из основных форм документа - основную форму списка.

Пока в нашей базе данных нет ни одного документа Приходная Накладная, поэтому создадим новый документ:

8. Выполните команду меню Действия > Добавить или нажмите соответствующую иконку, или нажмите клавишу Insert (рис. 3).

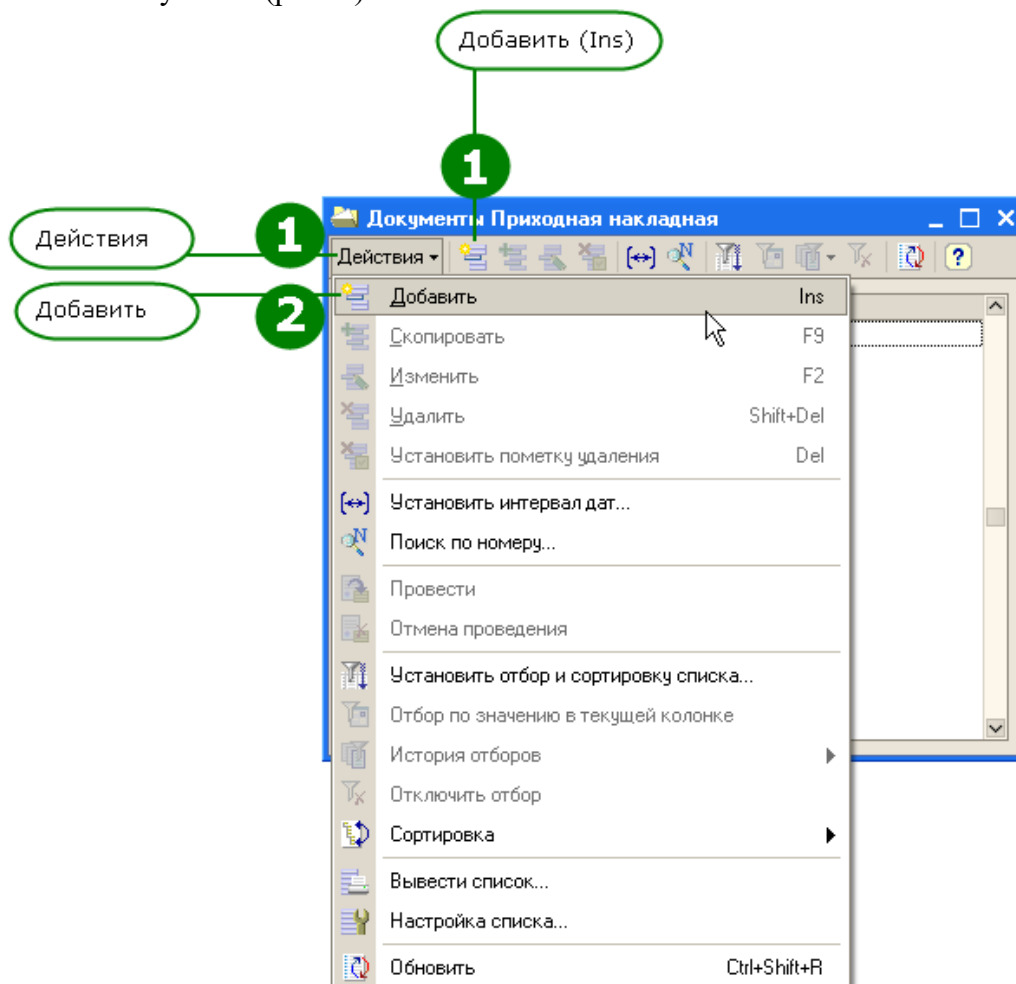


Рис. 3. Добавим новый документ при помощи меню или иконки

Система автоматически присвоит номер новому документу, и останется только заполнить табличную часть.

9. Заполните ее материалами для ремонта телевизоров так, как показано на рисунке (рис. 4):

N	Материал	Количество	Цена	Сумма
1	Строчный трансформатор GoldStar	10,000	270,00	2 700,00
2	Строчный трансформатор Samsung	10,000	600,00	6 000,00
3	Транзистор Philips 2N2369	10,000	3,00	30,00

Рис. 4. Заполнение документа «ПриходнаяНакладная №1»

10. Нажмите ОК.

11. Аналогичным образом создайте второй документ, который будет приходить следующие материалы для установки стиральных машин (рис. 5):

N	Материал	Количество	Цена	Сумма
1	Кабель электрический	5,000	20,00	100,00
2	Шланг резиновый	5,000	100,00	500,00

Рис. 3.5. Заполнение документа «ПриходнаяНакладная №2»

Наверняка вы обратили внимание на то, что при заполнении документа приходится вводить сумму в каждой строке. Это неудобно, и возникает естественное желание автоматизировать работу документа так, чтобы сумма вычислялась автоматически каждый раз при изменении цены или количества материалов в строке.

Для этого нам потребуется сначала создать собственную форму документа, а затем воспользоваться возможностями встроенного языка.

Лабораторная работа №8. Создание формы документа

Прежде всего, следует заметить, что до сих пор мы использовали predetermined forms of objects, which the system 1С:Enterprise created for us itself «by default».

Теперь же у нас возникла необходимость слегка изменить логику работы формы документа, а значит, нам придется создать свою собственную форму документа Приходная Накладная для того, чтобы в ней мы могли описать тот алгоритм, который нам нужен. Для этого:

1. Вернитесь в конфигуратор и откройте окно редактирования объекта конфигурации Документ Приходная Накладная. В этом окне нас интересует закладка Формы. Как мы видим, ни одна из основных форм документа пока не задана.
2. Нажмите на символ лупы в поле ввода (рис. 6):

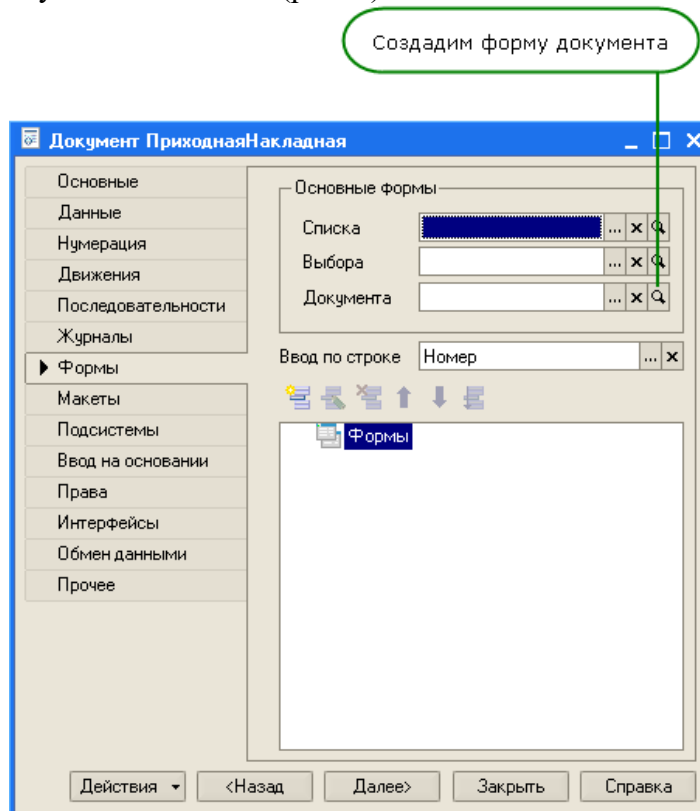


Рис. 6. Создадим форму документа...

3. Система вызовет *конструктор форм*. Этот инструмент также построен по принципу «мастеров»: ввод данных в определенной последовательности и передвижение кнопками Далее и Назад (рис. 7).

Рис. 7. Конструктор форм

4. Выберите тип формы ФормаДокумента и нажмите кнопку Готово, согласившись тем самым со всем, что нам предложила система.

5. Обратите внимание, что в дереве объектов конфигурации у объекта конфигурации Документ ПриходнаяНакладная появилась форма ФормаДокумента, а на экране открылось окно редактора форм, содержащее эту форму (рис. 8):

Рис. 8. Новая форма документа и окно редактора форм

Как видите, форма документа Приходная накладная содержит большое количество всевозможных полей. Эти поля называются *элементами управления*. Они имеют разное назначение и разное поведение, которое соответствует их назначению. Однако все они

служат для того, чтобы отображать информацию, хранящуюся в базе данных, и организовывать интерактивную работу с этой информацией. Сейчас мы обратим свое внимание только на те элементы управления, которые нас интересуют, – это три поля ввода, расположенные в колонках Количество, Цена и Сумма (рис. 9):

Поля ввода в колонках Количество, Цена и Сумма

Приходная накладная

Действия ▾

Номер:

Дата: .. : :

Склад: ... x

N	Материал	Количество	Цена	Сумма

OK Записать Закреть

Рис. 9. Интересующие нас элементы управления...

Мы хотим, чтобы каждый раз, когда меняется значение в поле Количество или в поле Цена, в поле Сумма автоматически устанавливалось значение, равное $\text{Количество} * \text{Цена}$. Очевидно, что для этого нужно написать на встроенном языке команду, похожую на $\text{Сумма} = \text{Количество} * \text{Цена}$, которая будет выполняться при изменении значения поля Количество или Цена.

Лабораторная работа №9. Создание процедуры обработки события в модуле формы

1. Щелкните правой кнопкой мыши на поле ввода в колонке Количество и откройте для него палитру свойств (пункт контекстного меню Свойства).

2. Прокрутите список до конца до перечня *событий*, которые могут быть связаны с этим полем ввода.

С большинством элементов управления связаны различные события, и именно с их помощью разработчик может влиять на ход работы программы.

3. Среди событий, связанных с полем ввода, найдите событие При изменении. Это событие возникает после изменения значения поля ввода.

4. Щелкните по кнопке с лупой в конце поля ввода, и система создаст заготовку процедуры обработчика этого события в *модуле* нашей формы (рис. 10).

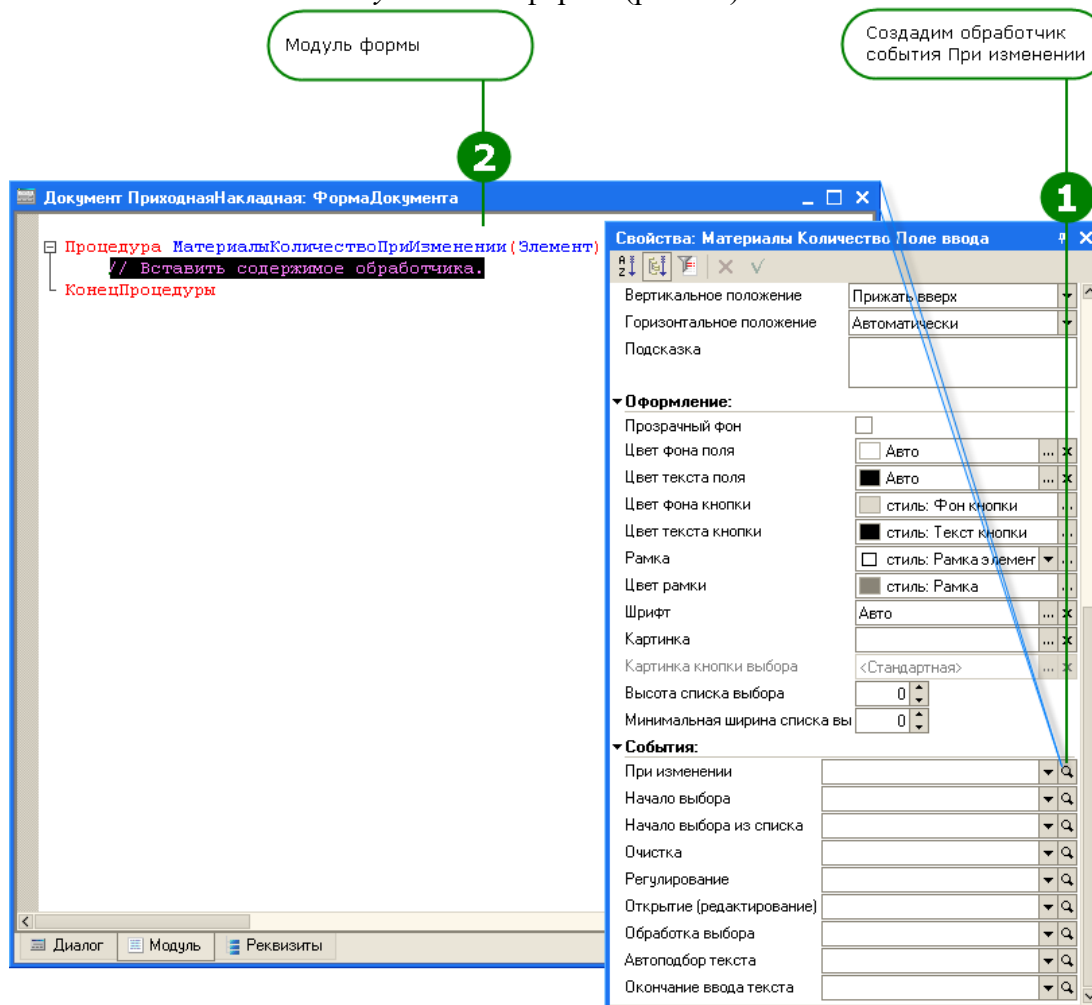


Рис. 10. Выберем событие «При изменении», и система создаст заготовку процедуры в модуле формы...

Модуль – это «хранилище» для текста программы на встроенном языке. В конфигурации существует большое количество модулей, которые расположены в различных ее точках. Они могут принадлежать некоторым объектам конфигурации (например, формам), а могут существовать сами по себе (принадлежать всей конфигурации в целом). Текст программы, содержащийся в модулях, будет использоваться платформой в заранее известные моменты работы системы 1С:Предприятие.

5. В модуль формы, в процедуру МатериалыКоличествоПриИзменении() добавьте следующий текст (листинг 1):

Листинг 1. Процедура МатериалыКоличествоПриИзменении()

```
СтрокаТабличнойЧасти = ЭлементыФормы.Материалы.ТекущиеДанные;  
СтрокаТабличнойЧасти.Сумма = СтрокаТабличнойЧасти.Количество *  
СтрокаТабличнойЧасти.Цена;
```

Объясним назначение этих строк.

В *первой строке* мы обращаемся к программному объекту ЭлементыФормы. Этот объект является коллекцией значений, содержащей все элементы управления, расположенные на нашей форме. Каждый элемент управления формы можно получить, указав его имя в качестве свойства объекта ЭлементыФормы. В данном случае мы обращаемся к элементу управления с именем Материалы (ЭлементыФормы.Материалы).

Этот элемент управления отображает строки табличной части нашего документа. Получить ту строку, в которой в настоящее время осуществляется редактирование, можно при помощи свойства программного объекта ТабличноеПоле – ТекущиеДанные. Таким образом, в результате выполнения первой строки переменная СтрокаТабличнойЧасти будет содержать объект ДокументТабличнаяЧастьСтрока.ПриходнаяНакладная.Материалы, в котором находятся редактируемые данные.

Во *второй строке* вычисляется сумма как произведение количества и цены. Объект ДокументТабличнаяЧастьСтрока.<имя> позволяет обратиться к данным конкретной колонки, указав имя колонки в качестве свойства объекта (например, СтрокаТабличнойЧасти.Количество).

Теперь необходимо проверить правильность работы процедуры.

6. В окне программы откройте список документов ПриходнаяНакладная и откройте любой из двух созданных документов. Если теперь поменять количество в любой строке документа, то сумма в строке будет пересчитана автоматически.

Если заглянуть вперед, то мы увидим, что подобное автоматическое заполнение поля Сумма может понадобиться и в других документах. Поэтому лучше будет поместить расчет суммы в некотором «общедоступном» месте, чтобы разные документы, имеющие аналогичные реквизиты табличной части, могли использовать этот алгоритм.

Для описания таких «общедоступных» мест служат объекты конфигурации Общий модуль, расположенные в ветке Общие> Общие модули. Процедуры и функции, содержащиеся в этих модулях, могут быть доступны для любых объектов конфигурации.

Создание процедуры обработки события в общем модуле

Для того чтобы алгоритм, выполняемый при обработке события, был доступен для разных документов, создадим общий модуль и перенесем в него процедуру расчета суммы. А в документе просто оставим вызовы этой процедуры из общего модуля. Для этого:

1. Создайте объект конфигурации Общий модуль в ветке Общие > Общие модули.
2. Назовите его РаботаСДокументами. Он будет содержать следующий текст (листинг 2):

Листинг 2. Процедура РассчитатьСумму()

Процедура РассчитатьСумму (СтрокаТабличнойЧасти) Экспорт

```
СтрокаТабличнойЧасти.Сумма = СтрокаТабличнойЧасти.Количество *  
СтрокаТабличнойЧасти.Цена;
```

КонецПроцедуры

Ключевое слово Экспорт в конце оператора Процедура как раз указывает на то, что эта процедура может быть доступна из других программных модулей.

3. Затем в модуле формы измените текст обработчика (листинг 3):

Листинг 3. Процедура МатериалыКоличествоПриИзменении()

Процедура МатериалыКоличествоПриИзменении (Элемент)

```
СтрокаТабличнойЧасти = ЭлементыФормы.Материалы.ТекущиеДанные;
```

```
РаботаСДокументами.РассчитатьСумму (СтрокаТабличнойЧасти) ;
```

КонецПроцедуры

4. Проверьте правильность работы процедуры.

Теперь осталось и для поля Цена установить такой же обработчик. Так как однажды мы уже написали в модуле формы нужную нам процедуру, то мы просто могли бы сопоставить ее также и другому событию другого элемента управления, расположенного в форме. Однако стандарты разработки конфигураций фирмы «1С» не допускают такого решения.

5. Создайте обработчик события При изменении для поля ввода, которое расположено в колонке Цена, и повторите в нем вызов процедуры РассчитатьСумму из общего модуля (листинг 4):

Листинг 4. Процедура МатериалыЦенаПриИзменении()

Процедура МатериалыЦенаПриИзменении (Элемент)

```
СтрокаТабличнойЧасти = ЭлементыФормы.Материалы.ТекущиеДанные;
```

```
РаботаСДокументами.РассчитатьСумму (СтрокаТабличнойЧасти) ;
```

КонецПроцедуры

6. Запустите 1С:Предприятие в режиме отладки и убедитесь, что теперь сумма в строках табличной части документов Приходная накладная пересчитывается как при изменении количества, так и при изменении цены.

Лабораторная работа №10. Создание документа «Оказание услуги»

Теперь аналогичным образом создадим второй документ, необходимый нам, – Оказание услуги. Для этого потребуется выполнить следующие действия:

1. Создать новый объект конфигурации Документ ОказаниеУслуги с реквизитами:
 - Склад, тип СправочникСсылка.Склады;
 - Клиент, тип СправочникСсылка.Клиенты;
 - Мастер, тип СправочникСсылка.Сотрудники.
2. Создать табличную часть этого документа ПереченьНоменклатуры с реквизитами:
 - Номенклатура, тип СправочникСсылка.Номенклатура;
 - Количество, тип Число, длина 15, точность 3, неотрицательное;
 - Цена, тип Число, длина 15, точность 2, неотрицательное;
 - Сумма, тип Число, длина 15, точность 2, неотрицательное.
3. Создать основную форму документа,
4. Для полей ввода колонок Количество и Цена создать обработчик события ПриИзменении, в котором вызывать процедуру РассчитатьСумму() из общего модуля РаботаСДокументами.

В результате документ ОказаниеУслуги будет выглядеть следующим образом (рис. 11):

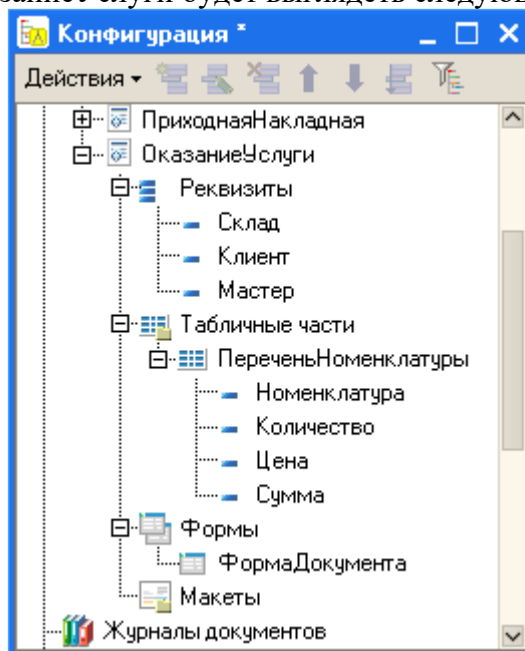


Рис. 11. Документ «ОказаниеУслуги»

5. После того, как эти действия будут выполнены, запустите 1С:Предприятие в режиме отладки и убедитесь, что при вводе цены и количества в табличную часть документа ОказаниеУслуги сумма пересчитывается по правильному алгоритму.

Контрольные вопросы

- 1) для чего предназначен объект конфигурации Документ
- 2) какими характерными особенностями обладает документ
- 3) для чего предназначены реквизиты и табличные части документа
- 4) какие существуют основные формы документа
- 5) что такое проведение документа

- 6) чем отличается оперативное проведение документа от неоперативного
- 7) что такое оперативная отметка времени
- 8) как создать объект конфигурации Документ и описать его основную структуру
- 9) как создать новый документ и заполнить его данными
- 10) как создать собственную форму документа
- 11) что такое конструктор форм
- 12) что такое элементы управления
- 13) что такое события и с чем они связаны
- 14) что такое обработчик события и как его создать
- 15) что такое модуль и для чего он нужен
- 16) зачем нужны общие модули
- 17) как сделать процедуру доступной в разных модулях
- 18) как назначить обработчик события
- 19) что такое типобразующие объекты

Лабораторная работа №11. Назначение регистра накопления

Цель:

1. Изучить назначение, структуру и отличительные особенности объекта конфигурации Регистр накопления.
2. Научиться создавать регистры накопления.

Один из главных моментов разработки любой конфигурации – созданию механизма учета накопления данных.

Не смотря на то, что в конфигурации создано что расходовать и приходить (справочники), и так же есть чем расходовать и приходить (документы), автоматизация ООО «На все руки мастер» не закончена.

Во-первых, путем анализа документов можно, конечно, получить требуемые нам выходные данные, но представьте, что завтра ООО «На все руки мастер» решит немного изменить свои бизнес-процессы, и нам потребуется ввести в конфигурацию еще один документ (или несколько документов!).

Например, сейчас мы полагаем, что товары поступают в ООО и затем расходуются.

Руководство захотело усилить материальный контроль и решило приходить товары на основной склад организации и затем выдавать их материально ответственным лицам. В этом случае нам придется добавить в конфигурацию еще один документ, который будет фиксировать перемещение материалов между основным складом и материально ответственными лицами. И очевидно, нам придется переработать все отчеты, которые были нами созданы к этому моменту с тем, чтобы они учитывали изменения, вносимые новым документом. А представьте, если в нашей конфигурации не два, а двадцать документов?!

Во-вторых, отчеты, анализирующие документы, будут работать довольно медленно, что будет вызывать раздражение пользователей и недовольство руководителей.

Поэтому в системе 1С:Предприятие есть несколько объектов конфигурации, которые позволяют создавать в базе данных структуры, предназначенные для накопления информации в удобном для последующего анализа виде.

Использование таких «хранилищ» данных позволяет, с одной стороны, накапливать в них данные, поставляемые различными документами (или другими объектами базы данных), а с другой стороны, легко создавать нужные отчеты или использовать эти данные в алгоритмах работы конфигурации (рис. 1).

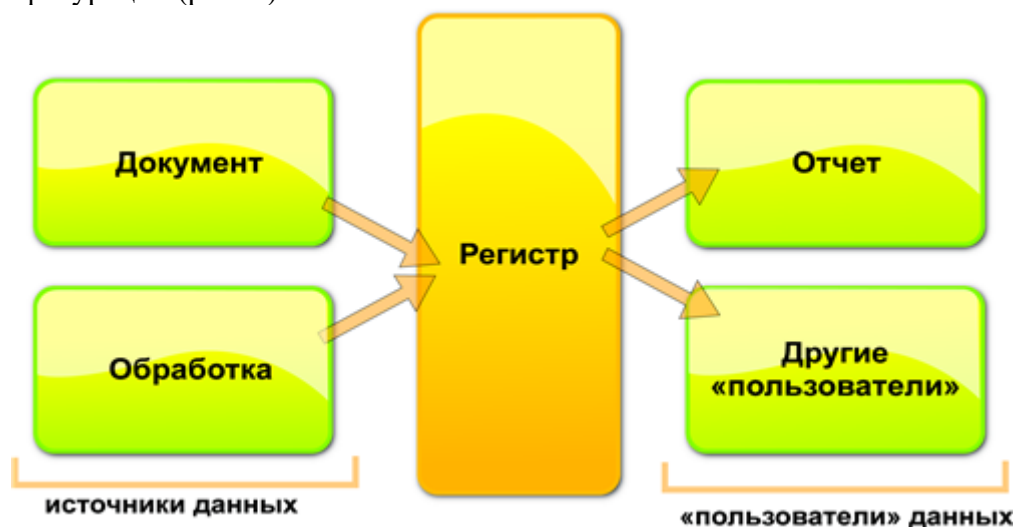


Рис. 1. Алгоритм работы конфигурации

В конфигурации существует несколько объектов, называемых регистрами, для описания подобных «хранилищ». рассмотрим один из них.

Объект конфигурации Регистр накопления

Объект конфигурации *Регистр накопления* является прикладным и предназначен для описания структуры накопления данных. На основе объекта конфигурации Регистр накопления платформа создает в базе данных информационную структуру, в которой будут накапливаться данные, «поставляемые» различными объектами базы данных. Эти данные будут храниться в регистре в виде отдельных записей, каждая из которых имеет одинаковую, заданную в конфигураторе структуру.

Отличительной особенностью регистра накопления является то, что он не предназначен для интерактивного редактирования пользователем. Разработчик может при необходимости предоставить пользователю возможность редактировать регистр накопления, но предназначение регистра накопления заключается в том, чтобы его модификация производилась на основе алгоритмов работы других объектов базы данных, а не в результате непосредственных действий пользователя.

Основным назначением регистра накопления является накопление числовой информации в разрезе нескольких *измерений*, которые описываются разработчиком в соответствующем объекте конфигурации Регистр накопления и являются подчиненными объектами конфигурации.

Виды числовой информации, накапливаемой регистром накопления, называются *ресурсами* и также являются подчиненными объектами и описываются в конфигураторе.

Например, регистр накопления может накапливать информацию о количестве и сумме товаров на складах. В этом случае он будет иметь измерения Товар и Склад, и ресурсы Количество и Сумма.

Изменение состояния регистра накопления происходит, как правило, при проведении документа, и заключается в том, что в регистр добавляется некоторое количество записей.

Каждая запись содержит значения измерений, значения приращений ресурсов, ссылку на документ, который вызвал эти изменения (регистратор) и «направление» приращения (приход или расход). Такой набор записей называется *движениями* регистра накопления.

Каждому движению регистра накопления всегда должен соответствовать регистратор.

Кроме того, регистр накопления может хранить дополнительную информацию, описывающую каждое движение. Набор такой дополнительной информации задается разработчиком при помощи *реквизитов* объекта конфигурации Регистр накопления.

Лабораторная работа №12. Создание регистра накопления «ОстаткиМатериалов»

Рассмотрим возможность применения регистра в нашем примере.

Прежде всего, нас интересует информация о том, сколько и каких материалов есть у нас на складах. Для накопления такой информации мы создадим регистр ОстаткиМатериалов. Для этого:

1. Откройте в конфигураторе учебную конфигурацию и создайте новый объект конфигурации Регистр накопления.

2. Задайте имя регистра – ОстаткиМатериалов.

3. Нажмите Далее и перейдите к созданию структуры регистра.

Создайте измерения регистра:

- Материал, с типом СправочникСсылка.Номенклатура,
- Склад, с типом СправочникСсылка.Склады.

Создайте ресурс Количество с длиной 15 и точностью 3.

4. В результате этих действий регистр ОстаткиМатериалов должен иметь следующий вид (рис. 2):

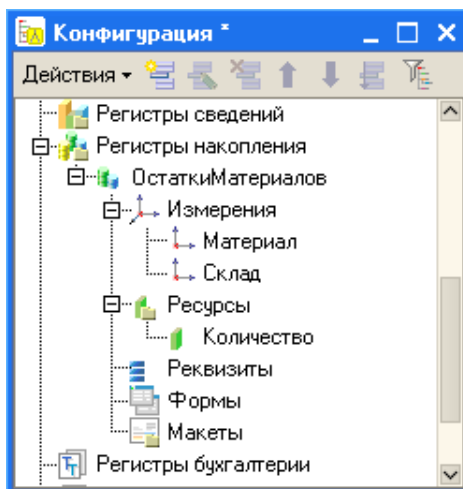


Рис. 2. Регистр «ОстаткиМатериалов»

Если сейчас запустить 1С:Предприятие в режиме отладки, то система выдаст сообщение об ошибке: РегистрНакопления.ОстаткиМатериалов: Ни один из документов не является регистратором для регистра. Это сообщение еще раз подтверждает тот факт, что назначение регистра накопления в том, чтобы аккумулировать данные, поставляемые различными документами.

Лабораторная работа №13. Создание движений документа «ПриходнаяНакладная»

Движения документа – это записи в регистрах, которые создаются в процессе проведения документа и отражают изменения, производимые документом. Для создания движения необходимо:

1. Откройте окно редактирования объекта конфигурации Документ ПриходнаяНакладная.
2. Перейдите на закладку Движения и в списке регистров конфигурации отметьте регистр накопления ОстаткиМатериалов (рис. 3):

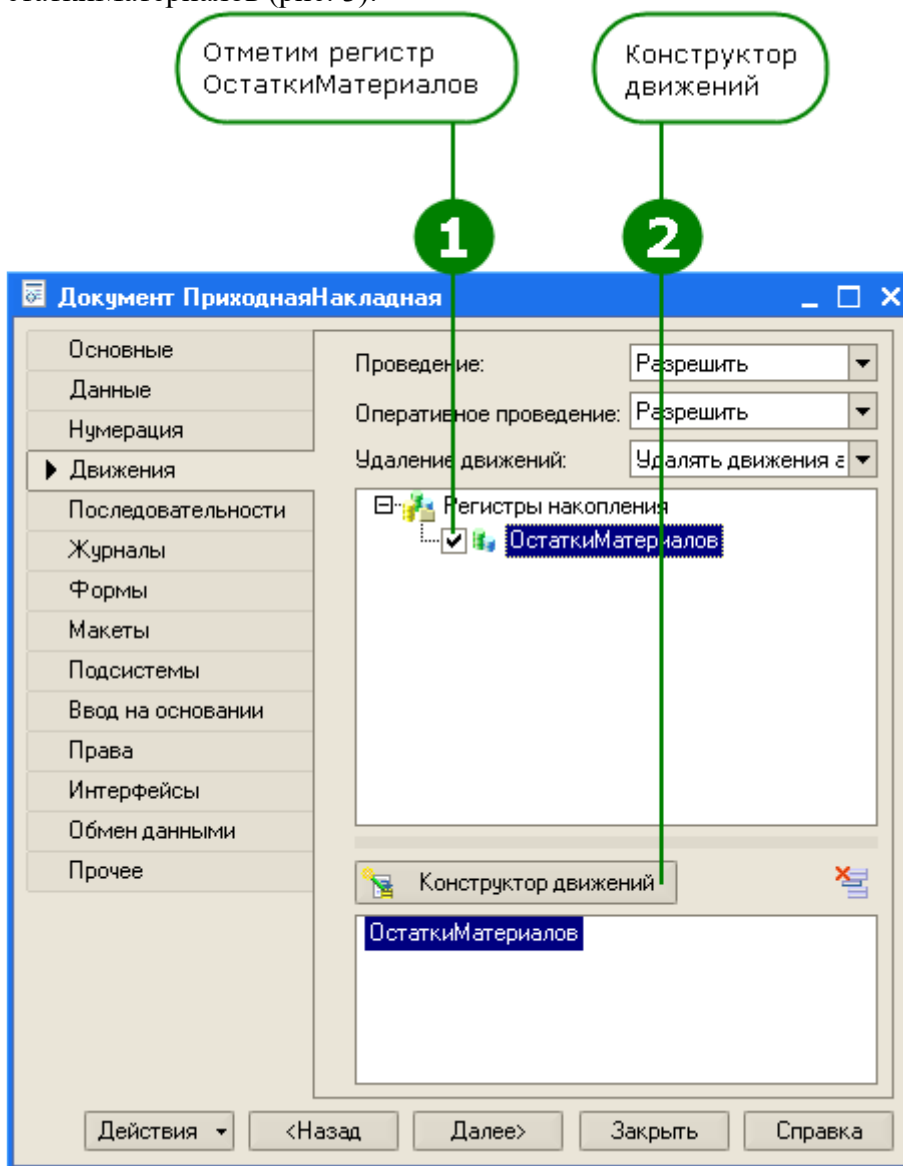


Рис. 3. Отметим регистр накопления и воспользуемся конструктором движений...

Обратите внимание, что сразу после отметки выбранного регистра становится доступной кнопка Конструктор движений. Этим конструктором мы и воспользуемся.

Конструктор устроен просто (рис. 4). В списке Регистры перечислены регистры, в которых документ может создавать движения. В нашем случае там пока один регистр ОстаткиМатериалов.

В списке Реквизиты документа должны находиться исходные данные для создания движений. А в таблице Поле – Выражение должны быть заданы формулы, по которым будут вычисляться значения измерений и ресурсов регистра при записи движений.

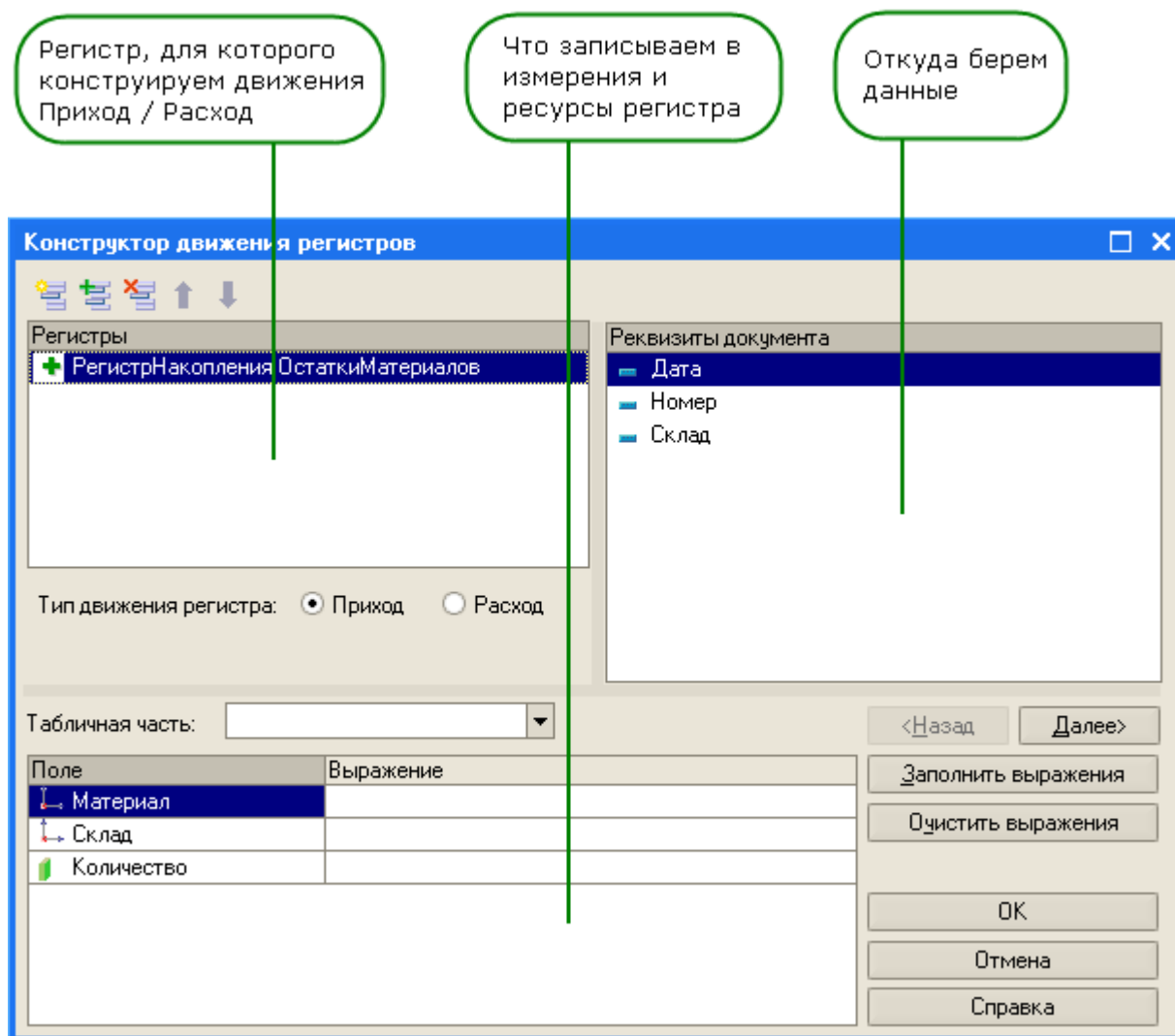


Рис. 4. Конструктор движений

Обратите внимание, что по умолчанию конструктор предлагает нам создавать движения

прихода (Тип движения регистра – Приход, символ **+** рядом с названием регистра) по регистру ОстаткиМатериалов. Это нас вполне устраивает, ведь документ ПриходнаяНакладная и должен приходовать материалы.

3. В поле выбора Табличная часть выберете табличную часть документа – Материалы. Список реквизитов документа автоматически заполнится реквизитами нашей табличной части.

4. Теперь нажмите кнопку Заполнить выражения.

В нижнем окне сформируется соответствие полей и выражений (рис. 5).

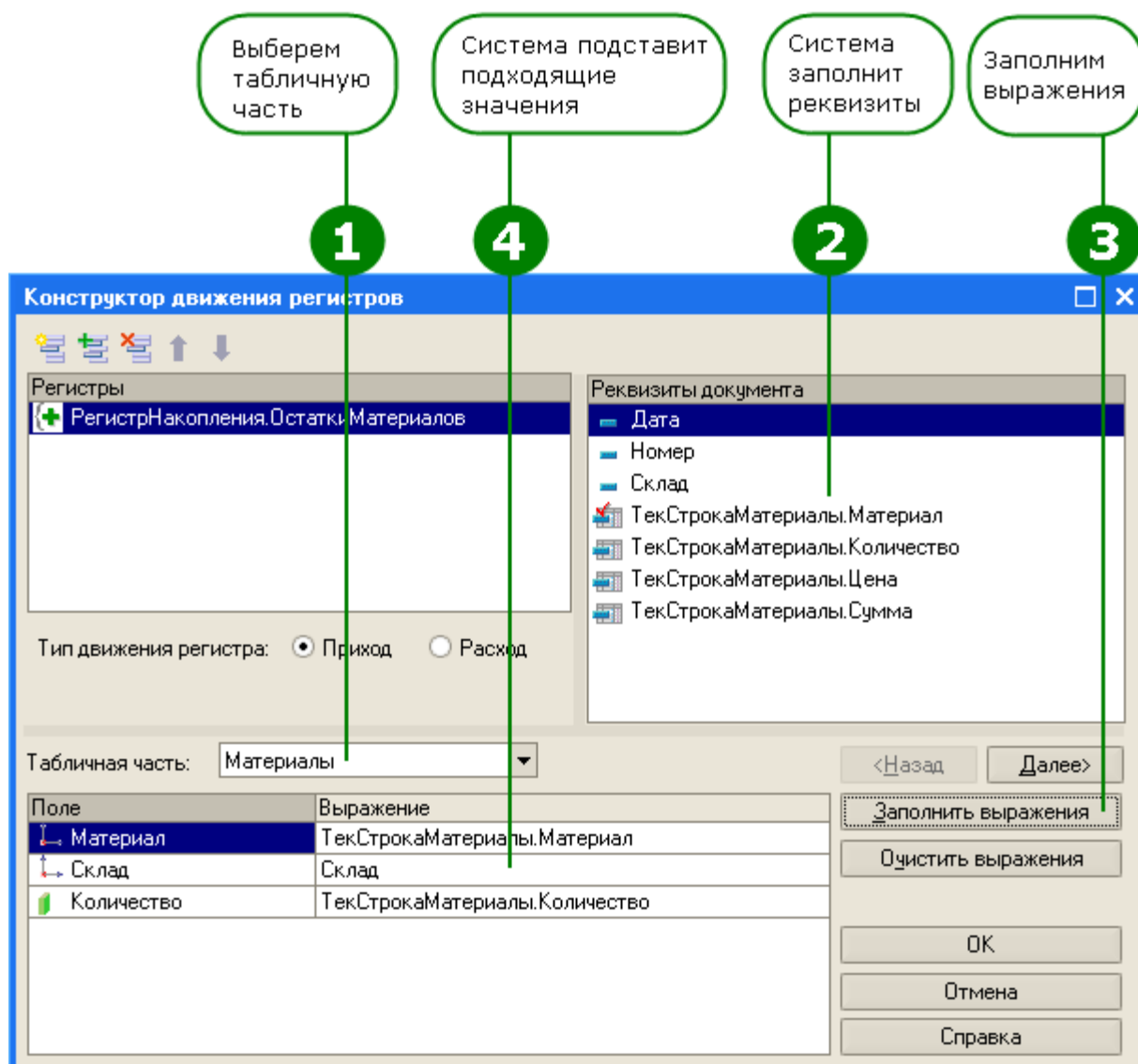


Рис. 5. Выберем табличную часть и нажмем «Заполнить выражения»...

Как видно, конструктор движений установил соответствия подходящим образом: в качестве материала в регистр будет записан материал из табличной части документа, в качестве склада – склад, указанный в шапке документа, а в качестве количества – количество из табличной части документа.

5. Нажмите кнопку ОК и посмотрите, какой текст сформировал конструктор в модуле объекта (листинг 1):

Листинг 1. Процедура ОбработкаПроведения()
Процедура ОбработкаПроведения(Отказ, Режим)

```

//{{__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора,
// внесенные вручную изменения будут утеряны!!!
Для Каждого ТекСтрокаМатериалы Из Материалы Цикл
    // Регистр ОстаткиМатериалов Приход
    Движение = Движения.ОстаткиМатериалов.Добавить ( ) ;
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Материал = ТекСтрокаМатериалы.Материал;
    Движение.Склад = Склад;
    Движение.Количество = ТекСтрокаМатериалы.Количество;
КонецЦикла;
// }}__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
КонецПроцедуры

```

Конструктор создал обработчик события ОбработкаПроведения объекта конфигурации Документ и поместил его в модуль объекта.

Внутри обработчика расположен цикл, который предназначен для перебора строк табличной части нашего документа. В цикле обращение к табличной части документа происходит по имени (Материалы), а строки табличной части документа представляют собой коллекцию значений, для перебора которой можно использовать конструкцию Для Каждого ... Из ... Цикл.

Объект встроенного языка ДокументОбъект имеет свойство Движения. Оно возвращает коллекцию наборов записей регистров, которые принадлежат этому документу. К набору записей документа, принадлежащему конкретному регистру, можно обратиться, указав через точку имя этого регистра.

Таким образом, в первой строке тела цикла мы добавляем к набору записей, который создает наш документ в регистре, новую запись и сохраняем ее в переменной Движение.

Затем мы присваиваем нужные значения всем полям этой записи и после перебора всех строк документа (после завершения цикла) «одним махом» записываем в регистр ОстаткиМатериалов весь набор записей движений документа.

Проверьте правильность работы регистра. Для этого:

1. Запустите 1С:Предприятие в режиме отладки.
2. Откройте одновременно два окна: список документов ПриходнаяНакладная и список регистра накопления ОстаткиМатериалов.
3. Откройте документ Приходная накладная №1 и нажмите ОК. Обратите внимание, что при проведении приходной накладной появляются соответствующие записи в регистрах накопления (рис. 6):

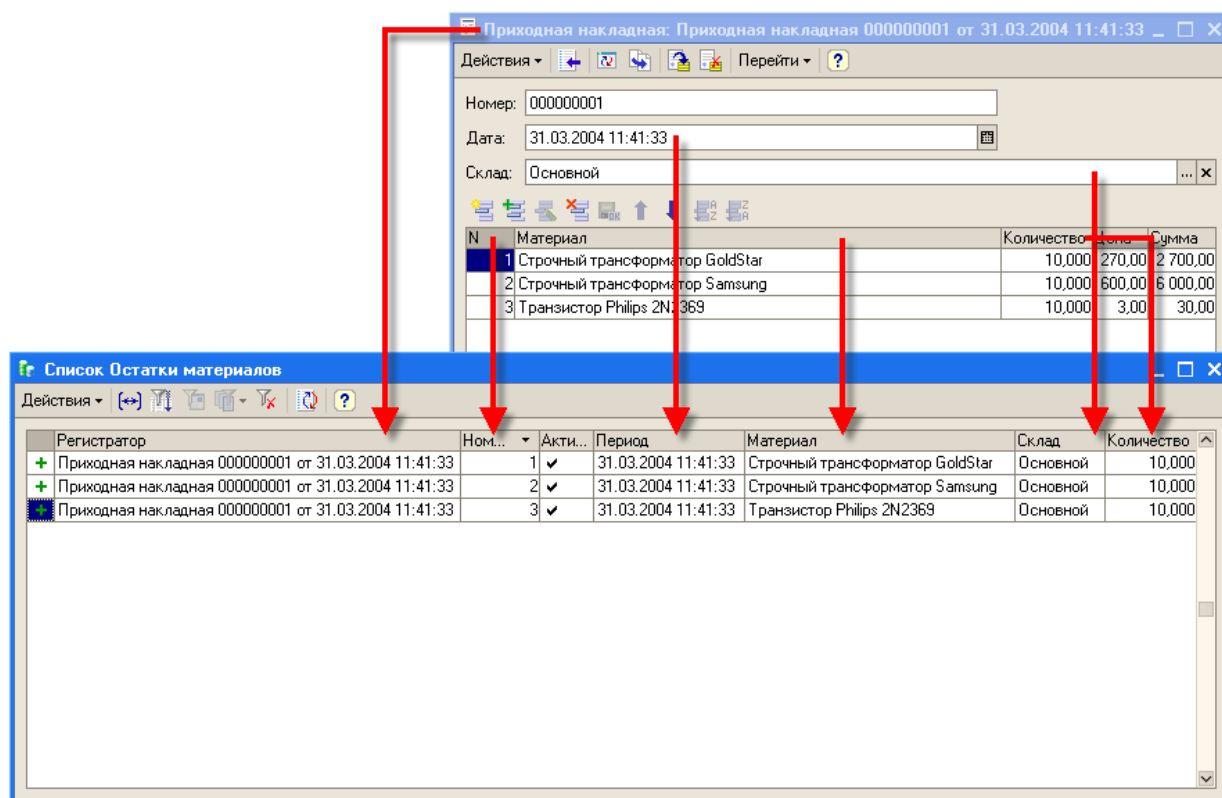


Рис. 6. Взаимодействие документа и регистра

4. Аналогичные действия проделайте и с документом Приходная накладная №2.

Лабораторная работа №14. Создание движений документа «ОказаниеУслуги»

Теперь аналогичным образом создайте движения документа ОказаниеУслуги.

При использовании конструктора будьте внимательны и обратите внимание на то, что документ Оказание услуги должен расходовать материалы. Поэтому перед тем, как нажать кнопку ОК, убедитесь, что выбран правильный тип движения регистров (нам нужен Расход).

Кроме того, при автоматическом заполнении поле Материал не заполнится автоматически.

Если мы оставим это так, как есть, то в регистре накопления в строках с типом Движение регистра – расход имя номенклатуры фиксироваться не будет. Чтобы избежать этого, надо выбрать поле Материал и в поле Реквизиты документа дважды щелкнуть по строке ТекСтрокаПереченьНомеклатуры.Номенклатура. Таким образом, имя номенклатуры для строк регистра накопления будет выбираться из табличной части документа.

Запустите отладку и создайте документ оказания услуги, который будет расходовать один транзистор Philips за 3 рубля.

Проведите документ оказания услуги и убедитесь, что в регистре накопления он создал верные движения.

Сформированные таким образом движения этого документа будут не совсем правильны.

Дело в том, что в документе ОказаниеУслуги, в отличие от документа ПриходнаяНакладная, могут содержаться не только расходующие материалы, но и услуги. Поэтому в регистр ОстаткиМатериалов будут попадать записи и о расходующих услугах, что неправильно.

Пока мы ничего не будем делать с движениями, которые сформировал конструктор, но как только познакомимся с перечислениями, мы внесем в обработчик проведения необходимые изменения.

Контрольные вопросы

- ☐ для чего предназначен объект конфигурации Регистр накопления
- ☐ почему следует использовать регистры, хотя необходимая информация содержится в других объектах
- ☐ для чего нужны измерения регистра, ресурсы и реквизиты
- ☐ что такое движения регистра и что такое регистратор
- ☐ как создать новый регистр накопления и описать его структуру
- ☐ как создать движения документа с помощью конструктора движений
- ☐ как средствами встроенного языка обойти табличную часть документа и обратиться к ее данным
- ☐ как средствами встроенного языка сформировать и записать движения документа в регистр накопления

Лабораторная работа №15. Создание отчетов

Цель:

1. Изучить назначение объекта конфигурации Отчет.
2. Научиться создавать отчет, который будет показывать движения и остатки материалов на предприятии.

Объект конфигурации Отчет

Объект конфигурации *Отчет* является прикладным и предназначен для описания алгоритмов, при помощи которых пользователь сможет получать необходимые ему выходные данные. Алгоритм формирования выходных данных описывается при помощи визуальных средств или с использованием встроенного языка. В реальной жизни объектам конфигурации Отчет соответствуют всевозможные таблицы выходных данных, сводных данных, диаграммы и пр.

Создание отчета «Материалы»

Приступим к созданию отчета, который будет показывать приход, расход и остатки материалов. Более глубоко работа с конструктором выходной формы будет рассмотрена в следующих лабораторных работах. Для создания отчета необходимо выполнить следующее:

1. Создайте новый объект конфигурации Отчет и назовите его Материалы.
2. Откройте окно редактирования объекта конфигурации Отчет Материалы и создайте схему компоновки данных. Для этого нажмем кнопку Открыть схему компоновки данных (рис. 1).

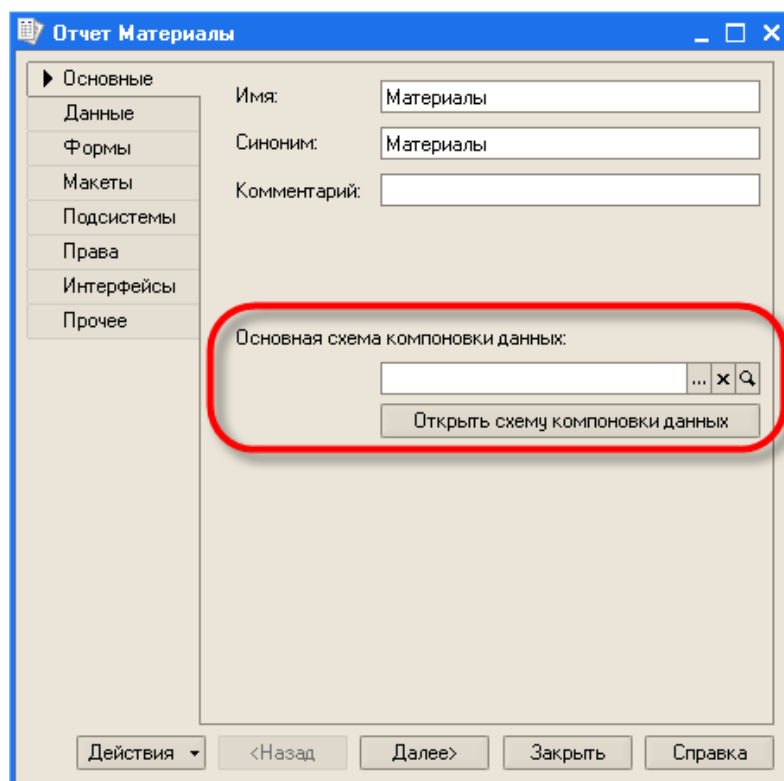


Рис. 1. Окно редактирования объекта конфигурации Отчет

3. Так как у отчета, который мы создаем, еще не существует схемы компоновки данных, платформа предложит создать новую схему. Схема компоновки данных с точки зрения конфигурации является макетом, поэтому будет открыт конструктор макета, предлагающий

выбрать единственный тип макета – Схема компоновки данных (рис. 2). Нажмем кнопку Готово.

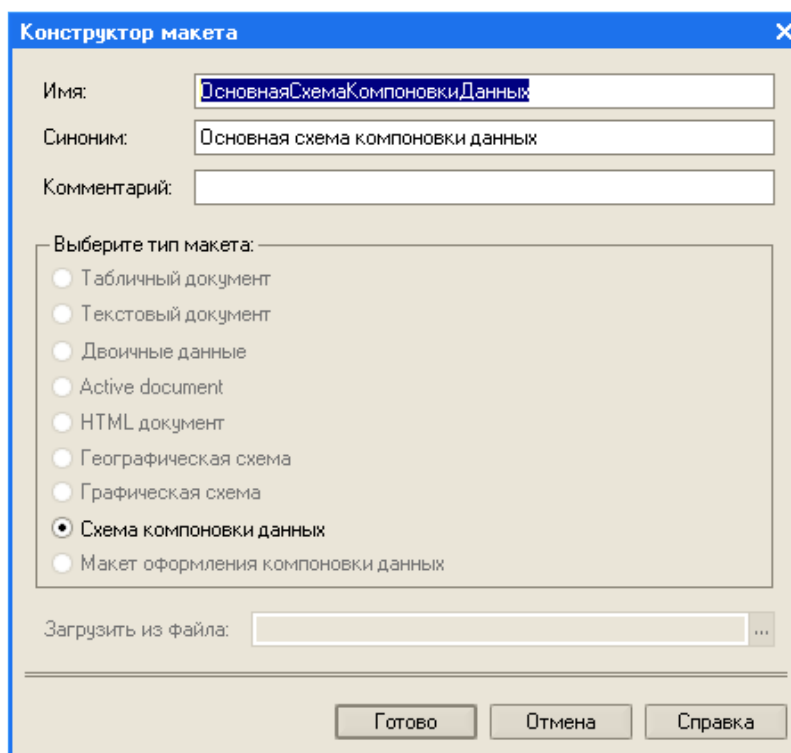


Рис. 2. Конструктор макета

Платформа создаст новый макет, содержащий схему компоновки данных, и сразу же откроет конструктор схемы компоновки данных.

Конструктор обладает большим количеством возможностей для визуального проектирования отчетов, но мы сейчас воспользуемся только самыми простыми его возможностями и определим те данные, которые хотим видеть в результате работы нашего отчета.

4. Добавим новый набор данных - запрос (рис. 3):

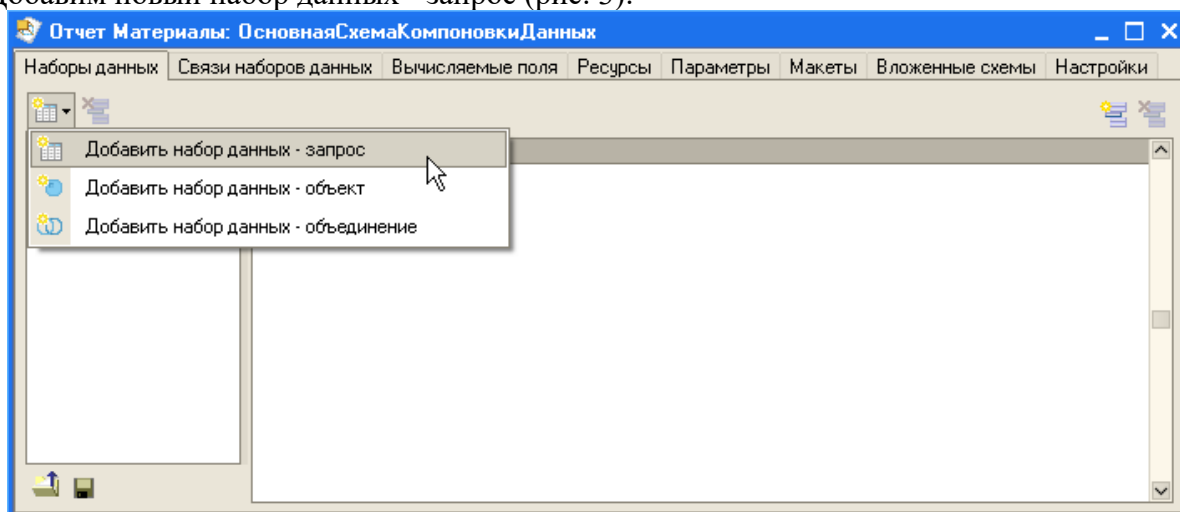


Рис 3. Добавление набора данных в конструкторе схемы компоновки

5. Для того чтобы создать текст запроса, запустите конструктор запроса – кнопка Конструктор запроса (рис. 4):

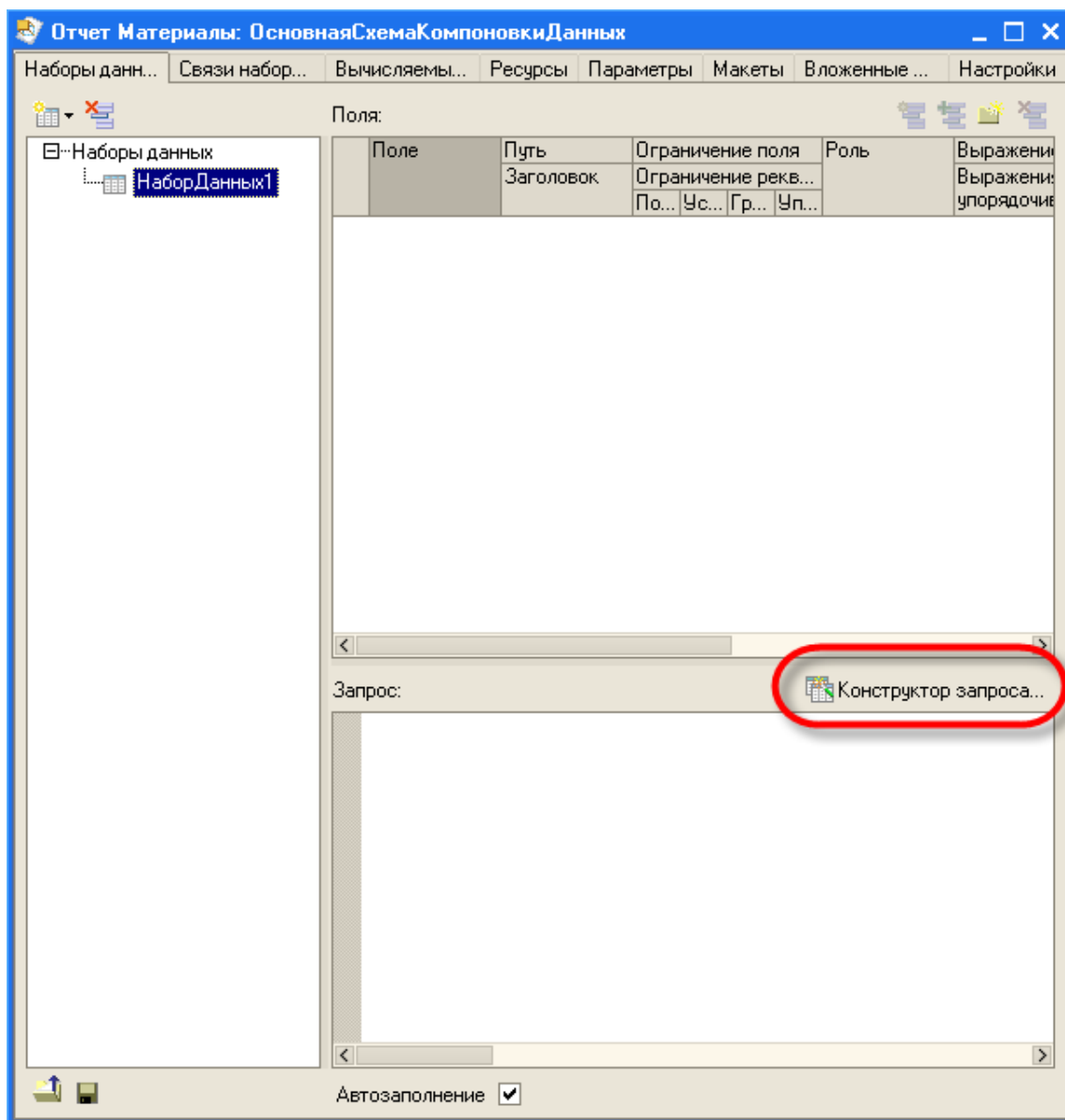


Рис. 4. Вызов конструктора запроса из конструктора схемы компоновки данных

В списке База данных представлены таблицы запросов. На основе их данных мы имеем возможность построить отчет. Если раскрыть ветку РегистрыНакопления, то мы увидим, что кроме таблицы регистра ОстаткиМатериалов в этой ветке присутствуют еще несколько *виртуальных таблиц*, которые формирует система (рис. 5).

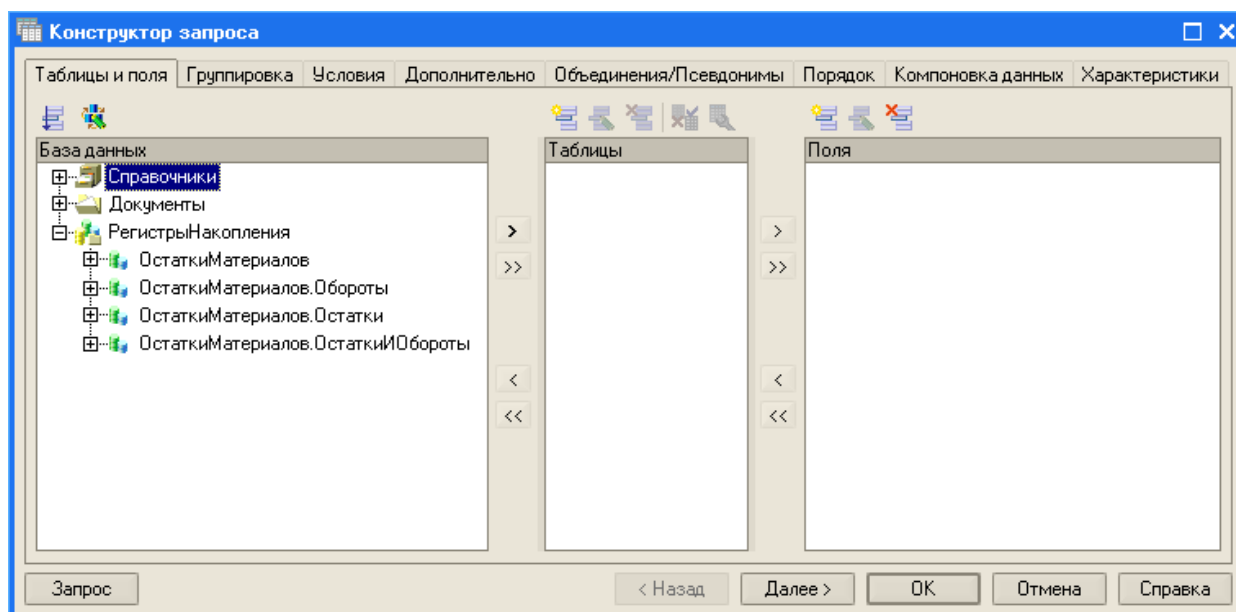


Рис. 5. Кроме таблицы регистра «ОстаткиМатериалов» присутствуют еще несколько виртуальных таблиц...

6. Раскройте виртуальную таблицу ОстаткиМатериалов.ОстаткиИОбороты, поскольку мы хотим видеть как остатки материалов, так и информацию об их поступлении и расходе (рис. 6).

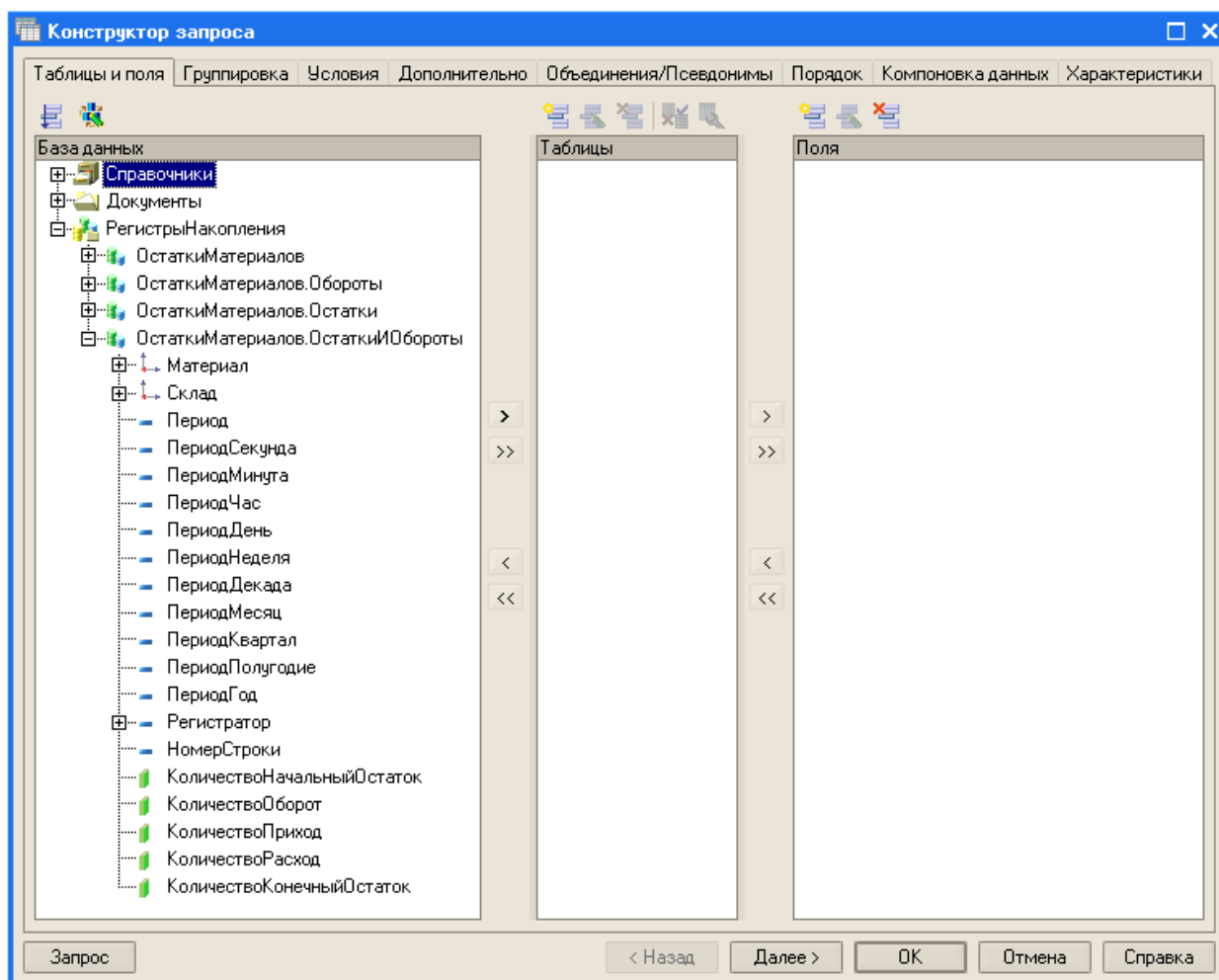


Рис. 6. Таблица «ОстаткиМатериалов.ОстаткиИОбороты»

Как вы видите, эта таблица содержит материал, склад и кроме этого начальные и конечные остатки, а также значения прихода, расхода и оборотов для всех ресурсов регистра ОстаткиМатериалов.

7. Начните выбирать поля таблицы двойным щелчком мыши в следующем порядке: Склад, Материал, КоличествоНачальныйОстаток, КоличествоПриход, КоличествоРасход и КоличествоКонечныйОстаток.

В результате окно Поля должно быть заполнено следующим образом (рис. 7):

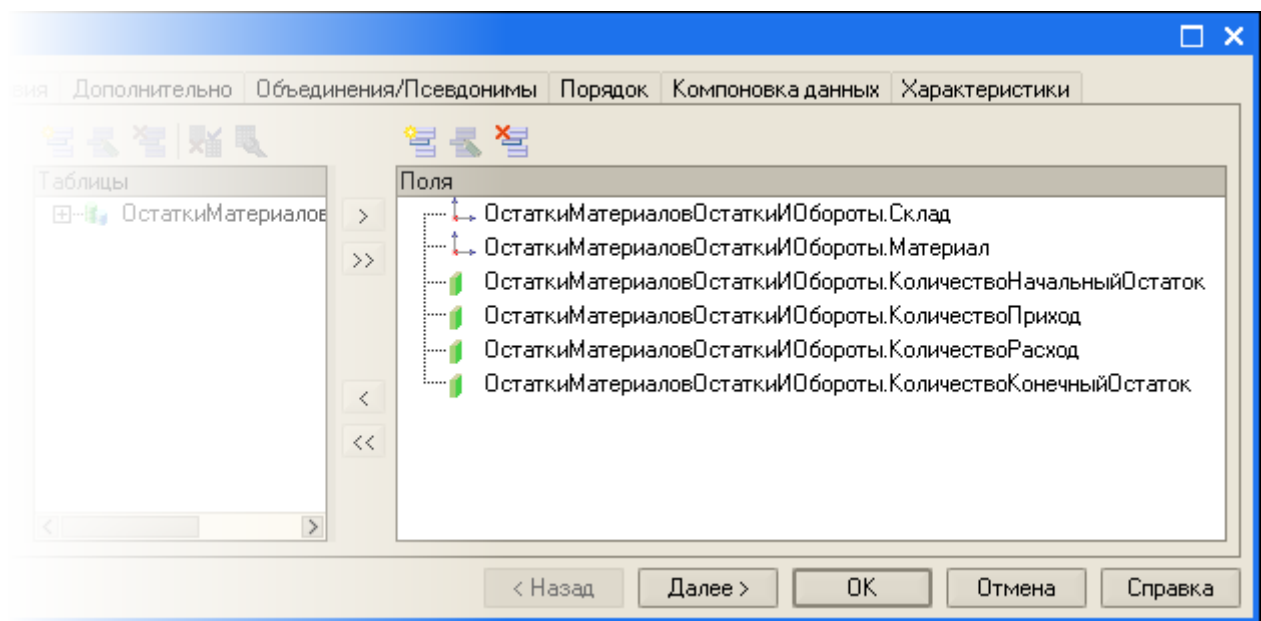


Рис. 5.7. Выбранные поля

8. После этого на закладке Группировка укажите групповое поле Склад:

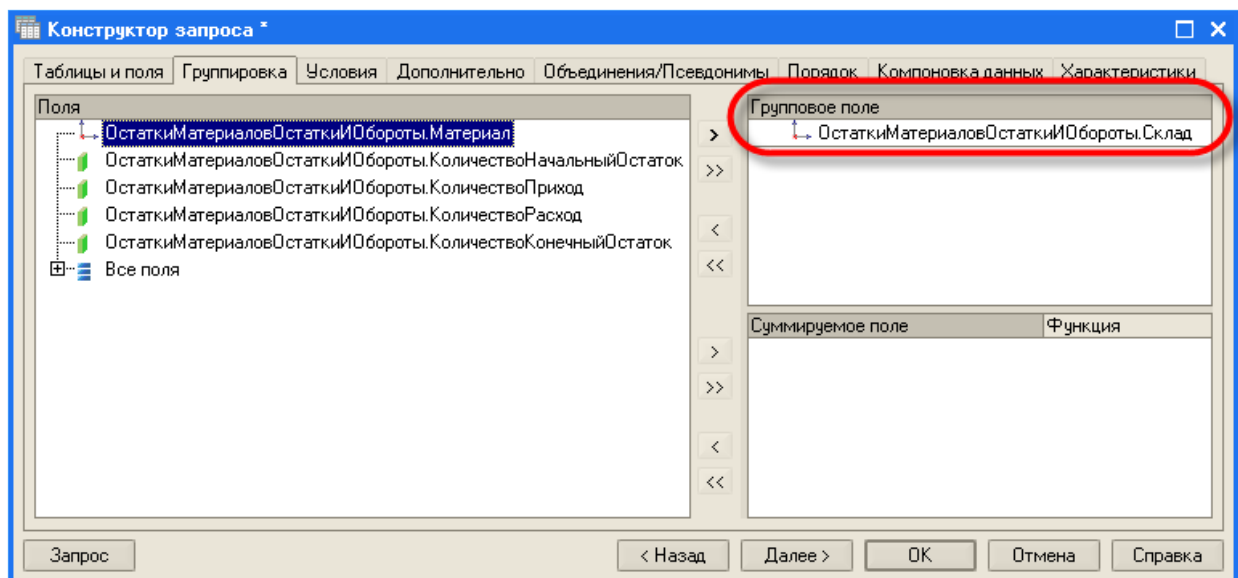
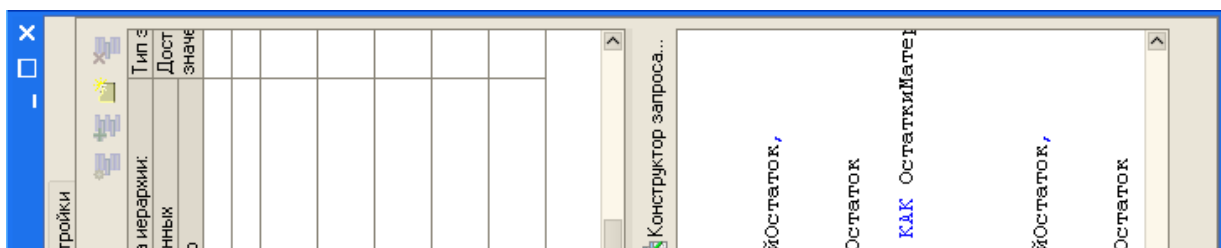


Рис. 5.8. Групповое поле «Склад»

9. Нажмите ОК и вернитесь в конструктор схемы компоновки данных (рис. 9).



Текст запроса, который был создан с помощью конструктора, платформа поместит в поле Запрос. Это поле представляет собой текстовый редактор, в котором можно вручную отредактировать существующий запрос. Кроме того, можно снова вызвать конструктор запроса и отредактировать запрос при помощи него.

Обратите внимание на список полей системы компоновки данных, который платформа заполнила в верхней части конструктора. В нем отображаются поля, которые доступны у текущего набора данных. В нашем случае система 1С:Предприятие заполнила данный список автоматически, из текста запроса, и нет необходимости в его ручной настройке.

10. Перейдите на закладку Настройки и добавьте группировку (контекстное меню – Новая группировка). При этом не указывайте поле группировки, а просто нажмите ОК. В отчет будут выводиться *детальные записи* из информационной базы – записи, получаемые в результате выполнения запроса без итогов (рис. 10).

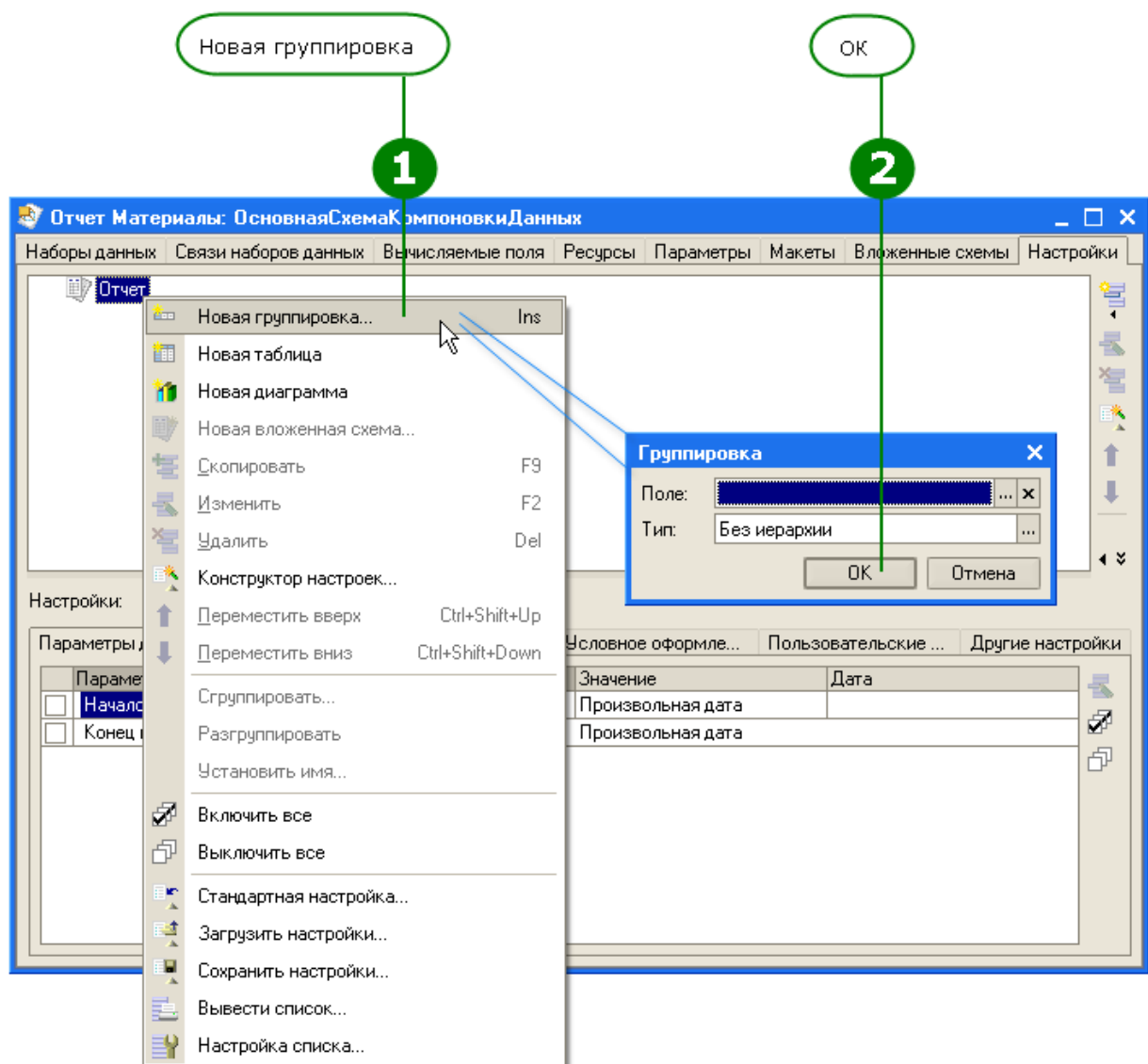


Рис. 5.10. Добавим новую группировку

11. Настройте поля, которые будут выводиться в результат отчета. Для этого перейдите на закладку Выбранные поля и перенесите мышью из списка доступных полей:

- Склад;
- Материал;
- КоличествоНачальныйОстаток;
- КоличествоПриход;
- КоличествоРасход;
- КоличествоКонечныйОстаток.

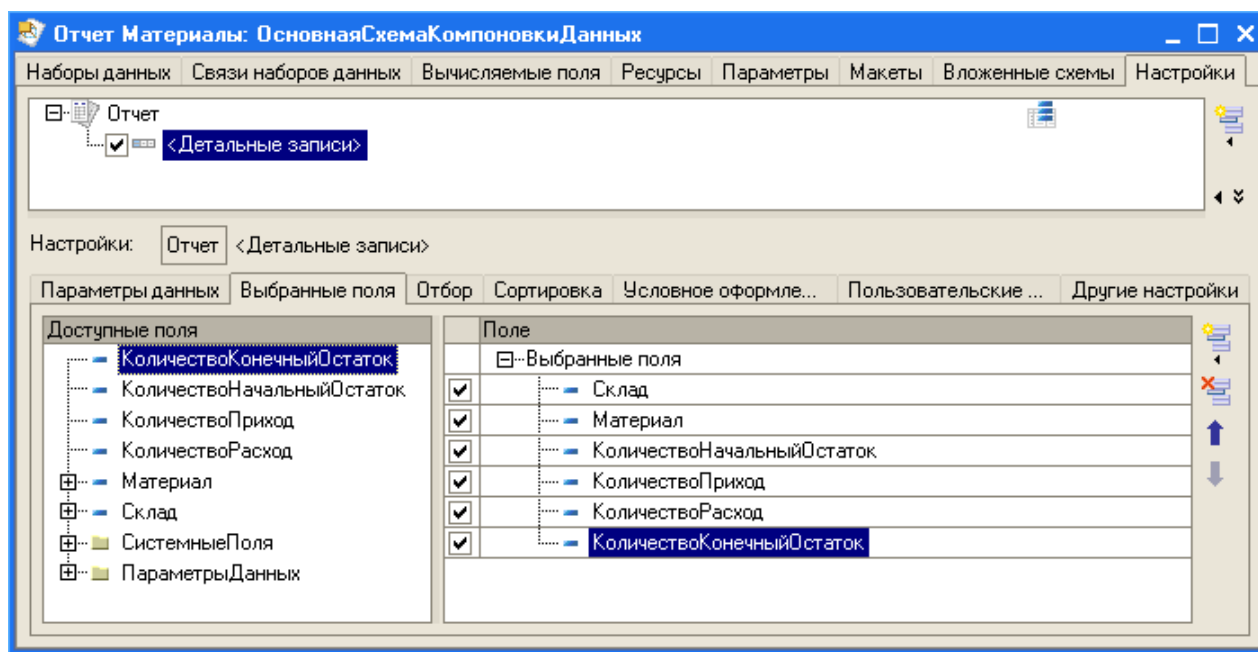


Рис. 5.11. Настройки отчета

12. Запустите 1С:Предприятие в режиме отладки и проверьте, как работает отчет. Выполните команду Операции > Отчет... > Материалы. Система автоматически сформирует форму отчета и форму настроек для него.

13. Нажмите Сформировать. Как видите, наш отчет вполне презентабелен и полностью отражает движение материалов, произошедшее в нашей организации (рис. 12):

Склад	Материал	Количество начальный остаток	Количество приход	Количество расход	Количество конечный остаток
Основной	Строчный трансформатор Samsung		10		10
Основной	Строчный трансформатор GoldStar		10		10
Основной	Транзистор Philips 2N2369		10	1	9
Основной	Шланг резиновый		5		5
Основной	Кабель электрический		5		5

Рис. 12. Отчет «Материалы»

Контрольные вопросы

- ❑ для чего предназначен объект конфигурации Отчет
- ❑ как создать отчет с помощью конструктора схемы компоновки данных

Лабораторная работа №16. Объект конфигурации Макет

- Цель:** 1. Узнать назначение объекта конфигурации Макет.
2. Научиться создавать макет документа.

Объект конфигурации *Макет* предназначен для хранения различных форм представления данных, которые могут потребоваться каким-либо объектам конфигурации или всему прикладному решению в целом. Макет может содержать табличный или текстовый документ, двоичные данные, HTML-документ или Active Document, графическую или географическую схему, схему компоновки данных или макет оформления схемы компоновки данных. Макеты могут существовать как сами по себе (общие макеты), так и быть подчинены какому-либо объекту конфигурации.

Одно из предназначений макета, подчиненного объекту конфигурации и содержащего табличный документ, – создание печатной формы этого объекта.

Создание печатной формы заключается в конструировании ее составных частей – именованных областей, из которых затем «собирается» готовая печатная форма. Порядок заполнения областей данными и порядок вывода их в итоговую форму описывается при помощи встроенного языка.

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

Помимо создания макета «вручную», конфигуратор предоставляет разработчику возможность воспользоваться специальным инструментом – *конструктором печати*, который берет на себя большинство рутинной работы по созданию макета.

Создание макета документа «Оказание услуги»

Для создания макета документа «Оказание услуги» необходимо выполнить следующее:

1. Откройте в конфигураторе окно редактирования объекта конфигурации Документ ОказаниеУслуги.
2. Перейдите на закладку Макеты и запустите конструктор печати (рис. 1):

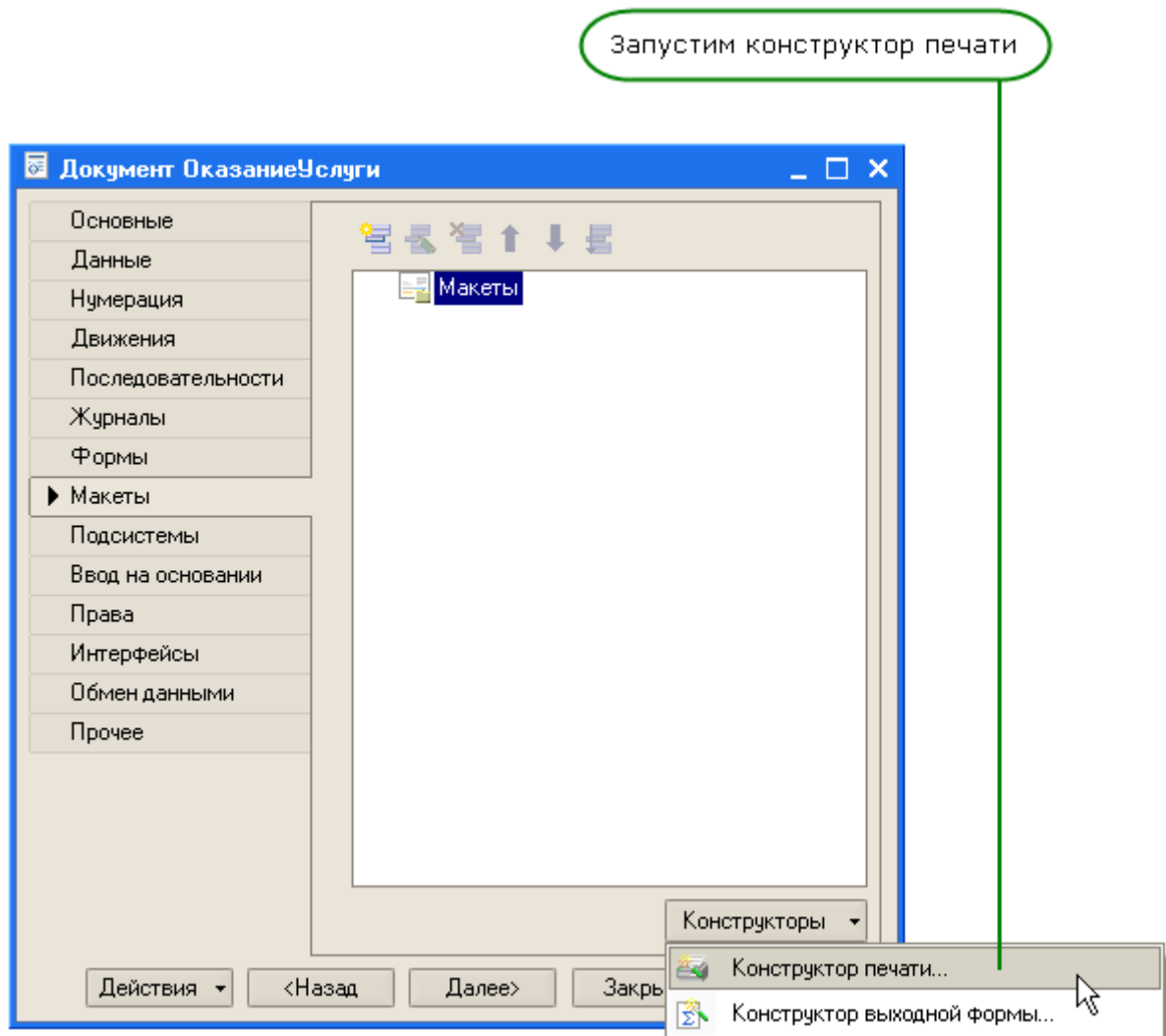


Рис. .1. Запустим конструктор печати...

- А) на первом шаге укажите, что новая процедура, которая будет создана конструктором для формирования печатной формы документа, будет располагаться в модуле формы;
- Б) на втором шаге нажатием двойной стрелки определите, что все реквизиты документа будут отображены в шапке печатной формы;
- В) на третьем шаге точно так же определите, что все реквизиты табличной части документа будут отображены в печатной форме;
- Г) на четвертом шаге конструктор предложит сформировать подвал (нижнюю часть) печатной формы, при этом ничего указывать не нужно. (подвал в данном случае использовать не будем) и перейдите к пятому шагу;
- Д) далее укажите, что конструктор должен вставить новую кнопку в форму документа для вызова процедуры формирования печатной формы, и нажмем ОК.
- В конфигураторе откроется форма документа и его макет (рис. 2):

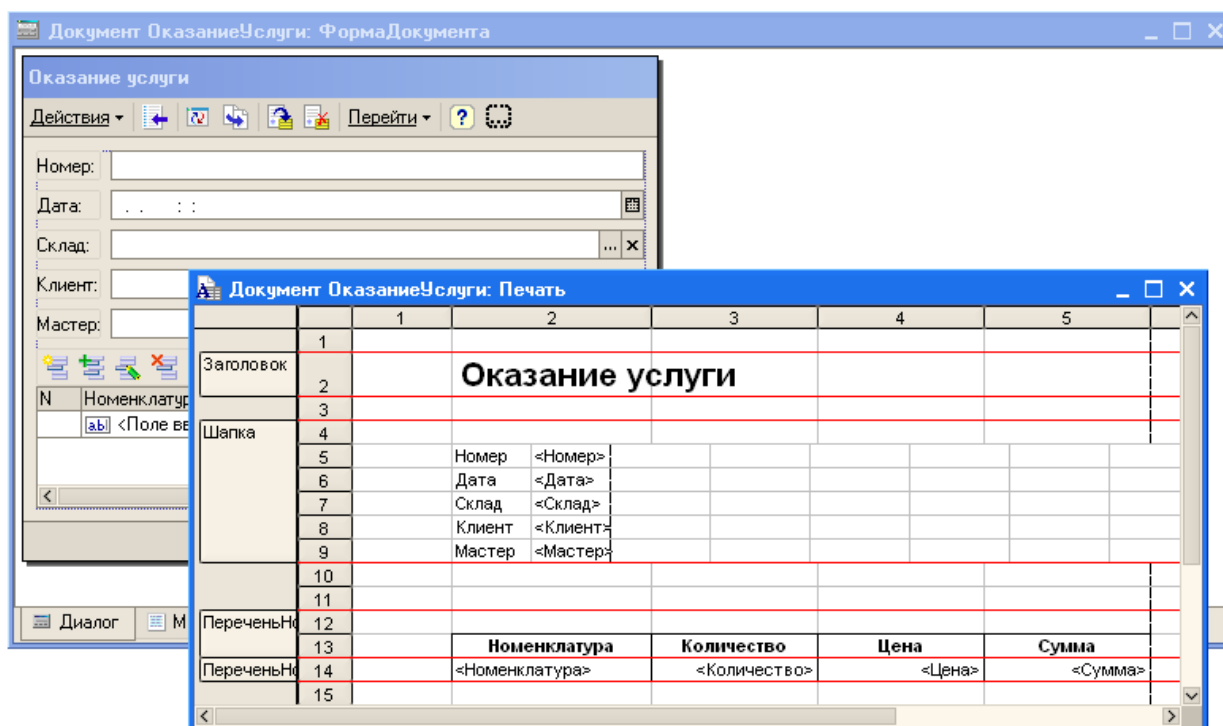


Рис. 2. Макет документа ОказаниеУслуги

Проверьте макет в работе.

3. Запустите 1С:Предприятие в режиме отладки и откройте документ Оказание услуги №1. Обратите внимание, что в правом нижнем углу документа появилась новая кнопка Печать (рис. 3).

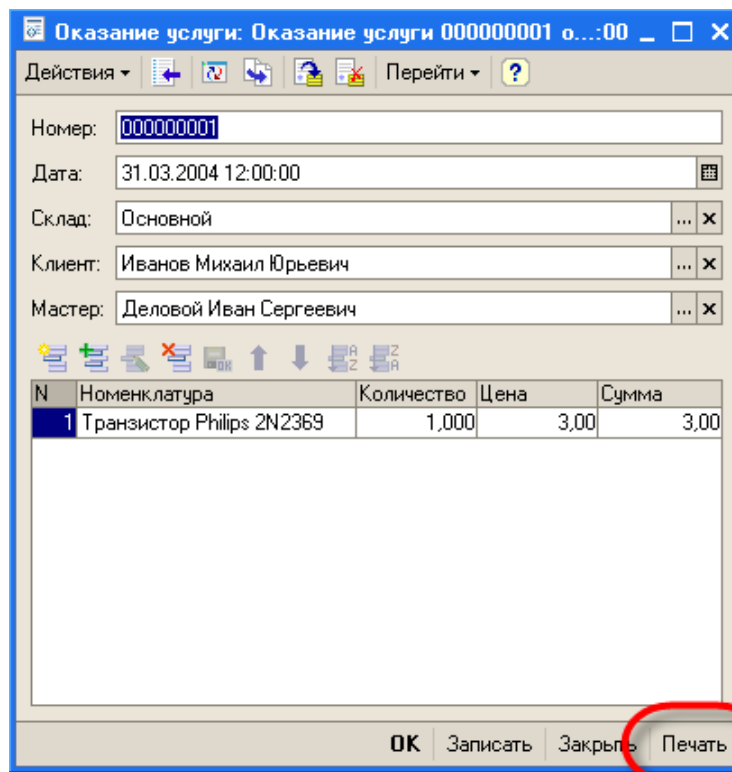


Рис. 3. Появилась новая кнопка Печать

Эту кнопку добавил конструктор.

4. Нажмите на нее и увидите печатную форму документа (рис. 4).

Номенклатура	Количество	Цена	Сумма
Транзистор Philips 2N2369	1	3	3

Рис. 4. Печатная форма документа

Как видно, конструктор сформировал вполне подходящую печатную форму для документа.

Единственное, чего не хватает в данной форме, – это итоговой суммы документа.

Далее познакомимся с тем, как можно редактировать макеты и формы объектов конфигурации, и добавим итоговую сумму к форме и макету документа ОказаниеУслуги.

Редактирование макета документа «Оказание услуги»

Для внесения указанных дополнений необходимо выполнить следующие действия:

1. Откройте конфигуратор, раскройте дерево документа ОказаниеУслуги и дважды щелкнем на макете Печать. Как видите, макет документа состоит из именованных областей, которые в определенном порядке выводятся на печать.

2. Добавьте новую область для вывода итоговой суммы документа (рис. 5). Для этого необходимо выделить мышью две пустые строки и выполним команду Таблица > Имена > Назначить имя...

	1	2	3	4	5	6
Заголовок	1	Оказание услуги				
Шапка	2					
	3					
	4					
	5	Номер	<Номер>			
	6	Дата	<Дата>			
Перечень	7	Склад	<Склад>			
	8	Клиент	<Клиент>			
	9	Мастер	<Мастер>			
Итого	10					
	11					
	12					
	13	Номенклатура	Количество	Цена	Сумма	
	14	<Номенклатура>	<Количество>	<Цена>	<Сумма>	
	15					
	16					
	17					
	18					
	19					

Рис. 5. Выделим мышью две строки и зададим имя области

3. Назовите область Всего. В созданной области, в колонке Цена, напишите Всего, а в колонке Сумма напишите ВсегоПоДокументу.

4. Вызвав палитру свойств для последней заполненной ячейки, укажите, что в этой ячейке будет находиться не текст, а параметр (рис. 6):

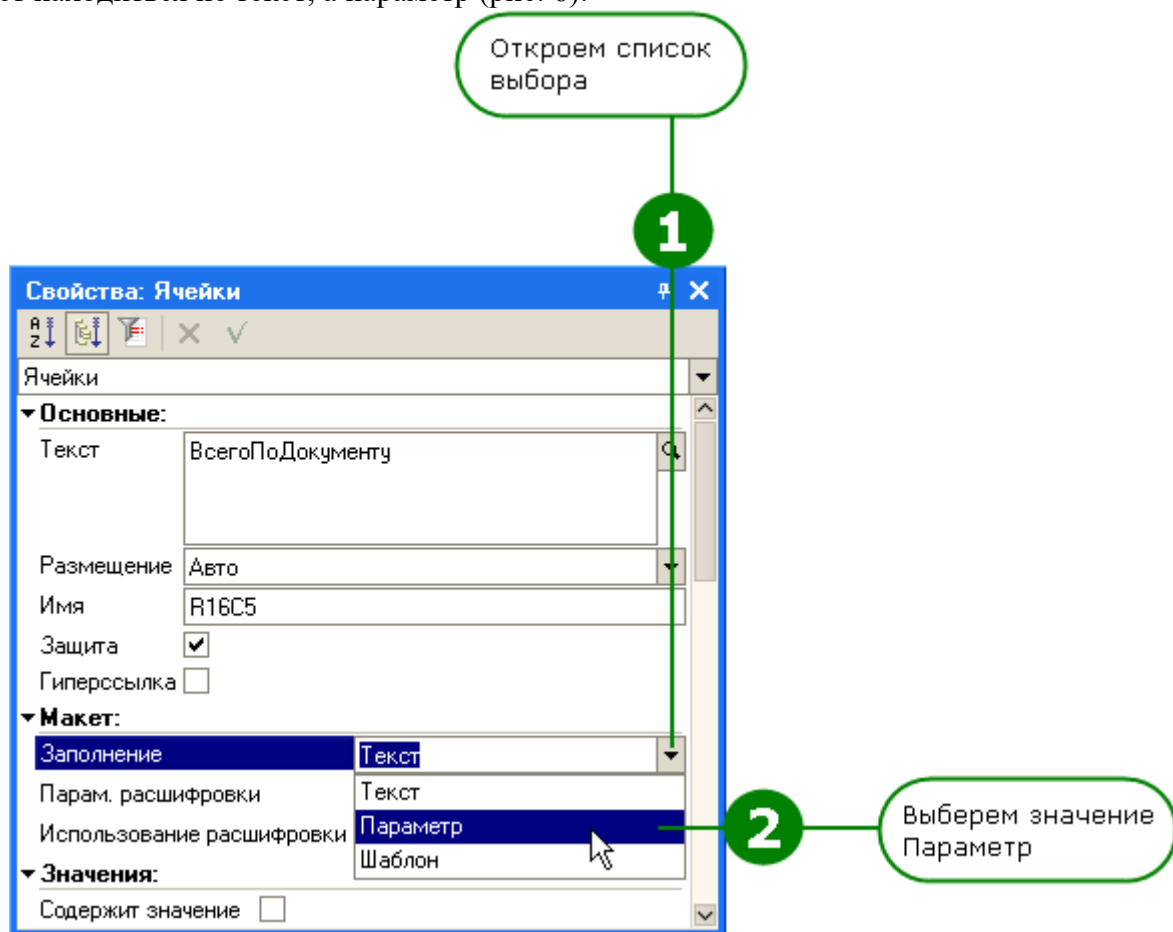


Рис. 6. Укажем, что в ячейке должен находиться не текст, а параметр...

Здесь следует сказать о том, что каждая ячейка редактируемого табличного документа может содержать либо текст, либо некоторый параметр, либо шаблон.

Текст, содержащийся в ячейке, будет показан на экране.

Параметр будет заменен некоторым значением, которое может быть присвоено ему средствами встроенного языка. Текст, содержащийся в ячейке, является именем этого параметра.

Шаблон представляет собой текстовую строку, в определенные места которой будут вставлены значения параметров.

Поэтому, указав для ячейки в качестве заполнения Параметр, мы определили параметр области с именем ВсегоПоДокументу, которому присвоим нужное нам значение при формировании печатной формы.

5. Теперь откройте модуль формы документа ОказаниеУслуги - ФормаДокумента.

6. Найдите в нем процедуру Печать и после цикла Для Каждого ТекСтрокатчНоменклатура Из тчНоменклатура Цикл добавьте в нее следующие строки (листинг 1):

Листинг 1. Печать формы документа

```
Область = Макет.ПолучитьОбласть ("Всего");
```

```
Область.Параметры.ВсегоПоДокументу = ПереченьНоменклатуры.Итог ("Сумма");
```

```
ТабДок.Вывести (Область);
```

Смысл добавленного фрагмента прост: мы получаем новую область Всего (ту, которую мы только что добавили к макету). Затем параметру ВсегоПоДокументу, находящемуся в этой

области, мы присваиваем значение суммы табличной части документа по колонке Сумма (обращение к табличной части документа выполняется по имени – ПереченьНоменклатуры). В заключение мы выводим область в итоговый табличный документ, который будет показан на экране и распечатан пользователем.

Подобным образом, используя свойства ячеек макета и управляя порядком их вывода, разработчик имеет возможность создать печатную форму любого дизайна.

А теперь, для того чтобы наш документ ОказаниеУслуги, выглядел вполне законченным, добавим итоговую сумму по документу и на экранную форму, чтобы пользователь мог видеть ее в процессе заполнения табличной части документа.

Редактирование формы документа ОказаниеУслуги

1. Откройте в конфигураторе форму документа ОказаниеУслуги – ФормаДокумента.
2. Откройте палитру свойств для табличного поля, расположенного в форме, и установите свойство Подвал, которое определяет наличие подвала у элемента управления Табличное поле (рис. 7).

3. Затем откройте свойства колонки Цена и установите:
 - текст подвала – Всего;
 - горизонтальное положение в подвале – Прижать вправо,
 - в шрифте подвала измените начертание на Жирный.
4. После этого откройте свойства колонки Сумма и установите:
 - горизонтальное положение в подвале – Прижать вправо,
 - установим флаг Показывать итог в подвале,
 - в шрифте подвала изменим начертание на Жирный.
5. Запустите 1С:Предприятие в режиме отладки и посмотрите, как теперь выглядит форма документа Оказание услуги №1 (рис. 8):

N	Номенклатура	Количество	Цена	Сумма
1	Транзистор Philips 2N2369	1,000	3,00	3,00
Всего:				3,00

Рис. 8. Форма документа Оказаниеуслуги №1

Подобным образом, используя свойства элементов управления и задавая их значения, разработчик имеет возможность создать экранную форму любого дизайна и поведения.

Контрольные вопросы

- ❑ для чего предназначен объект конфигурации Макет
- ❑ что такое конструктор печати
- ❑ как создать макет с помощью конструктора печати
- ❑ как изменить табличный документ
- ❑ какая разница в заполнении ячейки табличного документа текстом, параметром и шаблоном
- ❑ как с помощью встроенного языка вывести в табличный документ новую область
- ❑ как изменить внешний вид и поведение элемента управления, расположенного в форме
- ❑ как отобразить сумму по колонке табличного поля

Лабораторная работа №17. Назначение периодического регистра сведений

Цель:

1. Узнать назначение, структуру объекта конфигурации периодический регистр сведений.
2. Научиться создавать периодический регистр сведений и использовать его данные средствами встроенного языка.

Рассмотрим документ ОказаниеУслуги, который предназначен для выбора услуги, которая оказывается, и указания цену.

Очевидно, что в ООО «На все руки мастер» существует перечень услуг, который определяет стоимость каждой услуги. Казалось бы, стоимость услуги является неотъемлемым свойством самой услуги, и поэтому ее следует добавить в качестве реквизита справочника

Номенклатура.

Однако стоимость услуг имеет особенность меняться со временем. И может сложиться такая ситуация, когда нам потребуется внести изменения или уточнения в один из ранее проведенных документов ОказаниеУслуги. В этом случае мы не сможем получить правильную стоимость услуги, поскольку в реквизите справочника будет храниться последнее введенное значение.

Кроме того, не исключено, что руководство ООО «На все руки мастер» пожелает видеть, как зависит прибыль предприятия от изменения стоимости оказываемых услуг. В этом случае просто необходимо будет иметь возможность анализировать изменение стоимости услуг во времени.

Поэтому для хранения стоимости услуг мы используем новый пока еще для нас объект – регистр сведений.

Объект конфигурации *Регистр сведений* является прикладным и предназначен для описания структуры хранения данных в разрезе нескольких измерений. На основе объекта конфигурации Регистр сведений платформа создает в базе данных информационную структуру, в которой может храниться произвольная информация, «привязанная» к набору измерений.

Принципиальное отличие регистра сведений от регистра накопления заключается в том, что каждое движение регистра сведений устанавливает новое значение ресурса, в то время как движение регистра накопления изменяет существующее значение ресурса. По этой причине регистр сведений может хранить любые данные (а не только числовые, как регистр накопления).

Следующей важной особенностью регистра сведений является его способность (при необходимости) хранить данные с привязкой ко времени. Благодаря этому регистр сведений может хранить не только актуальные значения данных, но и историю их изменения во времени. Регистр сведений, использующий привязку ко времени, называют обычно *периодическим регистром сведений*.

Периодичность регистра сведений можно определить одним из следующих значений:

- в пределах секунды;
- в пределах дня;
- в пределах месяца;
- в пределах квартала;
- в пределах года;
- в пределах регистратора (если установлен режим записи –Подчинение регистратору).

Периодический регистр сведений всегда содержит служебное поле Период, добавляемое системой автоматически. Оно имеет тип Дата и служит для указания факта принадлежности записи к какому-либо периоду. При записи данных в регистр платформа всегда приводит значение этого поля к началу периода, в который он попадает. Например, если в регистр сведений с периодичностью в пределах месяца записать данные, в которых период указан как 08.14.2004, то регистр сохранит эти данные со значением периода, равным 01.04.2004. Как и для других регистров, система контролирует уникальность записей для регистра сведений. Однако если для прочих регистров уникальным идентификатором записи является регистратор и номер строки, то для регистра сведений применяется другой принцип формирования ключевого значения.

Ключом записи, однозначно идентифицирующим запись, является в данном случае совокупность значений измерений регистра и периода (в случае, если регистр сведений периодический). Регистр сведений не может содержать несколько записей с одинаковыми ключами.

Если продолжать сравнение с регистром накопления, то можно сказать, что регистр сведений предоставляет больше свободы в редактировании хранимых данных. Наряду с возможностью использования в режиме подчинения регистратору (когда записи регистра сведений «привязаны» к документу-регистратору) регистр сведений может применяться и в независимом режиме, в котором пользователю предоставляется полная свобода интерактивной работы с данными регистра. Регистр сведений, не использующий подчинение регистратору, называют *независимым регистром сведений*.

Создание периодического регистра сведений Цены

Приступим к созданию периодического регистра сведений, который будет хранить развернутые во времени розничные цены материалов и стоимости услуг, оказываемых нашим ООО «На все руки мастер».

1. Откройте конфигуратор и создадим новый объект конфигурации Регистр сведений.
2. Назовите его Цены.
3. Установите периодичность этого регистра в пределах секунды.
4. Перейдем на закладку Данные и создайте измерение регистра Номенклатура с типом СправочникСсылка.Номенклатура.
5. Укажите, что это измерение будет *ведущим* (рис 1.).

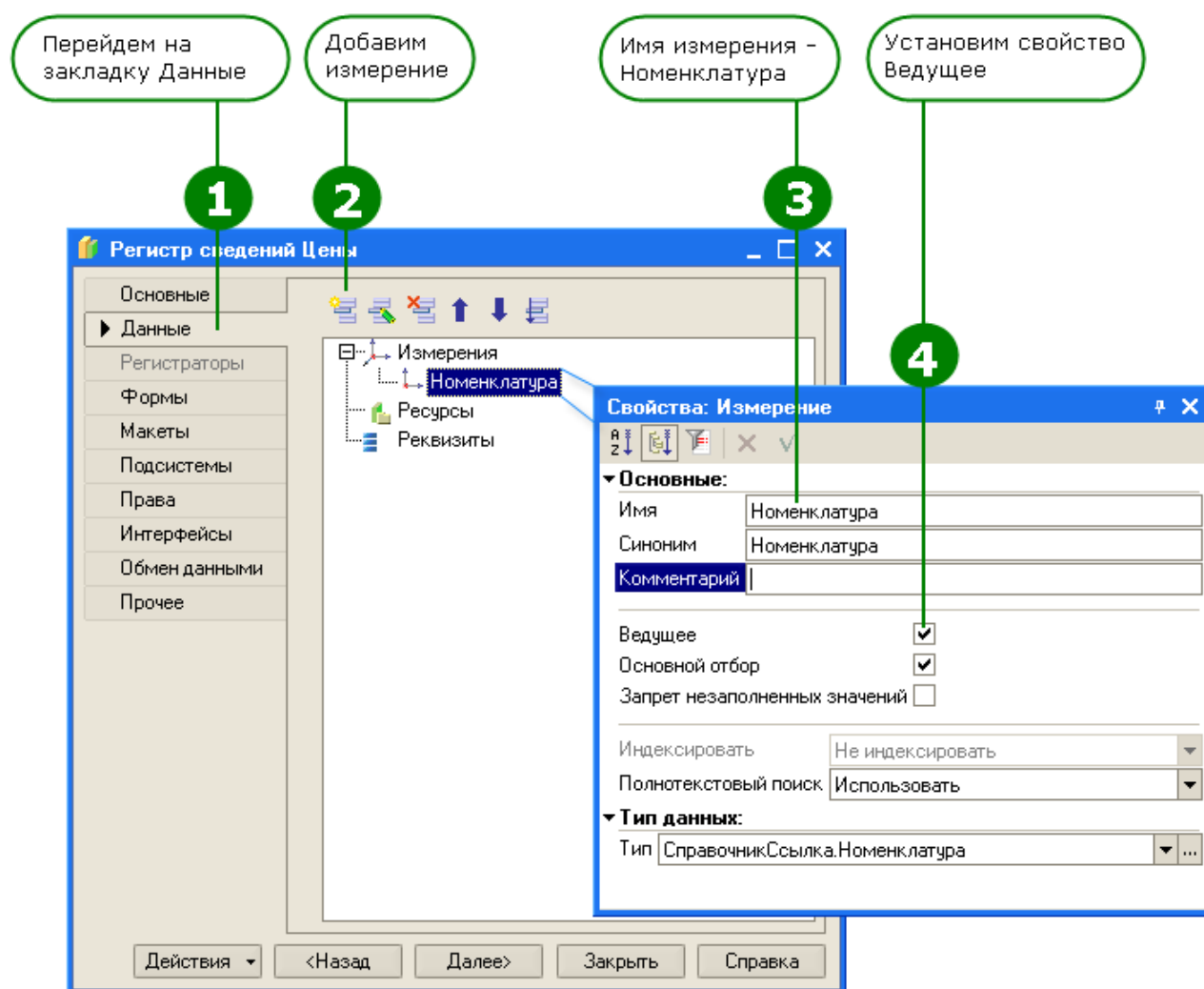


Рис. 1. Создадим измерение Номенклатура и укажем, что оно будет ведущим...

Свойство Ведущее имеет смысл использовать лишь тогда, когда измерение имеет тип ссылки на объект базы данных. Установка свойства Ведущее будет говорить о том, что запись регистра сведений представляет интерес, пока существует этот объект. При удалении объекта все записи регистра сведений по этому объекту тоже будут автоматически удалены. Кроме того, в форме списка справочника появляется кнопка командной панели Перейти. По ней возможен переход к записям регистра, отобранным по значению выбранного элемента справочника.

6. После этого создайте новый ресурс Цена, тип Число, длина 15, точность 2, неотрицательное.

Теперь запустите 1С:Предприятие в режиме отладки и посмотрите, как работает периодический регистр сведений Цены.

7. Задайте стоимость услуг ООО «На все руки мастер» следующим образом (рис. 2):

Период	Номенклатура	Цена
31.03.2004 0:00:00	Диагностика	200,00
31.03.2004 0:00:00	Ремонт отечественного телевизора	600,00
31.03.2004 0:00:00	Ремонт импортного телевизора	800,00
31.03.2004 0:00:00	Подключение воды	800,00
31.03.2004 0:00:00	Подключение электричества	800,00

Рис. 2. Регистр сведений Цены

8. После этого задайте розничные цены на материалы (рис. 3).

Период	Номенклатура	Цена
31.03.2004 0:00:00	Строчный трансформатор Samsung	900,00
31.03.2004 0:00:00	Строчный трансформатор GoldStar	400,00
31.03.2004 0:00:00	Транзистор Philips 2N2369	5,00
31.03.2004 0:00:00	Шланг резиновый	150,00
31.03.2004 0:00:00	Кабель электрический	30,00

Рис. 3. Регистр сведений Цены

Таким образом, в программе была создана возможность установки цен на услуги и материалы. Поскольку цены хранятся с привязкой к дате, мы можем заранее установить новые цены и быть уверенными в том, что новые цены вступят в действие не раньше указанной для них даты.

Создание функции РозничнаяЦена()

Теперь посмотрим, как можно использовать заданные цены в документе ОказаниеУслуги. Для этого создадим функцию, которая будет возвращать актуальную розничную цену номенклатуры.

1. Откройте конфигуратор, в ветке Общие > Общие модули создайте новый объект конфигурации Модуль и назовите его РаботаСоСправочниками.

2. Поместим в нем следующий текст (листинг 1.1):

Листинг 1.1. Функция РозничнаяЦена()

Функция РозничнаяЦена (АктуальнаяДата, ЭлементНоменклатуры) Экспорт

```
// Создать вспомогательный объект Отбор
Отбор = Новый Структура ("Номенклатура", ЭлементНоменклатуры);

// Получить актуальные значения ресурсов регистра
```

```
ЗначенияРесурсов = РегистрыСведений.Цены.ПолучитьПоследнее(АктуальнаяДата,  
Отбор);
```

```
Возврат ЗначенияРесурсов.Цена;
```

КонецФункции

Для получения розничной цены в функцию передается два параметра:

- АктуальнаяДата – параметр типа Дата определяет точку на оси времени, в которой нас интересует значение розничной цены.
- ЭлементНоменклатуры – ссылка на элемент справочника Номенклатура, для которого мы хотим получить розничную цену.

В теле процедуры мы сначала создаем вспомогательный объект Отбор. С его помощью определяем, что нас будут интересовать записи регистра, в которых измерение Номенклатура равно переданной в процедуру ссылке на элемент справочника.

Во второй строке мы обращаемся к менеджеру регистра сведений Цены (РегистрыСведений.Цены) и выполняем метод ПолучитьПоследнее(), который возвращает нам значения ресурсов наиболее поздней записи регистра, соответствующей передаваемой дате (АктуальнаяДата) и значениям измерений регистра (Отбор).

Значения ресурсов возвращаются в объекте Структура, поэтому в следующей строке мы получаем искомую нами розничную цену, просто указав имя нужного нам ресурса регистра через точку (ЗначенияРесурсов.Цена).

Теперь проверим, как работает эта функция.

Автоматическое заполнение цены в документе ОказаниеУслуги

Итак, задача, которая перед нами стоит, заключается в следующем. При создании документа ОказаниеУслуги необходимо обеспечить автоматическое заполнение поля Цена после того, как пользователь выберет услугу. Причем цена услуги должна определяться исходя из даты создаваемого документа.

1. Найдите в конфигураторе документ ОказаниеУслуги и откройте его форму ФормаДокумента.
2. Откройте свойства поля ввода, расположенного в колонке Номенклатура
3. Внизу списка найдите событие При изменении.
4. Нажмите на кнопку с лупой и в открывшейся заготовке обработчика события напишите следующий текст (листинг 1.2):

Листинг 1.2. Процедура ПереченьНоменклатурыНоменклатура ПриИзменении()

```
// Получить текущую строку табличной части  
СтрокаТабличнойЧасти = ЭлементыФормы.ПереченьНоменклатуры.ТекущиеДанные;  
  
// Установить цену  
СтрокаТабличнойЧасти.Цена =  
РаботаСоСправочниками.РозничнаяЦена (Дата, Элемент.Значение);  
  
// Пересчитать сумму строки  
РаботаСДокументами.РассчитатьСумму (СтрокаТабличнойЧасти);
```

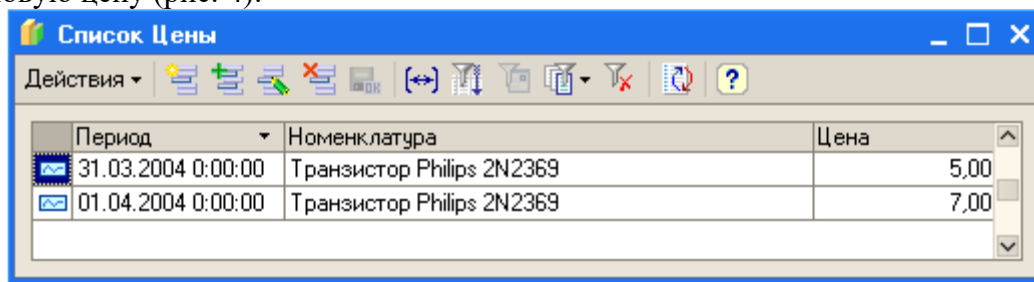
Прокомментируем содержимое обработчика.

Первая строка обработчика вам уже знакома: мы получаем текущую строку табличной части документа, так как она нам понадобится в дальнейшем.

Во второй мы устанавливаем полученную цену в документе, вызывая нашу процедуру РозничнаяЦена. Первым параметром мы передаем дату документа, на которую необходимо получить цену. Вторым параметром мы передаем ссылку, которую отображает элемент управления формой, вызвавший это событие (Элемент.Значение), то есть ссылку на элемент справочника Номенклатура.

В заключение мы вызываем нашу процедуру РассчитатьСумму из общего модуля РаботаСДокументами, чтобы она пересчитала итоговую сумму в строке нашего документа.

5. Проверьте, как теперь работает наш документ. Запустите 1С:Предприятие в режиме отладки и откройте регистр сведений Цены. Для транзистора Philips добавьте следующим числом новую цену (рис. 4):

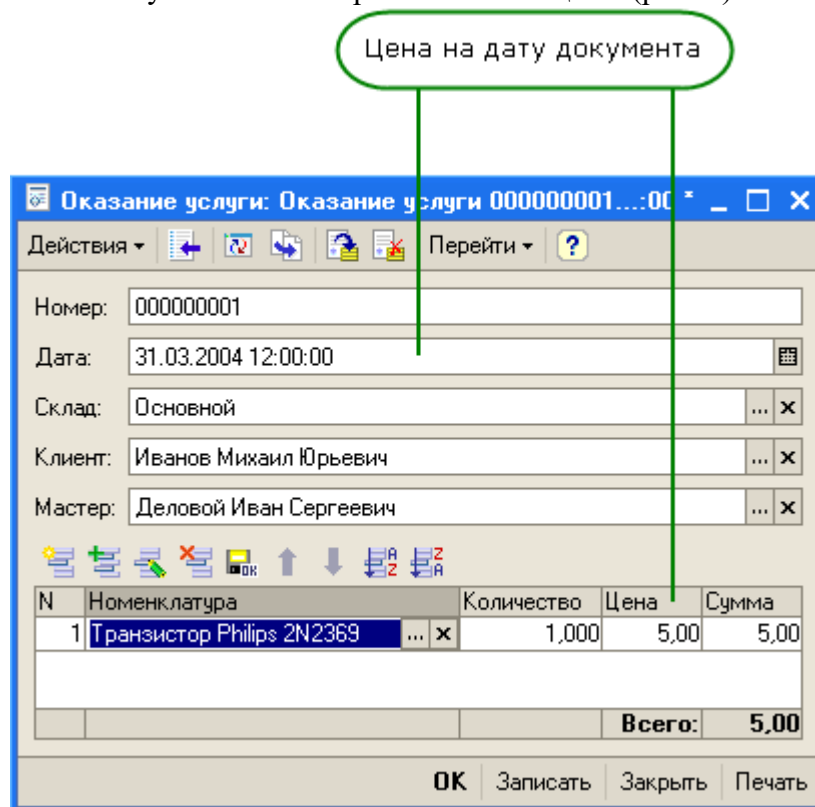


Период	Номенклатура	Цена
31.03.2004 0:00:00	Транзистор Philips 2N2369	5,00
01.04.2004 0:00:00	Транзистор Philips 2N2369	7,00

Рис. 4. Регистр сведений Цены

6. Теперь откройте документ Оказаниеуслуги №1. Как вы помните, этим документом мы как раз «израсходовали» один такой транзистор.

7. Установите дату документа равной той дате, когда было задано первое значение цены транзистора, и повторите выбор транзистора в колонке Номенклатура табличной части документа. Автоматически установится первое значение цены (рис. 5).



Цена на дату документа

N	Номенклатура	Количество	Цена	Сумма
1	Транзистор Philips 2N2369	1,000	5,00	5,00
Всего:				5,00

OK Записать Закрывать Печать

Рис. 5. Заполнение документа Оказание услуги

8. Теперь измените дату документа на следующий день и снова повторите выбор транзистора. Будет установлено новое значение цены (рис. 6).

Цена на дату документа

N	Номенклатура	Количество	Цена	Сумма
1	Транзистор Philips 2N2369	1,000	7,00	7,00
			Всего:	7,00

Рис. 6. Заполнение документа Оказание услуги

Таким образом, в документе появляется актуальная на момент создания документа цена услуги.

Контрольные вопросы

- ❑ для чего предназначен объект конфигурации Регистр сведений
- ❑ какими особенностями обладает объект конфигурации Регистр сведений
- ❑ в чем главные отличия регистра сведений от регистра накопления
- ❑ какие поля определяют ключ уникальности регистра накопления
- ❑ что такое периодический регистр сведений и что такое независимый регистр сведений
 - ❑ как создать периодический регистр сведений
 - ❑ что такое ведущее измерение регистра
 - ❑ как получить значения ресурсов наиболее поздних записей регистра средствами встроенного языка

Лабораторная работа №18. Объект конфигурации Перечисление

Цель:

1. Узнать назначение объекта конфигурации Перечисление.
2. Научиться работать с перечислением средствами встроенного языка.

До сих пор мы с вами не обращали внимания на то, что у нас нет никакого признака, по которому мы могли бы сказать, чем является конкретный элемент справочника Номенклатура: материалом или услугой. То, что все элементы справочника разложены у нас по некоторым группам, не может являться надежным критерием оценки: группы можно удалить, переименовать, сгруппировать элементы по другим принципам...

Поэтому требуется некоторый признак, позволяющий однозначно определять принадлежность элемента справочника к материалам или услугам, независимо от изменения иерархической структуры справочника. Для этого предназначен объект конфигурации Перечисление.

Объект конфигурации Перечисление является прикладным и предназначен для описания структуры хранения постоянных наборов значений, не изменяемых в процессе работы конфигурации. На основе объекта конфигурации Перечисление платформа создает в базе данных информационную структуру, в которой может храниться набор некоторых постоянных значений.

В реальной жизни этому объекту может соответствовать, например, перечисление вариантов указания цены (включая НДС, без НДС).

Набор всех возможных значений, которые содержит перечисление, задается при конфигурировании системы, и пользователь не может изменять их, удалять или добавлять новые.

Из этого следует важная особенность перечисления: *значения перечисления не «обезличены» для конфигурации, на них могут опираться алгоритмы работы программы.*

Реорганизация справочника «Номенклатура»

1. Откройте конфигуратор и создайте новый объект конфигурации Перечисление с именем ВидыНоменклатуры.
2. На закладке Данные добавьте два значения перечисления: Материал и Услуга (рис. 1).

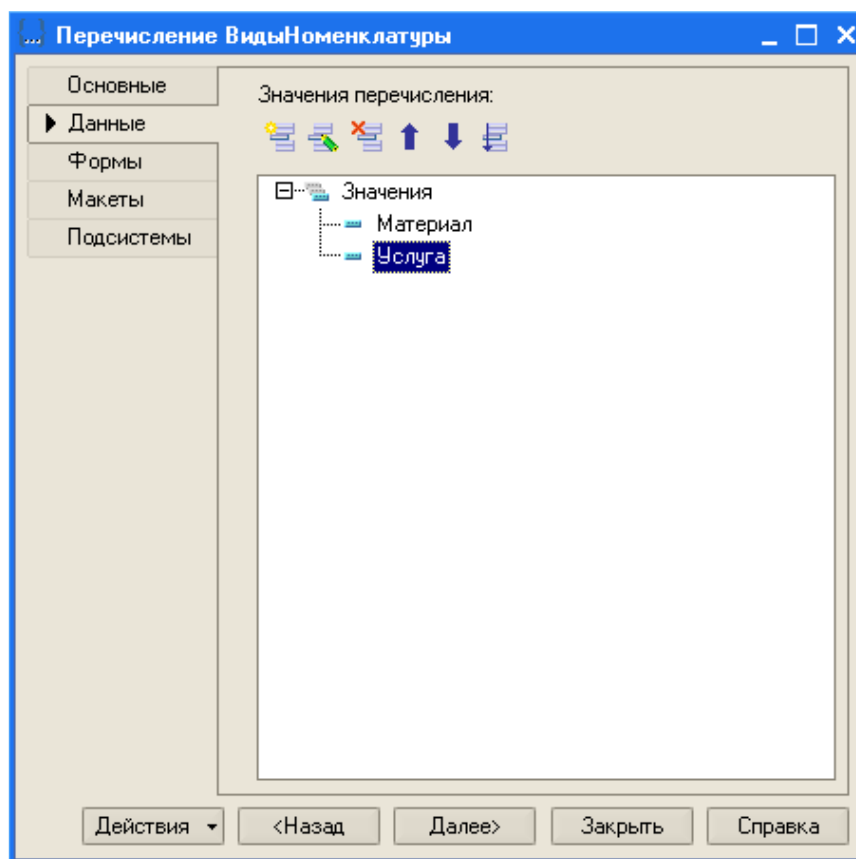


Рис. 1. Данные перечисления «ВидыНоменклатуры»

3. Затем добавьте в справочник Номенклатура новый реквизит ВидНоменклатуры с типом ПеречислениеСсылка.ВидыНоменклатуры (рис. 2).

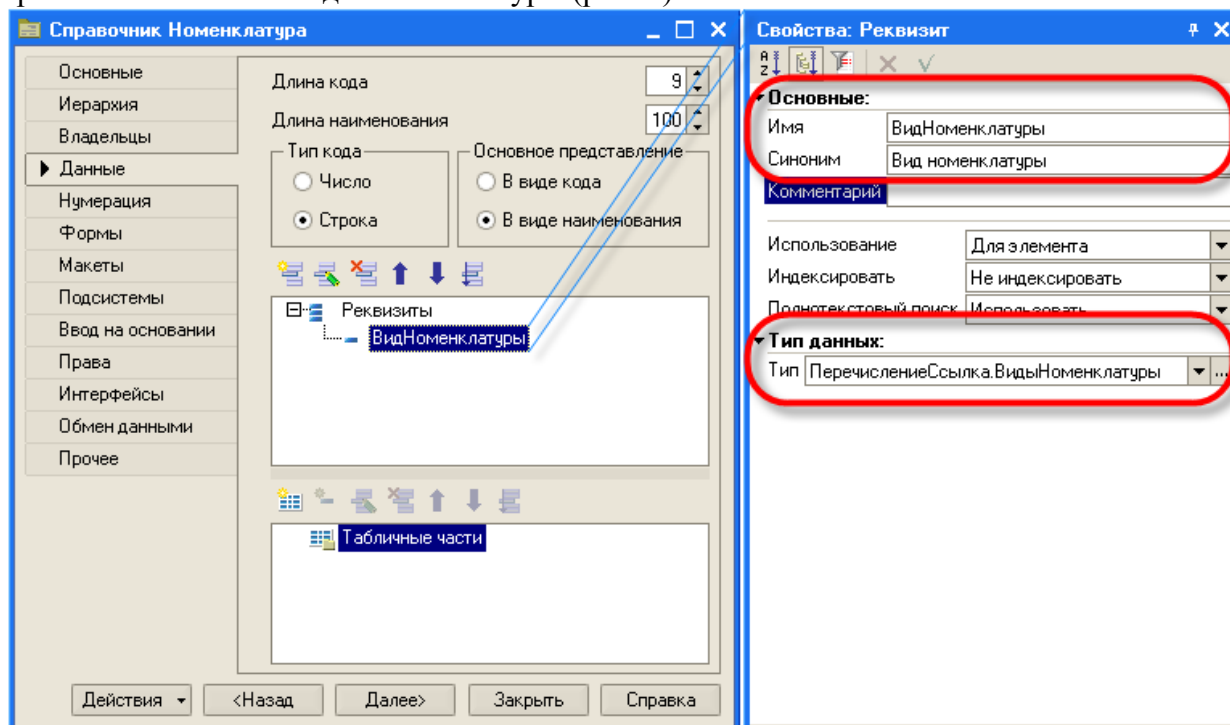


Рис. 2. Данные справочника «Номенклатура»

4. После этого запустите 1С:Предприятие в режиме отладки и задайте для каждого элемента справочника Номенклатура соответствующее значение реквизита ВидНоменклатуры (рис. 3).

Код	Наименование	Вид номенклатуры
000000003	Строчный трансформатор Samsung	Материал
000000004	Строчный трансформатор GoldStar	Материал
000000005	Транзистор Philips 2N2369	Материал
000000006	Шланг резиновый	Материал
000000007	Кабель электрический	Материал
000000008	Диагностика	Услуга
000000009	Ремонт отечественного телевизора	Услуга
000000010	Ремонт импортного телевизора	Услуга
000000011	Подключение воды	Услуга
000000012	Подключение электричества	Услуга

Рис. 3. Данные справочника «Номенклатура»

Теперь посмотрим, как можно применить новые данные, полученные благодаря использованию перечисления ВидНоменклатуры.

Изменение процедуры проведения документа «Оказание услуги»

На прошлых занятиях, когда создавались движения документа ОказаниеУслуги по регистру накопления ОстаткиМатериалов, было отмечено, что они не совсем правильные, поскольку в регистр будут попадать не только записи об израсходованных материалах, но и записи об оказанных услугах.

Доработаем документ таким образом, чтобы в регистре появлялись только записи, относящиеся к расходу материалов.

Скорректируем движения документа, исключив из обработки те строки табличной части, в которых находятся услуги. Для этого:

1. В обработчик события ОбработкаПроведения, расположенный в модуле документа ОказаниеУслуги, добавьте следующий текст (листинг 1). Текст следует добавить в начало цикла обхода табличной части документа после строки:
Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл

Листинг 1. Движения документа «ОказаниеУслуги»

```
Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры <>
    Перечисления.ВидыНоменклатуры.Материал Тогда
    Продолжить ;
КонецЕсли;
```

Добавленный текст исключает из выполнения операторов цикла те строки документа, в которых номенклатура не является материалом. К значению перечисления Материал мы обращаемся, используя менеджер перечисления ВидНоменклатуры (Перечисления.ВидыНоменклатуры) и указывая в качестве его свойства имя нужного нам значения перечисления.

Запустите 1С:Предприятие в режиме отладки и проверьте работу процедуры проведения документа ОказаниеУслуги.

2. Откройте документ ОказаниеУслуги №1 и внесите в него следующие изменения (обратите внимание, что изменен не только состав номенклатуры в табличной части, но и время документа) – рис. 4.

Оказание услуги: Оказание услуги 000000001....00

Действия

Номер: 000000001

Дата: 31.03.2004 14:00:00

Склад: Основной

Клиент: Иванов Михаил Юрьевич

Мастер: Деловой Иван Сергеевич

N	Номенклатура	Количество	Цена	Сумма
1	Подключение воды	1,000	800,00	800,00
2	Шланг резиновый	1,000	150,00	150,00
			Всего:	950,00

OK Записать Закрыть Печать

Рис. 4. Измененный документ «ОказаниеУслуги №1»

3. Перед тем как провести документ, откройте список регистра Остатки материалов, содержащий движения этого документа. Для этого выполните команду Перейти > Остатки материалов из командной панели списка документа.
4. Проведите документ и убедитесь, что в движения по регистру ОстаткиМатериалов включаются только строки, содержащие материалы (рис. 5).

Регистратор	Ном...	Акти...	Период	Материал	Склад	Количество
- Оказание услуги 000000001 от 31.03.2004 14:00:00	1	✓	31.03.2004 14:00:00	Шланг резиновый	Основной	1,000

Рис. 5. Записи регистра «ОстаткиМатериалов»

Контрольные вопросы

- ❑ для чего предназначен объект конфигурации Перечисление
- ❑ как создать новое перечисление
- ❑ как с помощью перечисления задать принадлежность элементов справочника к той или иной смысловой группе
- ❑ как обратиться к значению перечисления средствами встроенного языка

Лабораторная работа №19. Проведение документа по нескольким регистрам

Цель: научиться создавать процедуру проведения документов так, чтобы они записывали необходимые данные как в один, так и в другой регистр.

Зачем нужно проведение документа по нескольким регистрам?

До сих пор мы с вами учитывали только количественное движение материалов в ООО «На все руки мастер». Для этих целей мы создали регистр накопления ОстаткиМатериалов. Однако, как вы, наверное, догадываетесь, одного только количественного учета совершенно недостаточно для нужд нашего ООО.

Очевидно, что необходимо также знать, какие денежные средства были затрачены на приобретение тех или иных материалов, и каковы материальные запасы ООО «На все руки мастер» в денежном выражении.

После того, как мы начали автоматизировать наше предприятие, руководство ООО «На все руки мастер» высказало пожелание, чтобы весь суммовой учет материалов велся бы теперь по средней стоимости. То есть при закупке материалов они должны учитываться в ценах приобретения, а при расходе – по средней стоимости, которая рассчитывается исходя из общей суммы закупок данного материала и общего количества этого материала, находящегося в ООО.

Поскольку подобная информация имеет совершенно другую структуру, нежели количественный учет, для хранения данных об общей стоимости тех или иных материалов мы будем использовать еще один регистр накопления СтоимостьМатериалов.

Таким образом, документы ПриходнаяНакладная и ОказаниеУслуги должны будут создавать движения не только в регистре ОстаткиМатериалов, но одновременно и в регистре СтоимостьМатериалов, отражая изменения суммового учета.

Создание регистра «Стоимость материалов»

Создайте регистр СтоимостьМатериалов. Этот регистр должен иметь всего одно измерение – Материал с типом СправочникСсылка.Номенклатура и один ресурс – Стоимость с длиной 15 и точностью 2.

После создания регистр СтоимостьМатериалов должен выглядеть в дереве конфигурации следующим образом (рис. 1):

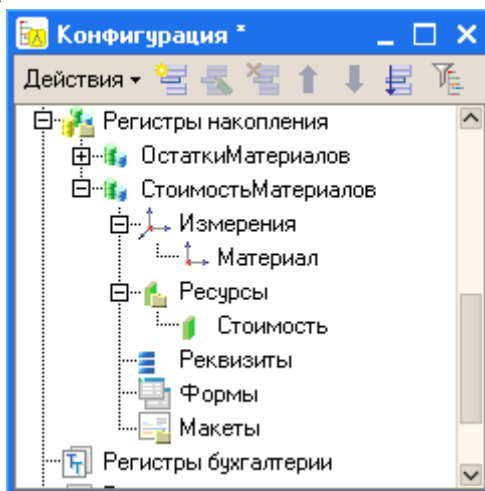


Рис. 1. Регистр «СтоимостьМатериалов»

Теперь можно приступить к внесению изменений в процедуры проведения документов.

Начнем с самого простого – документа Приходная Накладная.

Изменение процедуры проведения документа «Приходная накладная»

1. Откройте в конфигураторе окно редактирования объекта конфигурации Документ Приходная Накладная и перейдите на закладку Движения.
2. В списке регистров отметьте, что документ будет создавать теперь движения и по регистру Стоимость Материалов.
3. Запустите конструктор движений и согласитесь с тем, что существующая процедура ОбработкаПроведения будет замещена.
4. Перед вами откроется окно конструктора движений, которое будет содержать созданные ранее движения документа по регистру Остатки Материалов.
5. Добавьте в список регистров, по которым формируются движения, еще один – Стоимость Материалов. Выберите для него ту же табличную часть Материалы и заполните выражения.
6. Для ресурса Стоимость выберите значения реквизита табличной части Сумма (рис. 2).

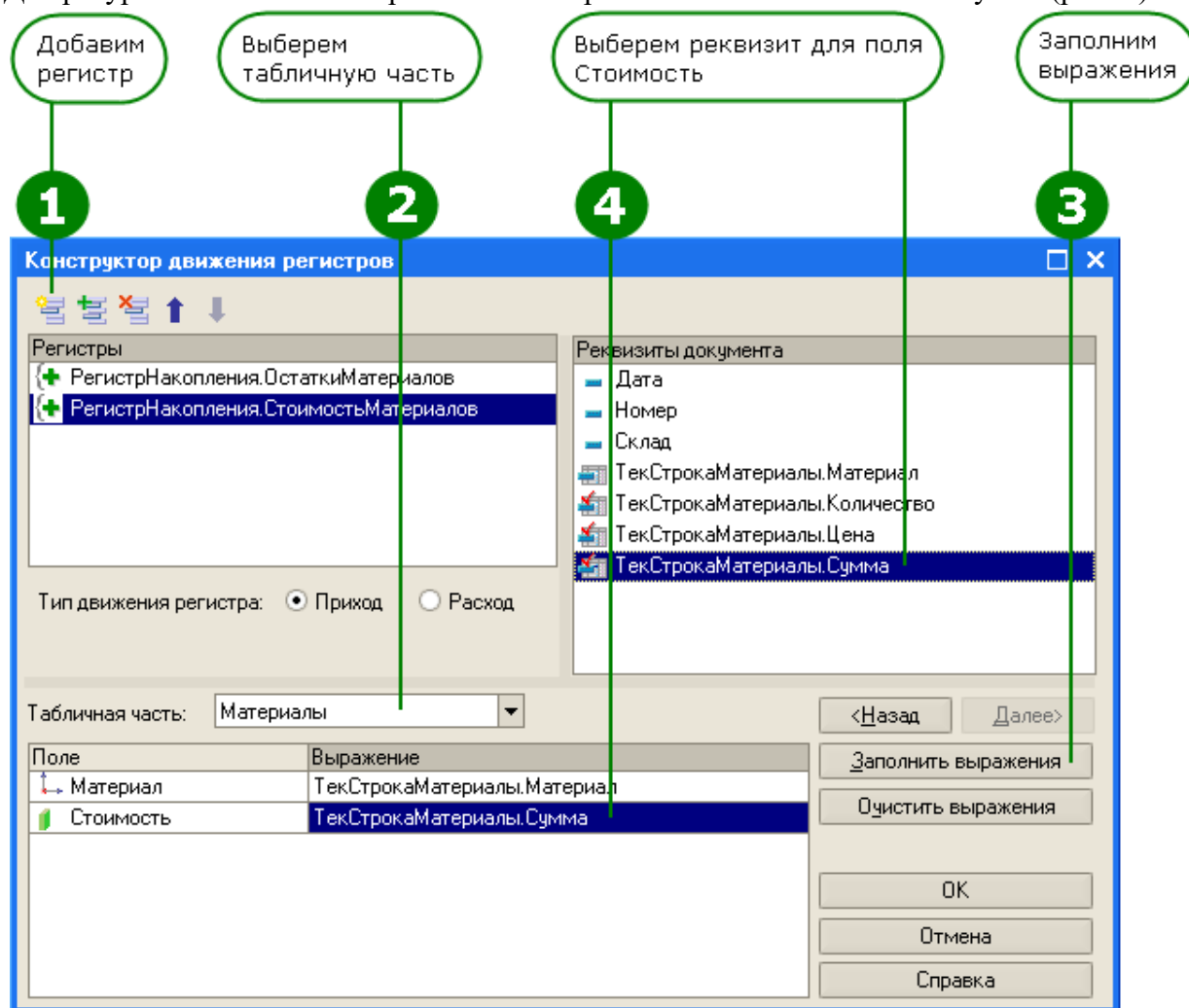


Рис. 2. Добавим в список регистров новый регистр

7. Нажмите ОК и посмотрите на текст, который сформировал конструктор (листинг 1).

Листинг 1. Процедура ОбработкаПроведения()

Процедура ОбработкаПроведения(Отказ, Режим)

```

//{{__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора,
// внесенные вручную изменения будут утеряны!!!

Для Каждого ТекСтрокаМатериалы Из Материалы Цикл
    // Регистр ОстаткиМатериалов Приход
    Движение = Движения.ОстаткиМатериалов.Добавить ( ) ;
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Материал = ТекСтрокаМатериалы.Материал;
    Движение.Склад = Склад;
    Движение.Количество = ТекСтрокаМатериалы.Количество;
КонецЦикла;

Для Каждого ТекСтрокаМатериалы Из Материалы Цикл
    // Регистр СтоимостьМатериалов Приход
    Движение = Движения.СтоимостьМатериалов.Добавить ( ) ;
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Материал = ТекСтрокаМатериалы.Материал;
    Движение.Стоимость = ТекСтрокаМатериалы.Сумма;
КонецЦикла;

//}}__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ

```

КонецПроцедуры

Как видно, конструктор создал два цикла обхода табличной части документа – отдельно для каждого регистра. Так происходит потому, что в общем случае документ может иметь несколько табличных частей, и информация, содержащаяся в каждой из них, может предназначаться для своего отдельного регистра.

8. В нашем случае табличная часть всего одна, поэтому можно объединить эти два цикла в один, закомментировав следующие строки (листинг 2):

Листинг 2. Объединение двух циклов в один

```

// КонецЦикла;
// Для Каждого ТекСтрокаМатериалы Из Материалы Цикл

```

9. Запустите 1С:Предприятие в режиме отладки и перепроведите документы Приходная Накладная №1 и №2.

10. Затем откройте Приходную накладную №1 и убедитесь, что документ создает желаемые записи в регистрах накопления (рис. 3, 4, 5).

Приходная накладная: Приходная накладная 00....33

Действия

Номер: 000000001

Дата: 31.03.2004 11:41:33

Склад: Основной

N	Материал	Количество	Цена	Сумма
1	Строчный трансформатор GoldStar	10,000	270,00	2 700,00
2	Строчный трансформатор Samsung	10,000	600,00	6 000,00
3	Транзистор Philips 2N2369	10,000	3,00	30,00

OK Записать Закрыть

Рис. 3. «ПриходнаяНакладная №1»

Список Остатки материалов

Регистратор	Ном...	Актив...	Период	Материал	Склад	Количество
Приходная накладная 000000001 от 31.03.2004 11:41:33	1	✓	31.03.2004 11:41:33	Строчный трансформатор GoldStar	Основной	10,000
Приходная накладная 000000001 от 31.03.2004 11:41:33	2	✓	31.03.2004 11:41:33	Строчный трансформатор Samsung	Основной	10,000
Приходная накладная 000000001 от 31.03.2004 11:41:33	3	✓	31.03.2004 11:41:33	Транзистор Philips 2N2369	Основной	10,000

Рис. 4. Записи регистра «ОстаткиМатериалов»

Список Стоимость материалов

Регистратор	Н...	Актив...	Период	Материал	Стоимость
Приходная накладная 000000001 от 31.03.2004 11:41:33	1	✓	31.03.2004 11:41:33	Строчный трансформатор GoldStar	2 700,00
Приходная накладная 000000001 от 31.03.2004 11:41:33	2	✓	31.03.2004 11:41:33	Строчный трансформатор Samsung	6 000,00
Приходная накладная 000000001 от 31.03.2004 11:41:33	3	✓	31.03.2004 11:41:33	Транзистор Philips 2N2369	30,00

Рис. 5. Записи регистра «СтоимостьМатериалов»

Изменение процедуры проведения документа «Оказание услуги»

Теперь внесем изменения в процедуру обработки проведения документа ОказаниеУслуги. На данном этапе будем исходить из пожелания, высказанного руководством ООО «На все руки мастер». Суть его заключается в том, что на первом этапе, при списании материалов, израсходованных в процессе оказания услуги, должна быть возможность указывать различную стоимость для одного и того же материала, которая рассчитана руководством исходя из текущих конъюнктурных соображений.

Поскольку в документе ОказаниеУслуги отражена только цена номенклатуры, необходимо добавить в табличную часть документа еще одно поле, в котором будет указываться стоимость номенклатуры. Для этого:

1. Откройте в конфигураторе окно редактирования объекта конфигурации Документ ОказаниеУслуги.
2. Перейдите на закладку Данные.

3. Создайте новый реквизит табличной части документа с именем Стоимость, типом Число, длиной 15 и точностью 2 (рис. 6).
4. После этого откройте форму ФормаДокумента документа ОказаниеУслуги.
5. Добавьте в табличное поле колонку, отображающую новый реквизит Стоимость, расположив ее после колонки Номенклатура (рис. 7). Для этого выделите табличное поле и выполните команду контекстного меню Размещение данных. В окне Размещение данных отметьте реквизит Стоимость и нажмите ОК. После этого с помощью мыши перетащите колонку Стоимость после колонки Номенклатура.

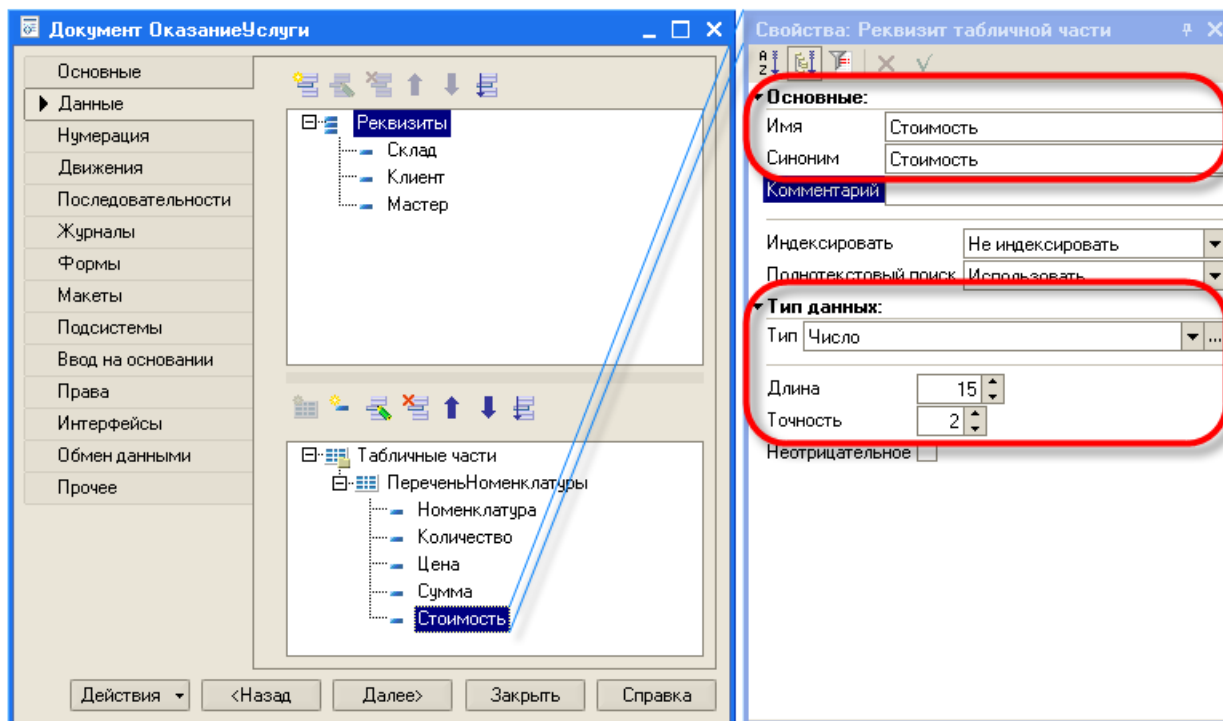


Рис. 6. Изменение документа «ОказаниеУслуги»

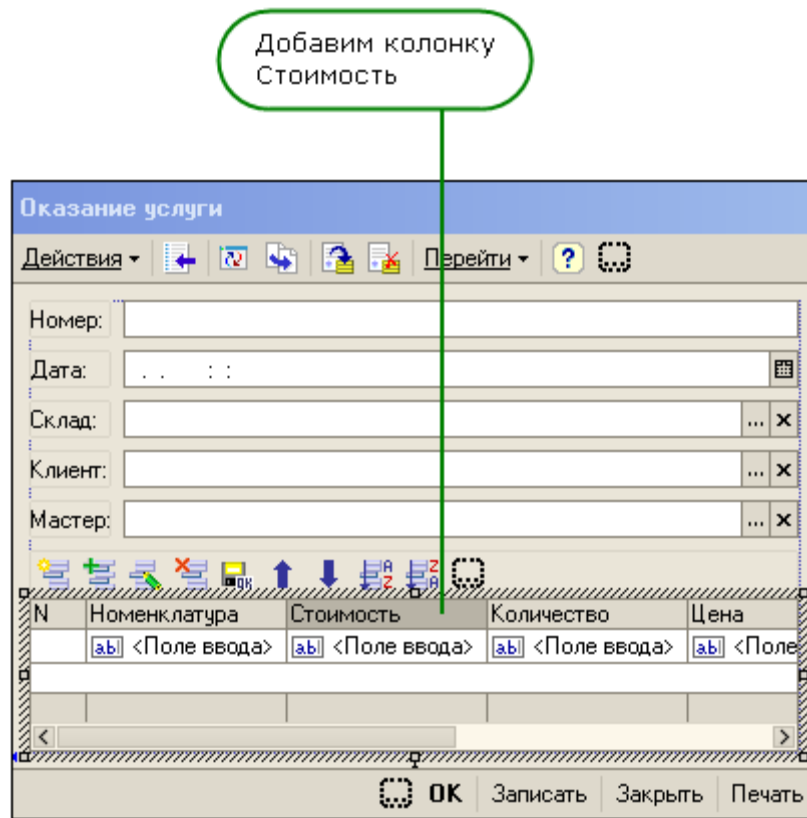


Рис. 7. Изменение формы документа «ОказаниеУслуги»

Теперь создайте движения документа ОказаниеУслуги таким же образом, как для документа ПриходнаяНакладная.

6. Откройте в конфигураторе окно редактирования объекта конфигурации Документ ОказаниеУслуги и укажите, что он будет создавать движения по регистру накопления СтоимостьМатериалов.

7. Запустите конструктор движений документа и добавьте в список регистров регистр СтоимостьМатериалов.

8. Опишите движения документа следующим образом (обратите внимание, что стоимость вычисляется как произведение стоимости и количества, указанных в табличной части) – рис. 8:

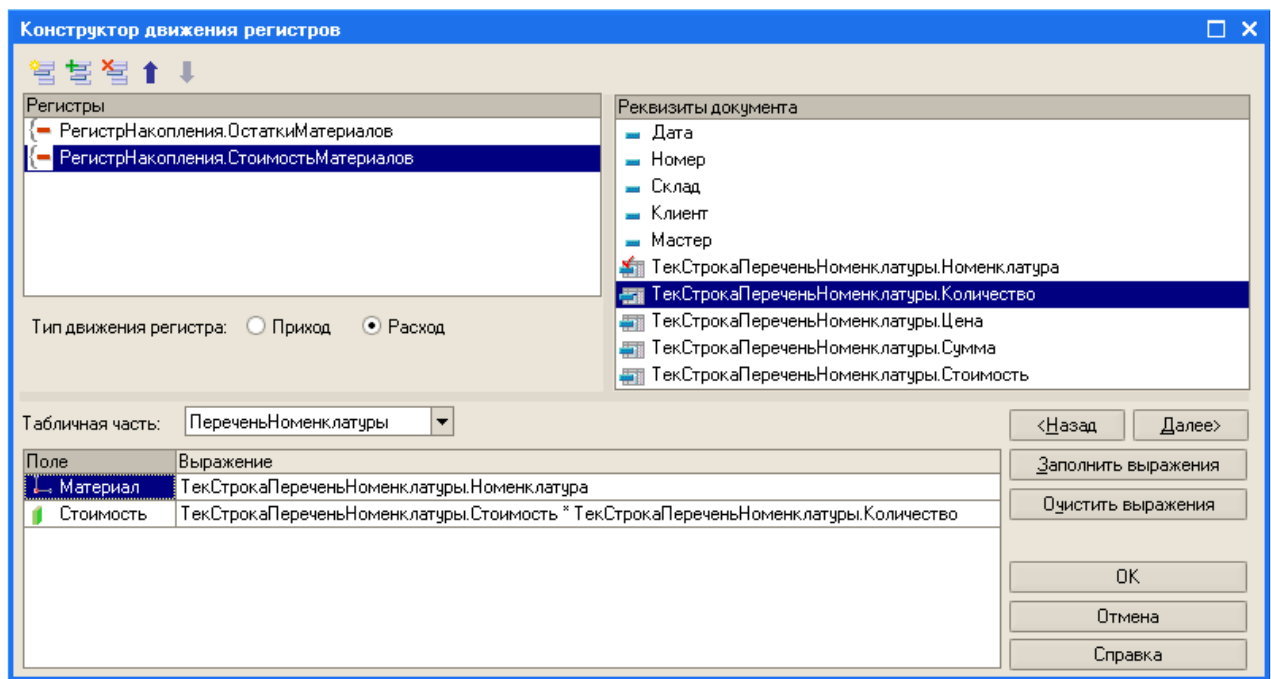


Рис. 8. Конструктор движений документа

9. Нажмите ОК и в тексте, сформированном конструктором (листинг 3), восстановите изменения, внесенные нами ранее (не записывать движения, если номенклатура – не материал) Также объедините два цикла обхода табличной части документа в один (листинг 3).

Листинг 3. Процедура ОбработкаПроведения()

Процедура ОбработкаПроведения(Отказ, Режим)

```
//{{__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора,
// внесенные вручную изменения будут утеряны!!!
```

Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл

Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры <>
Перечисления.ВидыНоменклатуры.Материал Тогда

Продолжить;
КонецЕсли;

```
// Регистр ОстаткиМатериалов Расход
Движение = Движения.ОстаткиМатериалов.Добавить ();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Склад = Склад;
Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;
```

```
// Регистр СтоимостьМатериалов Расход
Движение = Движения.СтоимостьМатериалов.Добавить ();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
```

```

Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Стоимость *
ТекСтрокаПереченьНоменклатуры.Количество;

```

```

КонецЦикла;

```

```

// Записываем движения регистров

```

```

Движения.ОстаткиМатериалов.Записать ();

```

```

Движения.СтоимостьМатериалов.Записать ();

```

```

//}}__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ

```

КонецПроцедуры

Проверьте, как теперь работает проведение документа ОказаниеУслуги.

10. Запустите 1С:Предприятие в режиме отладки и укажите стоимость выбранных материалов в документе ОказаниеУслуги №1 (рис. 9).

N	Номенклатура	Стоимость	Количество	Цена	Сумма
1	Подключение воды		1,000	800,00	800,00
2	Шланг резиновый	100,00	1,000	150,00	150,00
Всего:					950,00

Рис. 9. Документ «ОказаниеУслуги №1»

11. Проведите документ ОказаниеУслуги №1 и посмотрите на движения этого документа по регистру СтоимостьМатериалов (рис. 10).

Регистратор	Н...	Актив...	Период	Материал	Стоимость
Оказание услуги 000000001 от 31.03.2004 14:00:00	1	✓	31.03.2004 14:00:00	Шланг резиновый	100,00

Рис.10. Записи регистра «СтоимостьМатериалов»

12. Теперь создайте и проведите еще два документа ОказаниеУслуги. Эти документы понадобятся нам в дальнейшем, поэтому будьте внимательны и обратите внимание на то, что эти документы созданы другими датами (рис. 11, 12).

Оказание услуги: Оказание услуги 000000002 от 14.04.2004 1...:42

Действия

Номер: 000000002

Дата: 14.04.2004 18:26:42

Склад: Основной

Клиент: Спиридонова Галина

Мастер: Гусаков Николай Дмитриевич

N	Номенклатура	Стоимость	Количество	Цена	Сумма
1	Ремонт импортного телевизора		1,000	800,00	800,00
2	Строчный трансформатор Samsung	600,00	1,000	900,00	900,00
					Всего: 1 700,00

OK Записать Закрыть Печать

Рис. 11. Документ «ОказаниеУслуги №2»

Оказание услуги: Оказание услуги (Новый) *

Действия

Номер: 000000003

Дата: 13.04.2004 16:48:31

Склад: Основной

Клиент: Роман

Мастер: Симонов Валерий Михайлович

N	Номенклатура	Стоимость	Количество	Цена	Сумма
1	Подключение электричества		1,000	800,00	800,00
2	Шланг резиновый	100,00	2,000	150,00	300,00
3	Кабель электрический	20,00	1,000	30,00	30,00
4	Ремонт отечественного телевизора		1,000	600,00	600,00
5	Строчный трансформатор GoldStar	270,00	1,000	400,00	400,00
6	Транзистор Philips 2N2369	3,00	2,000	7,00	14,00
					Всего: 2 144,00

OK Записать Закрыть Печать

Рис. 12. Документ «ОказаниеУслуги №3»

Движения документов ОказаниеУслуги №2 и №3 должны выглядеть соответственно следующим образом (рис. 13, 14):

Регистратор	Н...	Актив...	Период	Материал	Стоимость
Оказание услуги 000000002 от 14.04.2004 18:26:42	1	✓	14.04.2004 18:26:42	Строчный трансформатор Samsung	600,00

Рис. 13. Движения документа «ОказаниеУслуги №2»

Регистратор	Н...	Актив...	Период	Материал	Стоимость
Оказание услуги 000000003 от 13.04.2004 16:48:31	1	✓	13.04.2004 16:48:31	Шланг резиновый	200,00
Оказание услуги 000000003 от 13.04.2004 16:48:31	2	✓	13.04.2004 16:48:31	Кабель электрический	20,00
Оказание услуги 000000003 от 13.04.2004 16:48:31	3	✓	13.04.2004 16:48:31	Строчный трансформатор GoldStar	270,00
Оказание услуги 000000003 от 13.04.2004 16:48:31	4	✓	13.04.2004 16:48:31	Транзистор Philips 2N2369	6,00

Рис. 14. Движения документа «ОказаниеУслуги №3»

Контрольные вопросы

- для чего может понадобиться проведение документа по нескольким регистрам
- как с помощью конструктора создать движения документа по нескольким регистрам
- исходя из каких соображений, конструктор формирует текст процедуры проведения по нескольким регистрам

Лабораторная работа №20. Оборотный регистр накопления

Цель: 1. Познакомимся с еще одним видом регистра накопления – *оборотным регистром накопления*.

2. Узнать принципы выбора измерений и реквизитов регистров накопления.
3. Научиться создавать оборотный регистр накопления и работать с ним.

Зачем нужно создавать еще один регистр?

Продолжим рассматривать работу нашего документа ОказаниеУслуги. До сих пор мы создавали в регистрах накопления движения только для строк документа, которые содержат материалы. Услуги, содержащиеся в документе, мы никак не учитывали.

Дело в том, что при учете услуг важны совершенно другие критерии, нежели при учете материалов. Прежде всего, бессмысленно говорить о том, сколько услуг было и сколько их осталось, важна только сумма и количество услуг, которые были оказаны за определенный промежуток времени. Кроме этого интересны следующие моменты:

- какие именно услуги были оказаны (чтобы составить рейтинг услуг);
- какому именно клиенту оказывались услуги (чтобы предоставить ему скидку от объема оплаченных ранее услуг, например);
- какой мастер предоставлял услуги (чтобы начислить ему заработную плату).

Очевидно, что существующие регистры накопления совершенно не подходят для решения таких задач. Поэтому мы создадим еще одно «хранилище» данных, которое будет использоваться в нашей программе, – оборотный регистр накопления Продажи.

Создание оборотного регистра накопления «Продажи»

Когда мы создавали регистры ОстаткиМатериалов и СтоимостьМатериалов, мы специально не останавливались на двух видах регистров накопления, которые существуют в системе 1С: Предприятие. Сейчас пришло время сказать об этом пару слов.

Регистры накопления могут быть *регистрами остатков* и *регистрами оборотов*.

Существующие в учебной конфигурации регистры ОстаткиМатериалов и СтоимостьМатериалов являются регистрами остатков. Если вы помните, при создании отчета Материалы в конструкторе запроса мы видели, что для таких регистров система создает три виртуальные таблицы: таблица остатков, оборотов и совокупная таблица остатков и оборотов.

Оборотный регистр накопления очень похож на знакомый уже нам регистр остатков, для которого понятие «остаток» не имеет смысла. Оборотный регистр накапливает только обороты, остатки ему безразличны. Поэтому единственной виртуальной таблицей, которую будет создавать система для такого регистра, будет таблица оборотов.

В остальном оборотный регистр ничем не отличается от регистра остатков.

Следует сказать об одной особенности конструирования регистров накопления, напрямую связанной с возможностью получения остатков.

При создании оборотного регистра накопления нет особой сложности в определении того, какие именно параметры должны являться измерениями регистра – мы можем назначить в качестве его измерений любые нужные нам параметры.

Совсем иная ситуация в случае регистра накопления, поддерживающего накопление остатков. Для него выбор измерений должен выполняться исходя из того, что движения регистра могут быть осуществлены в две стороны: приход и расход. Таким образом, в качестве измерений нужно выбирать те параметры, по которым движения точно будут осуществляться как в одну, так и в другую сторону.

Например, если ведется учет материалов в разрезах номенклатуры и склада, очевидно, что и номенклатура, и склад могут быть измерениями, поскольку как приход, так и расход

материалов всегда будут осуществляться с указанием конкретной номенклатуры и конкретного склада. Если же в этой ситуации появляется желание отразить учет материалов еще и в разрезе поставщика, то здесь уже нужно исходить из конкретной схемы учета, принятой на предприятии.

Скорее всего, при поступлении материалов поставщик будет указан, а вот при расходе материалов, с большой долей вероятности, поставщик указываться не будет, так как в большинстве случаев это совершенно лишняя информация. Значит, поставщика следует добавить как реквизит регистра накопления.

Если же при расходе материалов поставщик будет указываться наверняка, имеет смысл добавить поставщика в измерения регистра.

Иными словами, по каждому из *измерений* регистра накопления остатков изменение ресурсов обязательно должно осуществляться в обе стороны: приход и расход.

Для *реквизитов* регистра этот принцип неважен. По реквизитам регистра ресурсы могут только приходоваться или только расходоваться.

Нарушение этого принципа построения регистров накопления будет вести к непроизводительному использованию ресурсов системы и как следствие замедлению работы и падению производительности.

Для создания регистра накопления необходимо выполнить следующие действия:

1. Откройте конфигуратор и создайте новый объект конфигурации Регистр накопления.
2. Назовите его Продажи и определите вид регистра – Обороты.
3. На закладке Данные создайте измерения регистра:
 - Номенклатура, тип СправочникСсылка.Номенклатура,
 - Клиент, тип СправочникСсылка.Клиенты,
 - Мастер, тип СправочникСсылка.Сотрудники.

У регистра будет три ресурса:

- Количество, тип Число, длина 15, точность 3,
 - Выручка, тип Число, длина 15, точность 2,
 - Стоимость, тип Число, длина 15, точность 2.
4. Откройте окно редактирования объекта конфигурации Документ ОказаниеУслуги и на закладке Движения укажите, что этот документ будет создавать движения по регистру Продажи.
 5. Запустите 1С:Предприятие в режиме отладки и откройте формы списка регистров накопления Продажи и ОстаткиМатериалов. Обратите внимание, что формы практически одинаковы, за исключением состава измерений и ресурсов.

Изменение процедуры проведения документа «Оказание услуги»

На этот раз мы не будем использовать конструктор движений документа, а внесем необходимые дополнения прямо в обработчик события ОбработкаПроведения документа ОказаниеУслуги.

1. Откройте в конфигураторе модуль объекта конфигурации Документ ОказаниеУслуги и найдите в нем процедуру обработчика события ОбработкаПроведения.
2. Сразу после окончания первого цикла создайте еще один цикл обхода табличной части и команду записи движений регистра Продажи (листинг 1).

Листинг 1. Добавление цикла обхода табличной части и записи движений регистра «Продажи»

...

КонецЦикла;

Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры
Цикл

// Регистр Продажи

КонецЦикла;

//}}__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ

3. Теперь в тело созданного цикла вставьте команды создания движений регистра Продажи (листинг 2).

Листинг 2. Команды создания движений регистра «Продажи»

Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры
Цикл

// Регистр Продажи

Движение = Движения.Продажи.Добавить () ;

Движение.Период = Дата;

Движение.Номенклатура =

ТекСтрокаПереченьНоменклатуры.Номенклатура;

Движение.Клиент = Клиент;

Движение.Мастер = Мастер;

Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;

Движение.Выручка = ТекСтрокаПереченьНоменклатуры.Сумма;

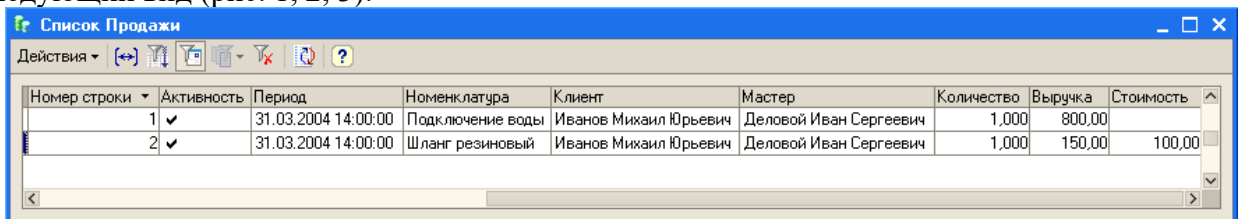
Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Стоимость *

ТекСтрокаПереченьНоменклатуры.Количество;

КонецЦикла;

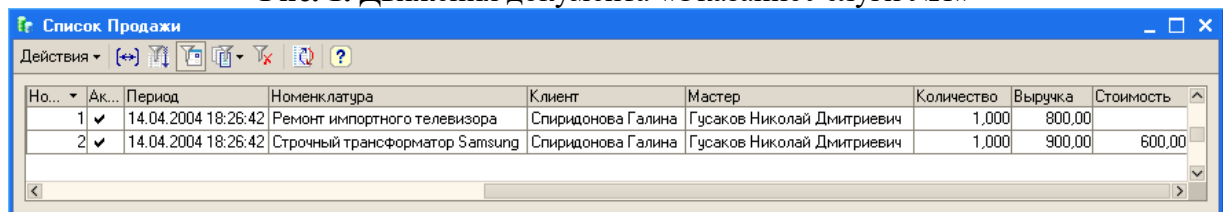
Все добавленные конструкции вам уже хорошо известны. Обратите внимание лишь на то, что у оборотного регистра отсутствует свойство ВидДвижения, поскольку отражение вида движения (приход или расход) имеет смысл лишь при учете остатков. В случае регистра оборотов нас интересует только значение, которое должно быть записано в ресурс регистра.

4. Запустите 1С:Предприятие в режиме отладки и перепроведите все документы ОказаниеУслуги. Движения этих документов по регистру Продажи должны иметь следующий вид (рис. 1, 2, 3):



Номер строки	Активность	Период	Номенклатура	Клиент	Мастер	Количество	Выручка	Стоимость
1	✓	31.03.2004 14:00:00	Подключение воды	Иванов Михаил Юрьевич	Деловой Иван Сергеевич	1,000	800,00	
2	✓	31.03.2004 14:00:00	Шланг резиновый	Иванов Михаил Юрьевич	Деловой Иван Сергеевич	1,000	150,00	100,00

Рис. 1. Движения документа «ОказаниеУслуги №1»



Но...	Ак...	Период	Номенклатура	Клиент	Мастер	Количество	Выручка	Стоимость
1	✓	14.04.2004 18:26:42	Ремонт импортного телевизора	Спиридонова Галина	Гусаков Николай Дмитриевич	1,000	800,00	
2	✓	14.04.2004 18:26:42	Строчный трансформатор Samsung	Спиридонова Галина	Гусаков Николай Дмитриевич	1,000	900,00	600,00

Рис. 2. Движения документа «ОказаниеУслуги №2»

Номер строки	Ак...	Период	Номенклатура	Клиент	Мастер	Количество	Выручка	Стоимость
1	✓	13.04.2004 16:48:31	Подключение электричества	Роман	Симонов Валерий Михайлович	1,000	800,00	
2	✓	13.04.2004 16:48:31	Шланг резиновый	Роман	Симонов Валерий Михайлович	2,000	300,00	200,00
3	✓	13.04.2004 16:48:31	Кабель электрический	Роман	Симонов Валерий Михайлович	1,000	30,00	20,00
4	✓	13.04.2004 16:48:31	Ремонт отечественного телевизора	Роман	Симонов Валерий Михайлович	1,000	600,00	
5	✓	13.04.2004 16:48:31	Строчный трансформатор GoldStar	Роман	Симонов Валерий Михайлович	1,000	400,00	270,00
6	✓	13.04.2004 16:48:31	Транзистор Philips 2N2369	Роман	Симонов Валерий Михайлович	2,000	14,00	6,00

Рис. 3. Движения документа «ОказаниеУслуги №3»

Контрольные вопросы

- ☐ что такое оборотный регистр накопления
- ☐ в чем отличие между регистром накопления остатков и оборотным регистром накопления
- ☐ как выбирать реквизиты и измерения при создании регистров накопления
- ☐ как создать оборотный регистр накопления
- ☐ как создать движения документа без использования конструктора движений

Лабораторная работа №21. Создание сложных отчетов

Цель: 1. Познакомиться с инструментом платформы 1С:Предприятие – *системой компоновки данных*.

2. Изучить основные возможности системы компоновки данных.

3. Научиться создавать простые отчеты.

Любой отчет, как правило, подразумевает получение сложной выборки данных, сгруппированных и отсортированных определенным образом. Система компоновки данных представляет собой мощный и гибкий механизм, позволяющий выполнить все необходимые действия – от получения данных из различных источников, до представления этих данных в виде, удобном для пользователя.

Чаще всего исходные данные, необходимые для отчета, находятся в базе данных. Для того, чтобы указать системе компоновки данных, какая информация и откуда должна быть получена, используется язык запросов системы 1С:Предприятие 8.1.

На этапе разработки отчета можно задать настройки отчета по умолчанию для того, чтобы пользователь мог сразу же запустить отчет на выполнение.

В то же время пользователь может самостоятельно изменить настройки отчета и выполнить его. При этом система компоновки данных сгенерирует другой запрос, и другим образом представит конечные данные – в соответствии с новыми настройками, заданными пользователем.

Работа с запросами

Для работы с запросами используется объект встроенного языка *Запрос*. Он позволяет получать информацию, хранящуюся в полях базы данных, в виде выборки, сформированной по заданным правилам.

Источники данных запросов

Исходную информацию запрос получает из набора таблиц. Эти таблицы представляют данные реальных таблиц базы данных в удобном для анализа виде. Их можно разделить на две большие группы: *реальные* и *виртуальные*.

Реальные таблицы, в свою очередь, могут быть объектными (ссылочными) или не объектными (не ссылочными), рис. 1.



Рис. 1. Таблицы запроса

Отличительной особенностью реальных таблиц является то, что они содержат данные какой-либо одной реальной таблицы, хранящейся в базе данных. Например, реальной является таблица *Справочник.Клиенты*, соответствующая справочнику *Клиенты*, или таблица *РегистрНакопления.ОстаткиМатериалов*, соответствующая регистру *накопления ОстаткиМатериалов*.

Виртуальные таблицы формируются в основном из данных нескольких таблиц базы данных. Например, виртуальной является таблица

РегистрНакопления.ОстаткиМатериалов.ОстаткиИОбороты, формируемая из нескольких таблиц регистра накопления ОстаткиМатериалов.

Иногда виртуальные таблицы могут формироваться и из одной реальной таблицы (например, виртуальная таблица Цены.СрезПоследних формируется на основе таблицы регистра сведений Цены). Однако общим для всех виртуальных таблиц является то, что им можно задать ряд параметров, определяющих какие данные будут включены в эти виртуальные таблицы. Набор таких параметров может быть различным для разных виртуальных таблиц, и определяется данными, хранящимися в исходных таблицах базы данных.

Реальные таблицы подразделяются на *объектные (ссылочные)* и *не объектные (не ссылочные)*.

В *объектных (ссылочных) таблицах* представлена информация ссылочных типов данных (справочники, документы, планы видов характеристик и т. д.). А в *не объектных (не ссылочных)* – всех остальных типов данных (константы, регистры и т. д.).

Отличительной особенностью *объектных (ссылочных) таблиц* является то, что они включают в себя поле Ссылка, содержащее ссылку на текущую запись. Кроме этого для таких таблиц возможно получение пользовательского представления объекта. Эти таблицы могут быть иерархическими, и поля таких таблиц могут содержать вложенные таблицы (табличные части).

Язык запросов

Алгоритм, по которому данные будут выбраны из исходных таблиц запроса, описывается в тексте запроса на специальном языке – *языке запросов*. Текст запроса состоит из нескольких частей:

- описание запроса;
- объединение запросов;
- упорядочивание результатов;
- АВТОУПОРЯДОЧИВАНИЕ;
- описание итогов.

Обязательной частью запроса является только первая – описание запроса. Все остальные присутствуют по необходимости.

Описание запроса определяет источники данных, поля выборки, группировки и т. д.

Объединение запросов определяет, как будут объединены результаты выполнения нескольких запросов.

Упорядочивание результатов определяет условия упорядочивания строк результата запроса. **АВТОУПОРЯДОЧИВАНИЕ** позволяет включить режим автоматического упорядочивания строк результата запроса.

Описание итогов определяет, какие итоги необходимо рассчитывать в запросе и каким образом группировать результат.

Следует заметить, что в случае, когда язык запросов используется для описания источников данных в системе компоновки данных, секция описания итогов языка запросов не используется. Это связано с тем, что система компоновки данных самостоятельно рассчитывает итоги на основании тех настроек, которые сделаны разработчиком или пользователем.

Система компоновки данных

Система компоновки данных предназначена для создания произвольных отчетов в системе 1С:Предприятие 8.1 и состоит из нескольких основных частей.

Исходные данные для компоновки отчета содержит в себе схема компоновки данных – это наборы данных и методы работы с ними (рис. 2).

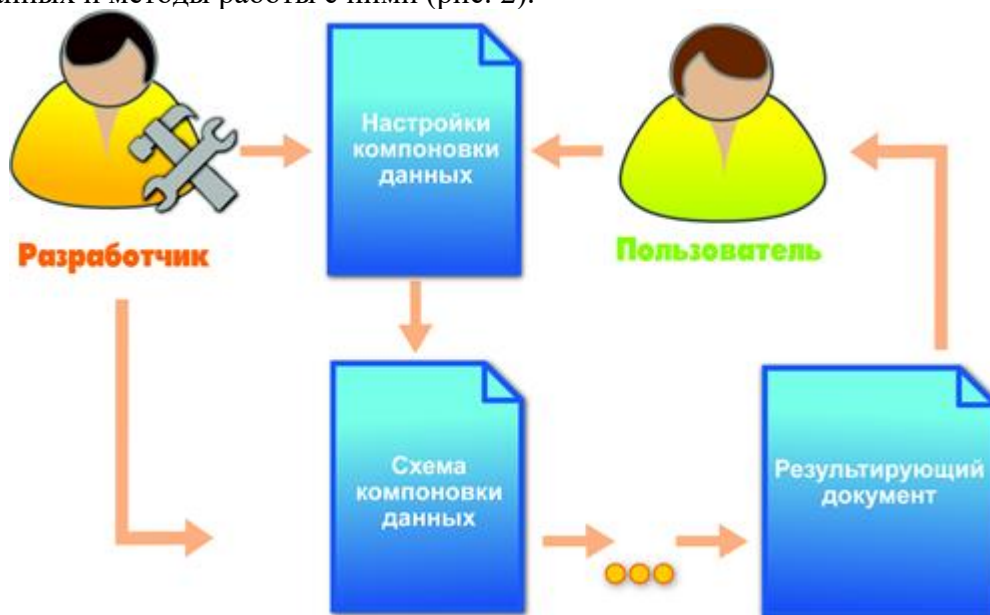


Рис. 2. Общая схема работы с системой компоновки данных

Разработчик создает *схему компоновки данных*, в которой описывает текст запроса, наборы данных, связи между ними, доступные поля, параметры получения данных и задает первоначальные настройки компоновки – структуру отчета, макет оформления данных и др. Например, схема компоновки может содержать следующий набор данных (рис. 3):

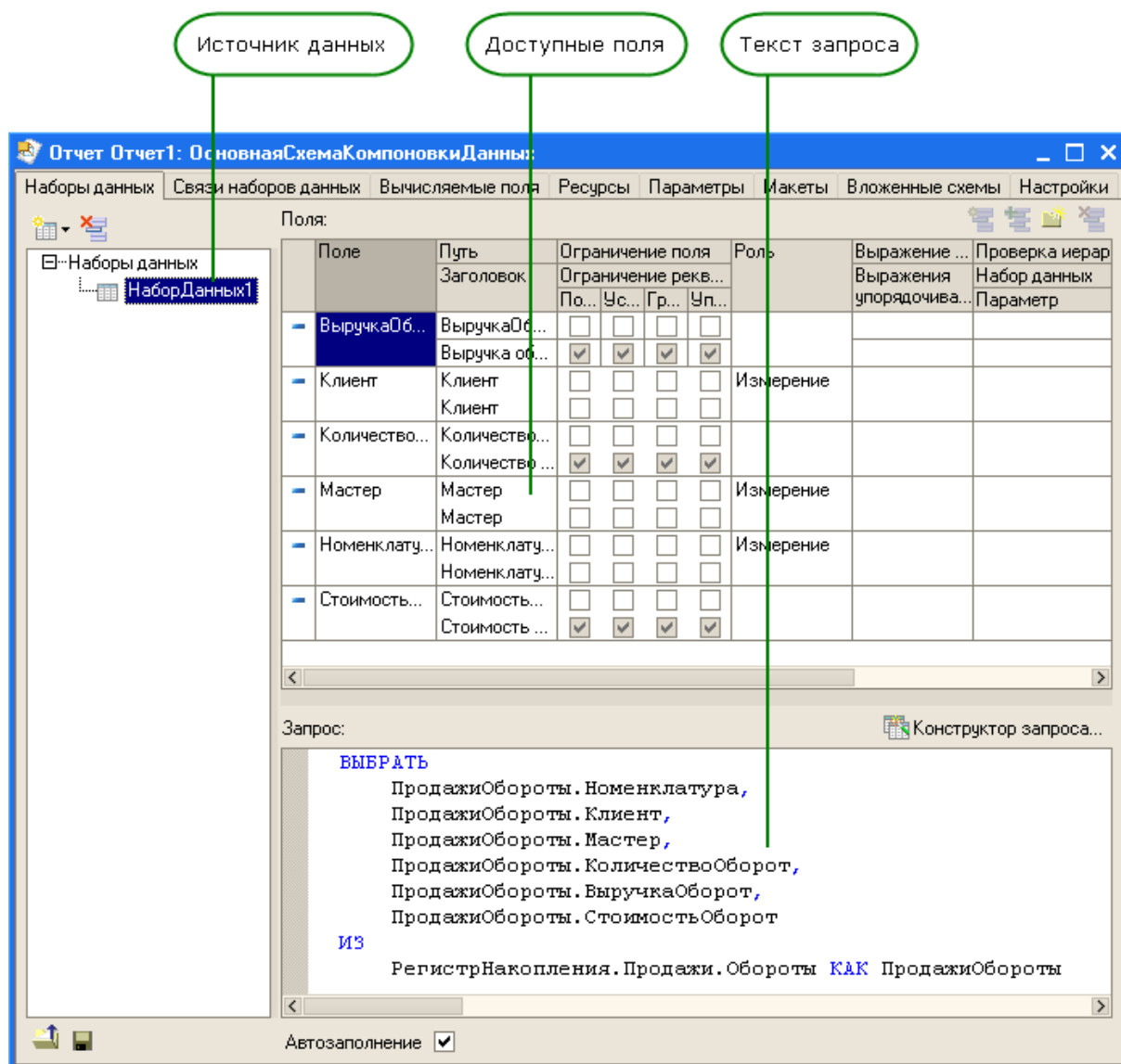


Рис. 3. Пример схемы компоновки (набор данных и запрос, его использующий)

На рисунке 3 показано окно конструктора схемы компоновки данных, в котором содержатся источник данных, текст запроса и поля, выбранные запросом.

Отчет системы компоновки данных, который получит пользователь, представляет собой не просто таблицу записей, удовлетворяющих запросу. Он имеет сложную иерархическую структуру и может состоять из различных элементов, таких как: *группировки*, *таблицы* и *диаграммы*. При этом пользователь может изменить существующую или вообще создать совершенно новую структуру отчета, настроить необходимый ему отбор, оформление элементов структуры отчета, получить расшифровку по каждому элементу и т. д. Например, может быть задана такая структура отчета (рис. 4):

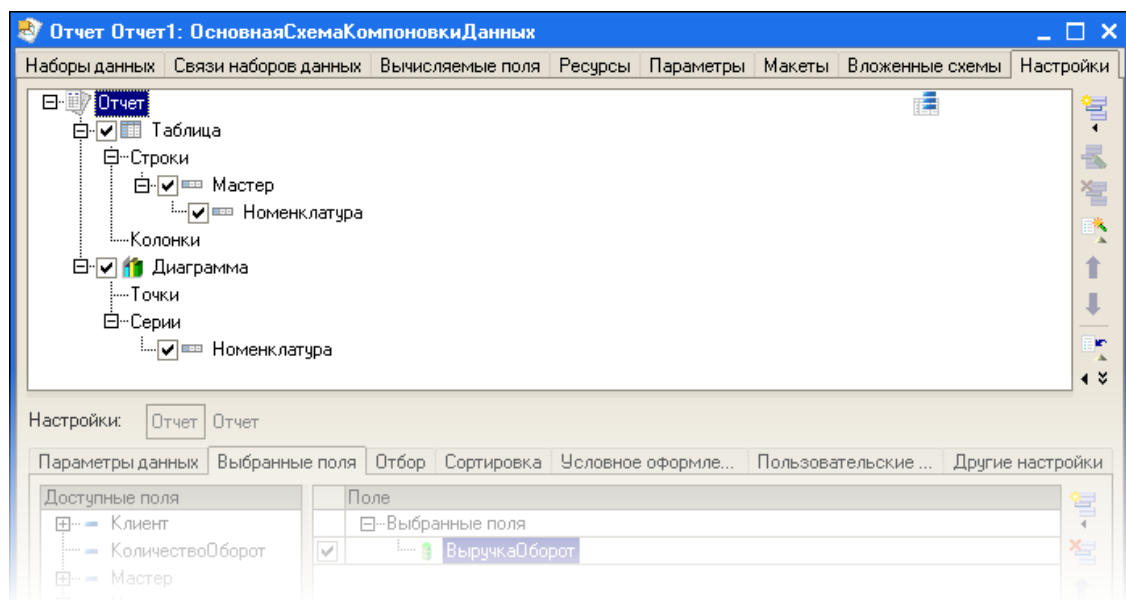


Рис. 4. Структура возможного отчета.

Сформированный отчет может иметь следующий вид (рис. 5):

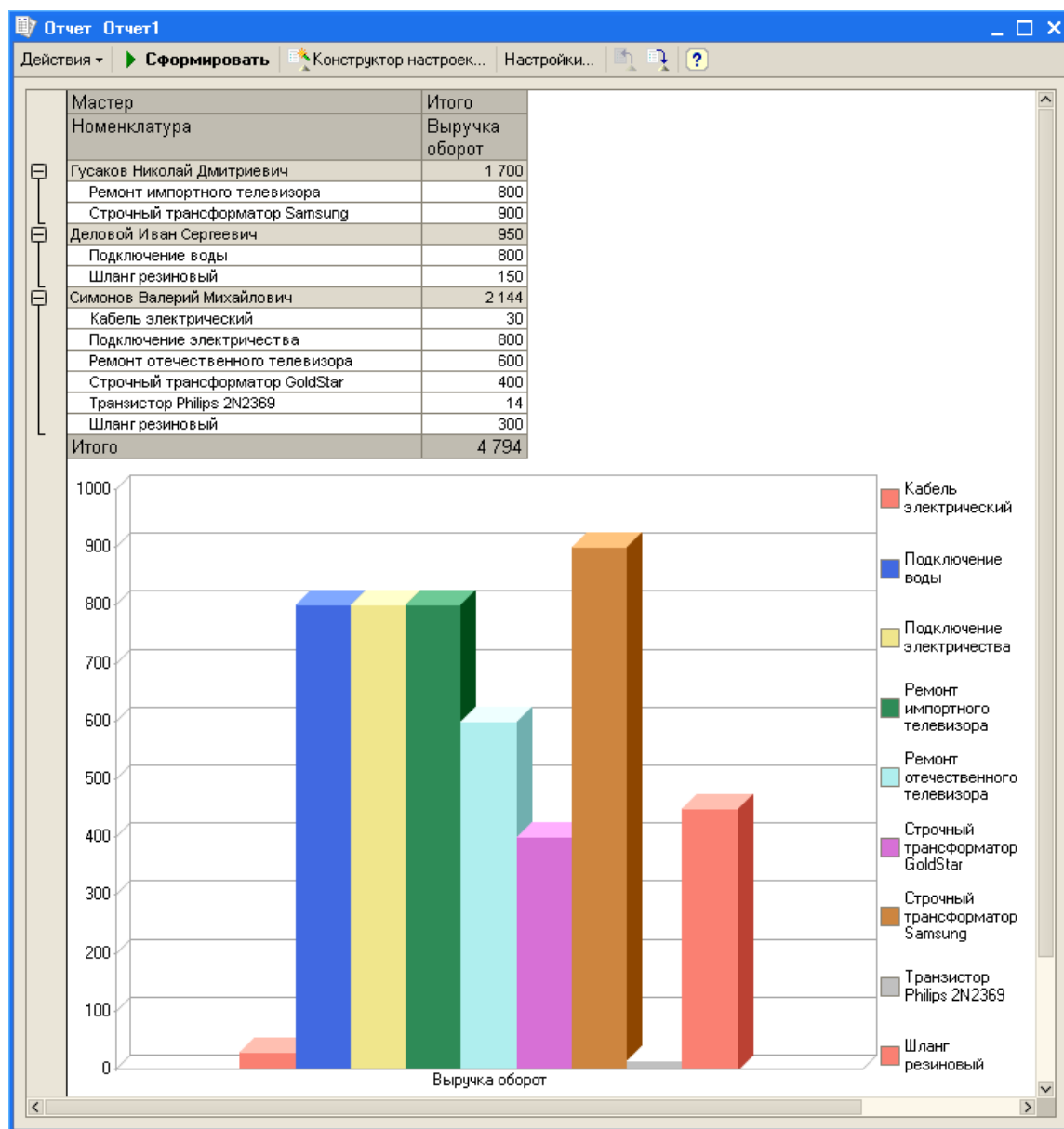


Рис. 5. Пример отчета

В представленном отчете таблица будет состоять из записей регистра накопления ПродажиОбороты о клиентах и оказанных им услугах. Эти записи сгруппированы по мастерам, которые выполняли заказы. А в группировке будет выведен список мастеров. В общем виде система компоновки данных представляет собой совокупность нескольких объектов. При формировании и исполнении отчета происходит последовательная передача данных от одного объекта системы компоновки данных к другому, до получения конечного результата – документа, показанного пользователю.

Алгоритм взаимодействия этих объектов выглядит следующим образом: разработчик создает *схему компоновки данных* и *настройки по умолчанию*. В общем случае на основе одной схемы компоновки данных может быть создано большое количество различных отчетов.

Настройки компоновки данных – создаваемые разработчиком или изменяемые пользователем – определяют, какой именно отчет будет получен в конкретном случае.

На основе схемы компоновки и имеющихся настроек *компоновщик макета* создает *макет компоновки данных*. Это этап подготовки к исполнению отчета. Макет компоновки данных

является уже готовым заданием для выполнения процессором компоновки. Макет компоновки содержит необходимые запросы, макеты областей отчета и др.

Процессор компоновки данных выбирает данные из информационной базы согласно макету компоновки, агрегирует и оформляет эти данные.

Результат компоновки обрабатывается процессором вывода, и в итоге пользователь получает результирующий табличный документ.

Последовательность работы системы компоновки можно представить в виде следующей схемы (рис. 6):

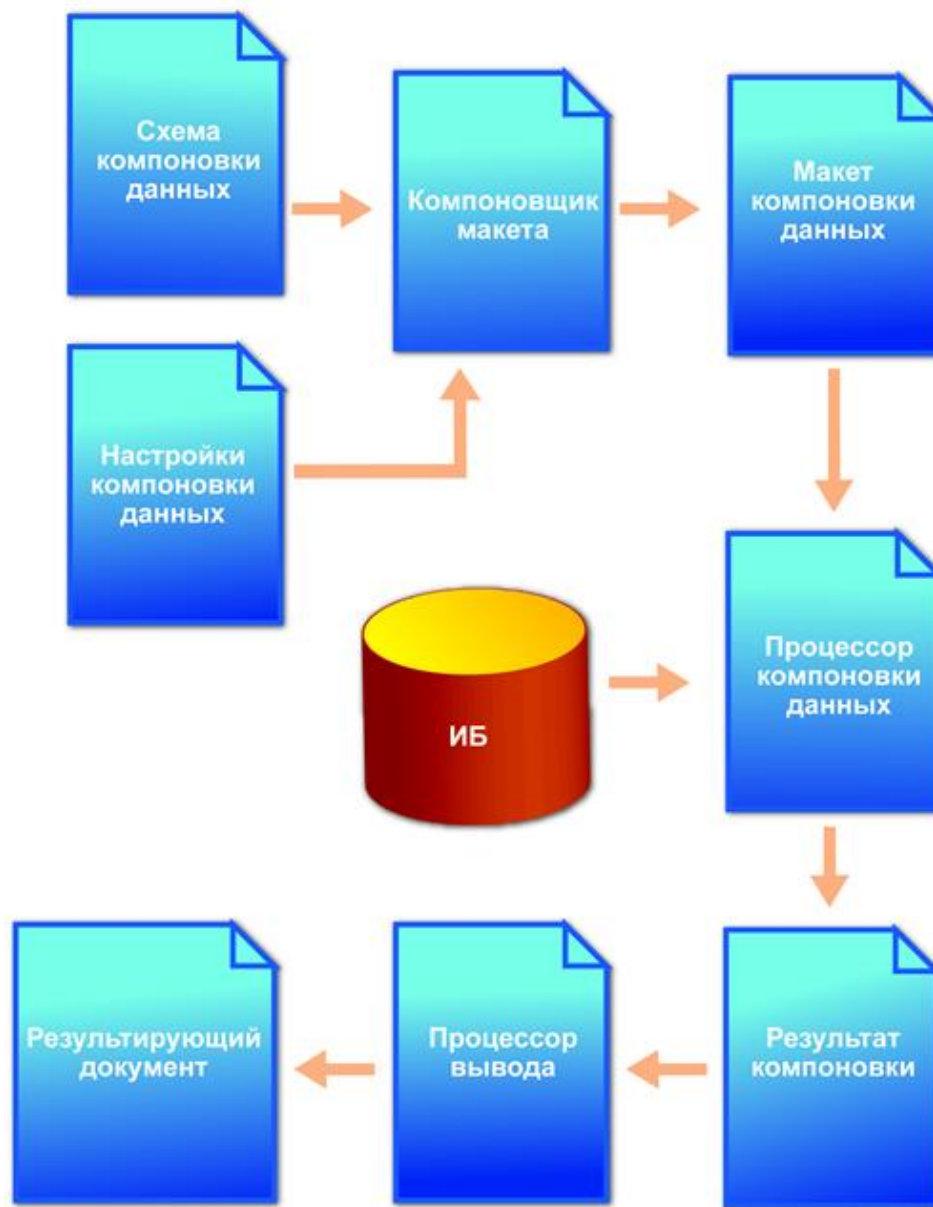


Рис. 6. Схема работы системы компоновки

Отчет «Реестр документов Оказание услуги»

Создадим отчет Реестр документов Оказание услуги, используя систему компоновки данных. Этот отчет будет выводить список существующих в базе данных документов

ОказаниеУслуги в порядке их дат и номеров. Для этого необходимо выполнить следующее:

1. Создайте в конфигураторе новый объект конфигурации Отчет.
2. Присвойте ему имя РеестрДокументовОказаниеУслуги.

-
- Оформление
- Псевдонимы Порядок Компоновка данных Характеристики
- Поля
- ОказаниеУслуги.Дата
 - ОказаниеУслуги.Номер
 - ОказаниеУслуги.Склад
 - ОказаниеУслуги.Мастер
 - ОказаниеУслуги.Клиент
- Назад Далее > OK Отмена Справка

8. После этого перейдите на закладку Порядок и укажите, что результат запроса должен быть сначала упорядочен по значению поля Дата, а затем – по значению поля ОказаниеУслуги.Ссылка (рис. 8).

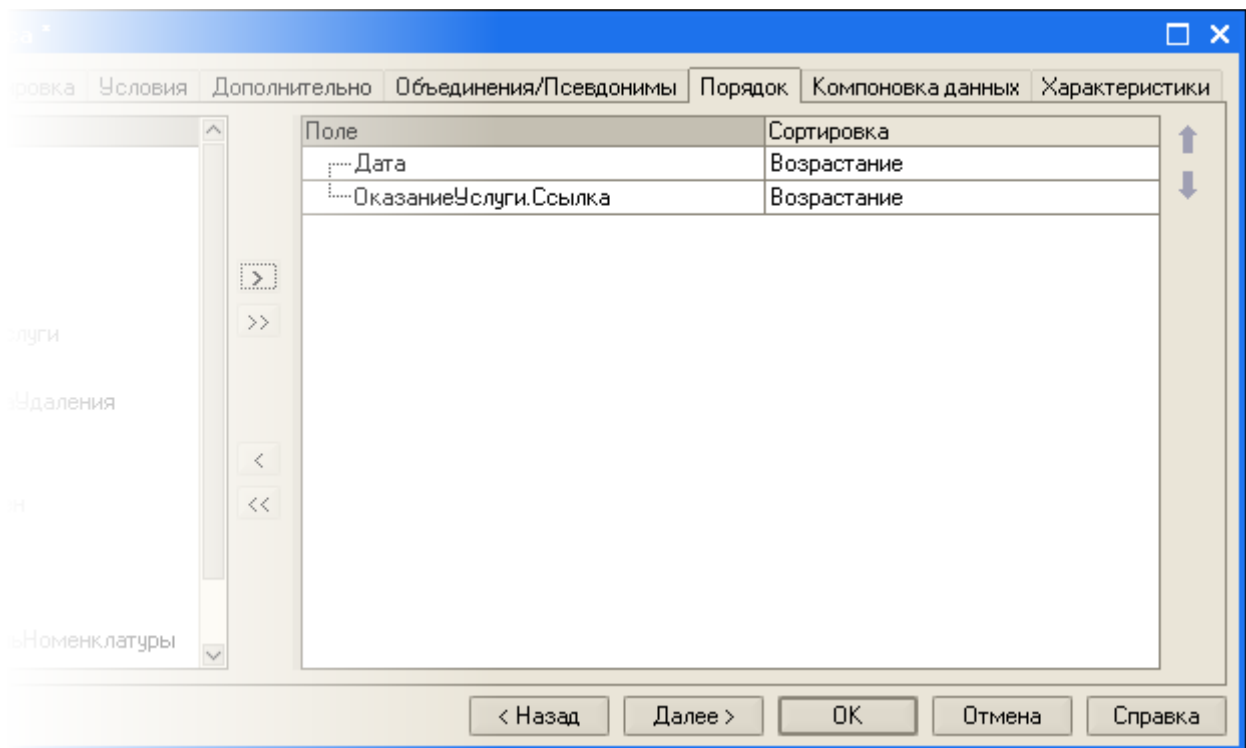


Рис. 8. Порядок вывода результата запроса

9. Нажмите ОК и посмотрите, какой запрос сформировал конструктор запроса (листинг 1).

Листинг 1. Текст запроса

ВЫБРАТЬ

ОказаниеУслуги.Дата КАК Дата,
ОказаниеУслуги.Номер,
ОказаниеУслуги.Склад,
ОказаниеУслуги.Мастер,
ОказаниеУслуги.Клиент

ИЗ

Документ.ОказаниеУслуги КАК ОказаниеУслуги

УПОРЯДОЧИТЬ ПО

Дата,
ОказаниеУслуги.Ссылка

Текст запроса начинается, как мы говорили выше, с части описания запроса (листинг 2).

Листинг 2. Описание запроса

ВЫБРАТЬ

ОказаниеУслуги.Дата КАК Дата,
ОказаниеУслуги.Номер,
ОказаниеУслуги.Склад,
ОказаниеУслуги.Мастер,
ОказаниеУслуги.Клиент

ИЗ

Документ.ОказаниеУслуги КАК ОказаниеУслуги

Описание запроса начинается с обязательного ключевого слова ВЫБРАТЬ. Затем следует список полей выборки. В нем описываются поля, которые должны содержаться в результате

запроса. Этот список может содержать как собственно поля, так и некоторые выражения, вычисляемые на основе значений полей.

После ключевого слова ИЗ указываются источники данных – исходные таблицы запроса, содержимое которых обрабатывается в запросе. В данном случае это объектная (ссылочная) таблица Документ.ОказаниеУслуги. После ключевого слова КАК указывается псевдоним источника данных. В нашем случае это ОказаниеУслуги. В дальнейшем к этому источнику данных можно будет обращаться в тексте запроса, используя псевдоним.

Такое обращение мы видим в описании полей выборки (листинг 5.3).

Листинг 3. Описание полей выборки

ВЫБРАТЬ

ОказаниеУслуги.Дата КАК Дата,
ОказаниеУслуги.Номер,
ОказаниеУслуги.Склад,
ОказаниеУслуги.Мастер,
ОказаниеУслуги.Клиент

Поля выборки также могут иметь псевдонимы, по которым в дальнейшем в тексте запроса можно обращаться к этому полю. В нашем случае это псевдоним Дата.

После части описания запроса в нашем примере следует часть упорядочивания результатов (листинг 4).

Листинг 4. Упорядочивание результатов запроса

УПОРЯДОЧИТЬ ПО

Дата,
ОказаниеУслуги.Ссылка

Предложение УПОРЯДОЧИТЬ ПО позволяет сортировать строки в результате запроса.

После этого ключевого предложения располагается выражение упорядочивания, которое в общем случае представляет собой перечисление полей (выражений) и порядка вывода. В нашем случае упорядочивание будет выполняться сначала по полю Дата, а потом по полю ОказаниеУслуги.Ссылка. В обоих случаях порядок сортировки будет по возрастанию (настроен по умолчанию). На этом закончим изучение текста запроса и перейдем к настройке схемы компоновки данных.

10. Перейдите на закладку Настройки и добавьте новую группировку в структуру отчета (рис. 9).

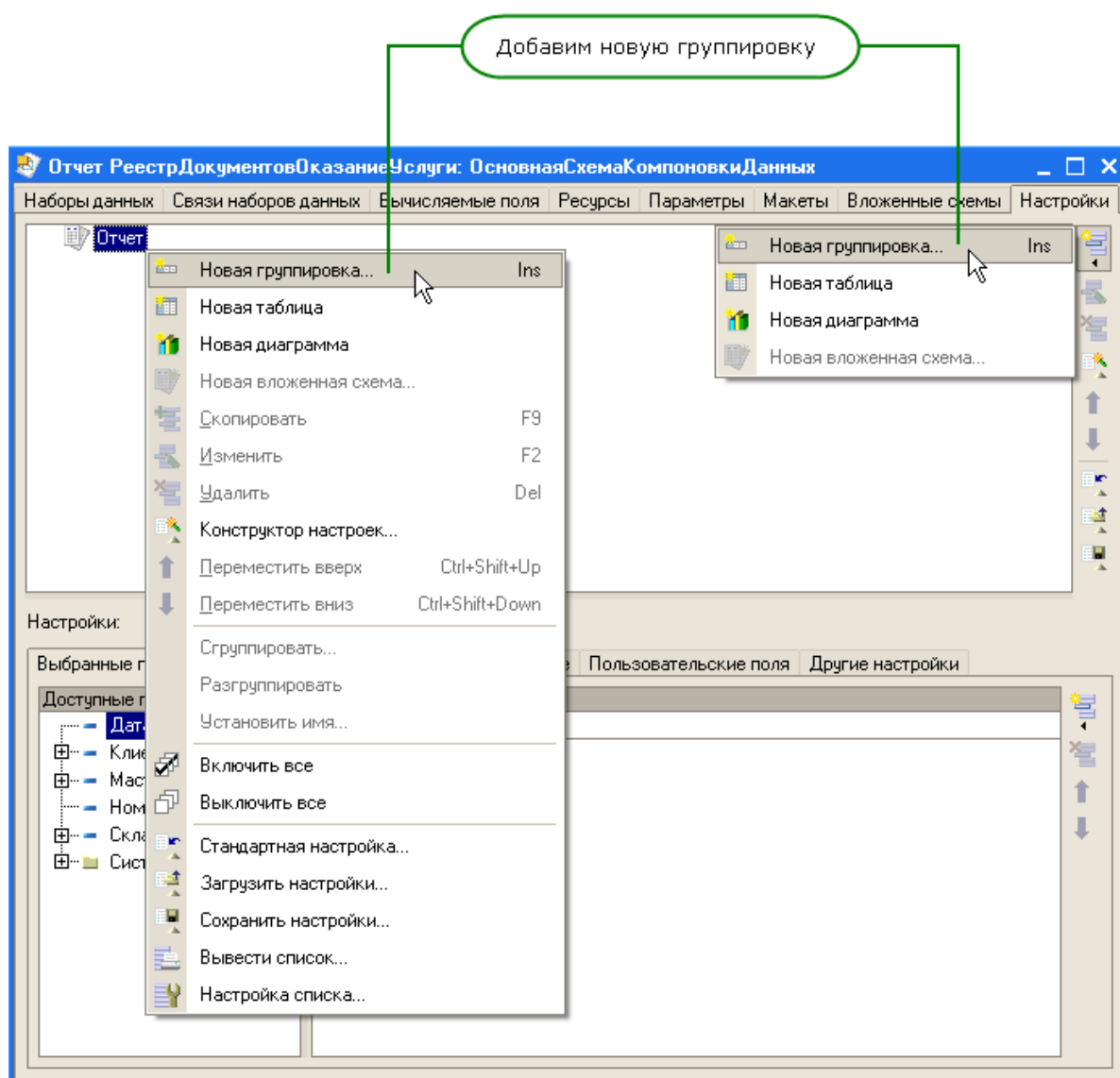


Рис. 9. Командная панель закладки Настройки

11. В окне выбора поля группировки просто нажмите ОК (тем самым мы указываем, что в группировке будут выводиться детальные записи из информационной базы) и на закладке Выбранные поля зададим поля, которые будут выводиться в отчет:

- *Дата;*
- *Номер;*
- *Склад;*
- *Мастер;*
- *Клиент.*

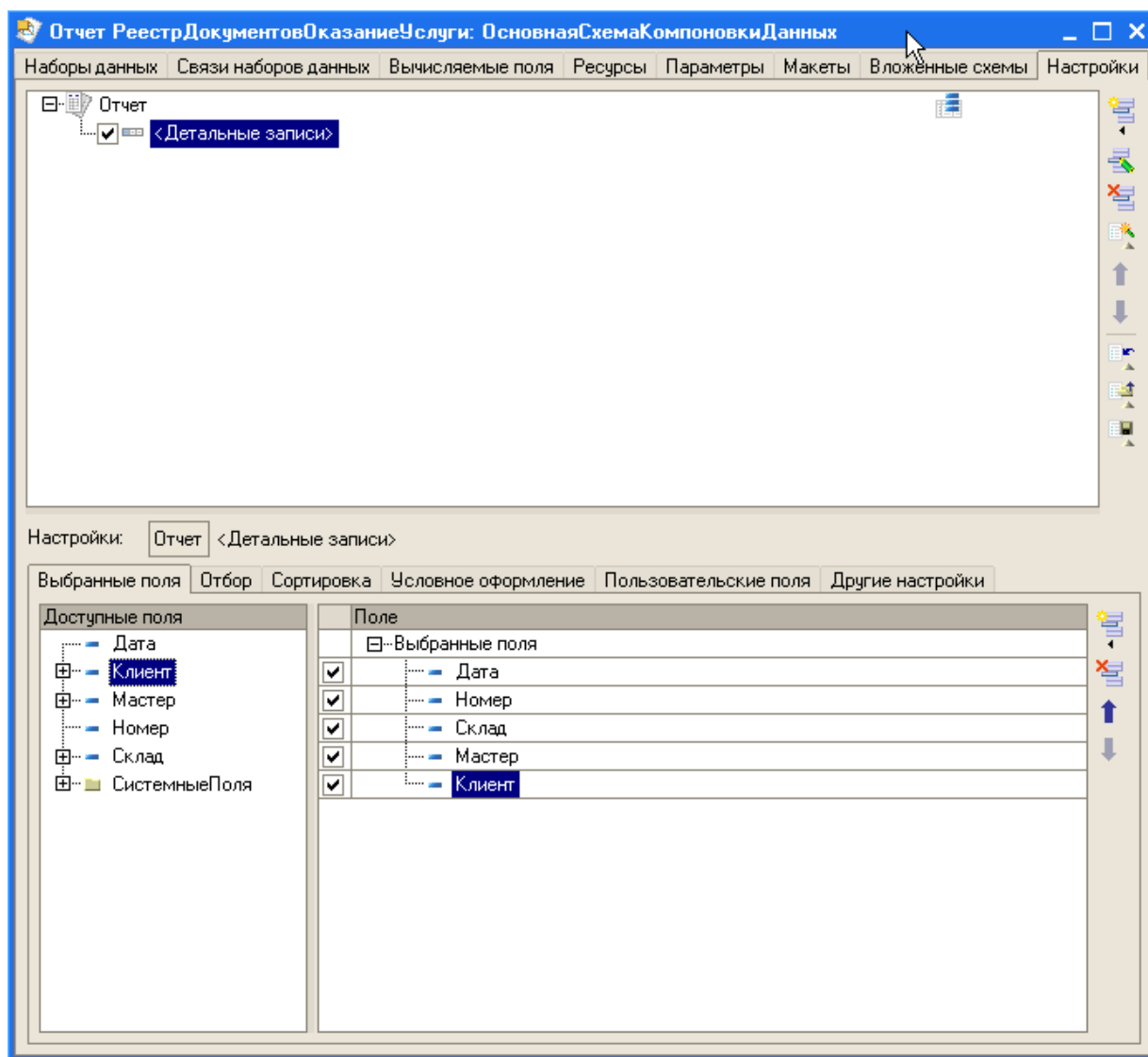


Рис. 10. Настройка структуры отчета

12. Теперь запустите 1С:Предприятие в режиме отладки и откройте отчет Реестр документов Оказание услуги.
13. Нажмите Сформировать и посмотрите на результат работы нашего отчета (рис. 11).

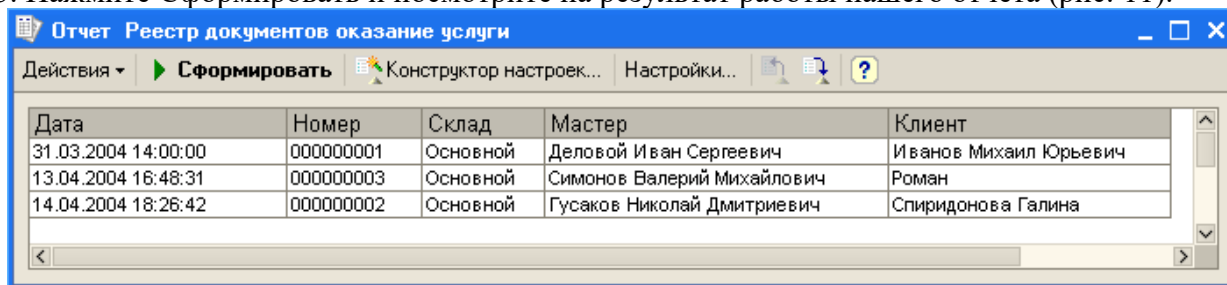


Рис. 11. Отчет Реестр документовоказаниеуслуги

Таким образом, на примере этого отчета вы узнали, как использовать конструктор схемы компоновки данных, и познакомились с некоторыми основными конструкциями языка запросов.

Контрольные вопросы

- ❑ для чего предназначен объект встроенного языка Запрос
- ❑ для чего предназначена система компоновки данных
- ❑ для чего предназначена схема компоновки данных
- ❑ для чего предназначены настройки компоновки данных
- ❑ для чего предназначен объект встроенного языка КомпоновщикНастроек
- ❑ в чем отличие между реальными, виртуальными и ссылочными таблицами
- ❑ из каких частей состоит текст запроса, какие из них являются обязательными
- ❑ каковы основные синтаксические конструкции языка запросов
- ❑ что является источником данных запроса
- ❑ что такое псевдонимы в языке запросов
- ❑ как использовать конструктор запроса