

## Лабораторная работа 4

### Справочники

Лекция посвящена разработке иерархических и подчиненных справочников, реализации дополнительных программных механизмов справочников.

#### О различных видах справочников

В этой лекции мы продолжим работу над справочниками нашей конфигурации, попутно рассмотрим некоторые программные механизмы работы с ними.

В частности, в предыдущей лекции мы создавали обычные справочники – такие справочники содержат простые списки элементов. Однако, справочники в 1С:Предприятие 8 могут иметь более сложную структуру.

Многие справочники, используемые на практике, являются иерархическими. Это означает, что каждому из элементов справочника может быть сопоставлен другой элемент, который называется родителем.

Возможна иерархия различных видов, в частности, при иерархии групп и элементов в справочник, помимо обычных элементов можно включать группы, которые, в свою очередь, могут включать в себя другие группы и элементы.

Примером такого справочника является справочник, хранящий списки товаров, материалов, услуг. Обычно такой справочник носит название Номенклатура. Как правило, подобный справочник имеет множество групп – например – Товары – для хранения записей о товарно-материальных ценностях, и Услуги – для хранения списков услуг.

Второй вид иерархии – это иерархия элементов. Хорошим примером справочника, для которого естественна иерархия элементов, является справочник для хранения сведений о подразделениях организаций. Такой справочник обычно так и называется – Подразделения. Логика иерархии элементов заключается в том, что отдельные элементы справочника, описывающие отдельные подразделения организации, могут являться родителями для других элементов – так же подразделений. Например, подразделение Администрация вполне может включать в себя подразделения Бухгалтерия, Отдел кадров и так далее.

Справочники, в сущности, являются хранилищами аналитических признаков учета. Например, в случае со справочником Номенклатура можно организовать учет таким образом, чтобы иметь сведения об остатках каких-либо номенклатурных позиций на складе, об их стоимости. В случае со справочником, поддерживающим иерархию элементов, любой элемент справочника можно использовать как аналитический "разделитель" учета (то есть, например, выбирать данные элементы при заполнении табличных частей других объектов, например, документов). А вот при иерархии групп и элементов использовать в качестве аналитического разделителя группу не получится.

Если родительские отношения существуют внутри справочника, такие справочники называются иерархическими. Существует еще один вид взаимоотношений между справочниками, который называется подчинением. Предположим, у нас имеется

справочник Контрагенты, содержащий список организаций, с которыми наша организация имеет какие-то взаимоотношения. Каждый из контрагентов, описанных в этом справочнике, имеет некоторое количество контактных лиц – представителей контрагента. Можно сказать, что каждый представитель контрагента "принадлежит" определенному контрагенту. Такие взаимоотношения между справочниками реализуются при помощи механизма указания владельцев справочника и настройки подчинения.

Рассмотрим работу с иерархическими справочниками.

## **Иерархические справочники**

Создадим новый справочник **Единицы измерения**, зададим следующие его параметры:

**Имя:** ЕдиницыИзмерения

**Длина наименования:** 100 символов

**Подсистемы:** БухгалтерскийУчет, ОперативныйУчетМатериалов

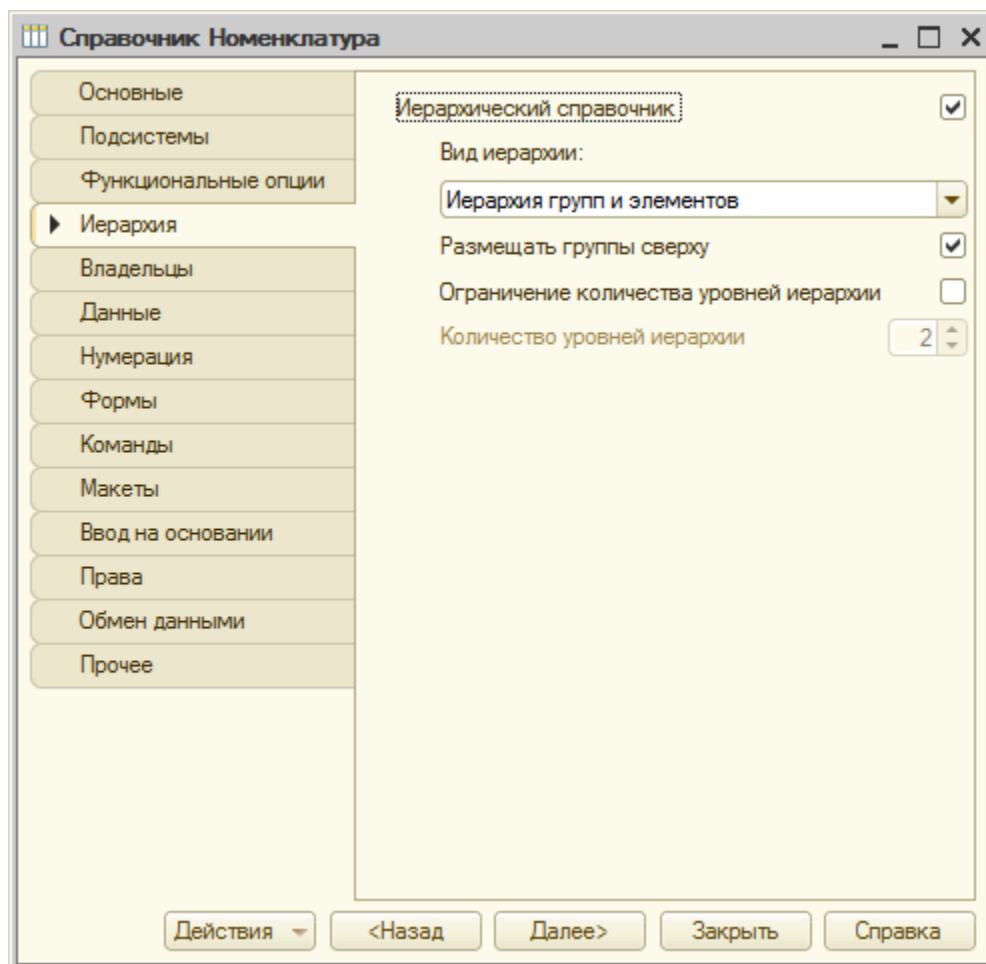
Это будет очень простой справочник, стандартный реквизит которого **Наименование** будет использоваться для хранения информации о наименовании единицы измерения.

Теперь создадим очередной справочник – **Номенклатура**. Зададим следующие параметры:

**Имя:** Номенклатура

**Подсистемы:** БухгалтерскийУчет, ОперативныйУчетМатериалов

На вкладке окна редактирования объекта **Иерархия**, рис. 4.1., установим следующие параметры:



**Рис. 4.1.** Настройка иерархического справочника

**Иерархический справочник:** Установлено

**Вид иерархии:** Иерархия групп и элементов.

Этот параметр может принимать значение **Иерархия элементов**. В нашем случае справочник сможет содержать отдельные элементы, собранные, в зависимости от их вида, в группы. Эту структуру можно сравнить с папками и файлами в файловой системе компьютера. Группы – это папки, отдельные элементы – это файлы.

На вкладке **Данные**, рис. 4.2., добавим следующие реквизиты:

**ЕдиницаИзмерения:** Тип СправочникСсылка.ЕдиницыИзмерения.

**Услуга:** Тип Булево, **Использование:** Для группы и элемента. Эта установка позволит задавать данный реквизит и для элементов и для групп.

**Заполнять из данных заполнения:** Истина

Отдельные группы нашего справочника планируется использовать для хранения исключительно услуг, и подобная установка (в частности, истинность параметра **Заполнять из данных заполнения**) позволит нам реализовать автоматический механизм заполнения данного реквизита для элементов, входящих в группы.

Длина наименования: 100

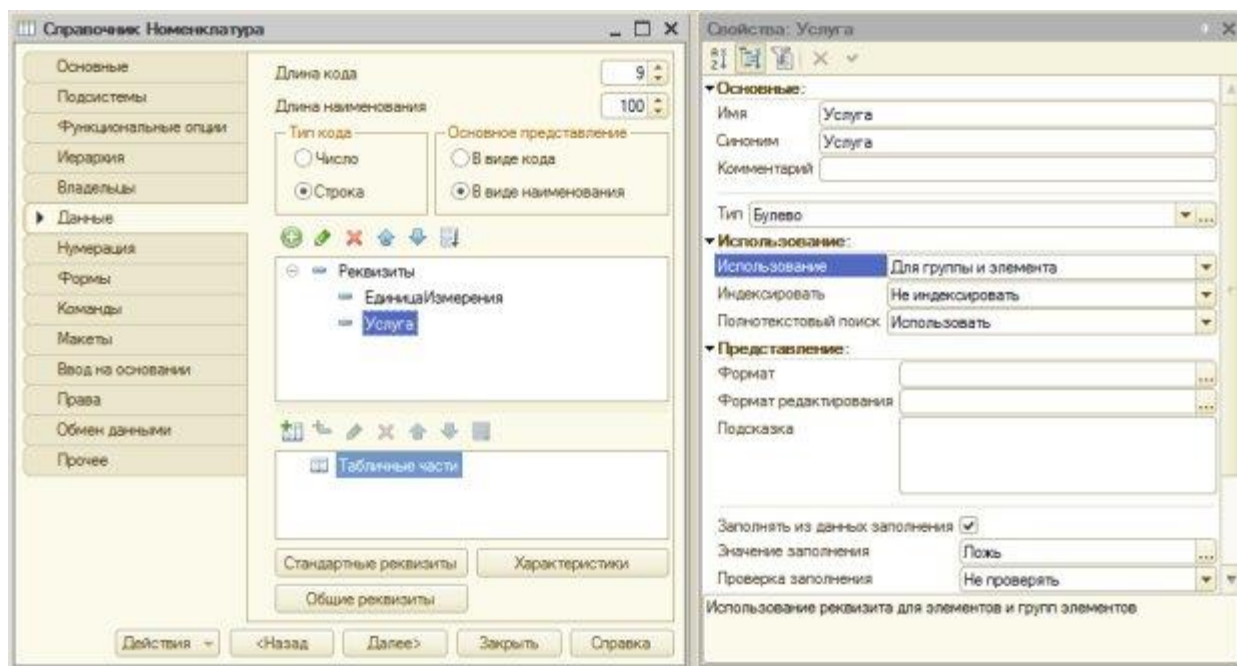


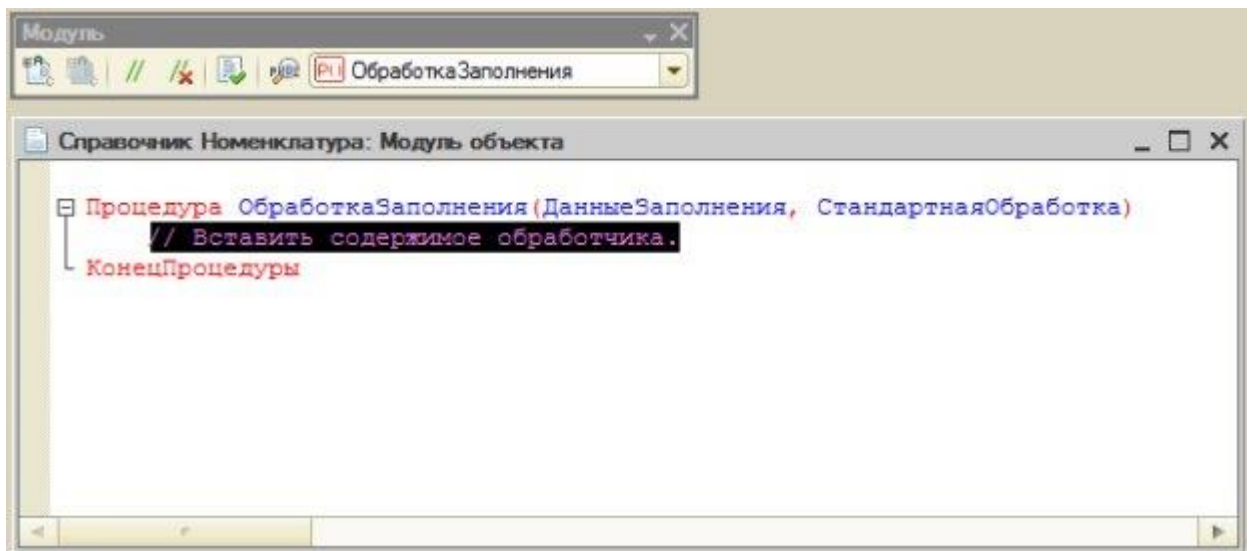
Рис. 4.2. Состав реквизитов справочника Номенклатура

Таким образом, при создании элемента справочника мы будем задавать название элемента в стандартном реквизите **Наименование**, указывать единицу измерения, а так же, для услуг, устанавливать флаг **Услуга**, причем, установка этого флага для группы будет означать, что в ней хранятся списки услуг, а для элемента – то, что он является услугой.

### Автозаполнение реквизитов

Реализуем функцию автоматического заполнения реквизита **Услуга** для элементов, входящих в группы. Нам нужно, чтобы элемент, создаваемый в группе с установленным флагом **Услуга**, при его создании, автоматически бы получал установленный флаг **Услуга**, соответственно, если данный флаг у группы не установлен, у элемента он так же не должен быть установлен. При этом нам нужно предусмотреть ситуацию, когда элемент создается вне группы – на верхнем уровне справочника **Номенклатура**. Для решения этой задачи мы можем воспользоваться обработчиком события **ОбработкаЗаполнения**, его процедура располагается в модуле объекта.

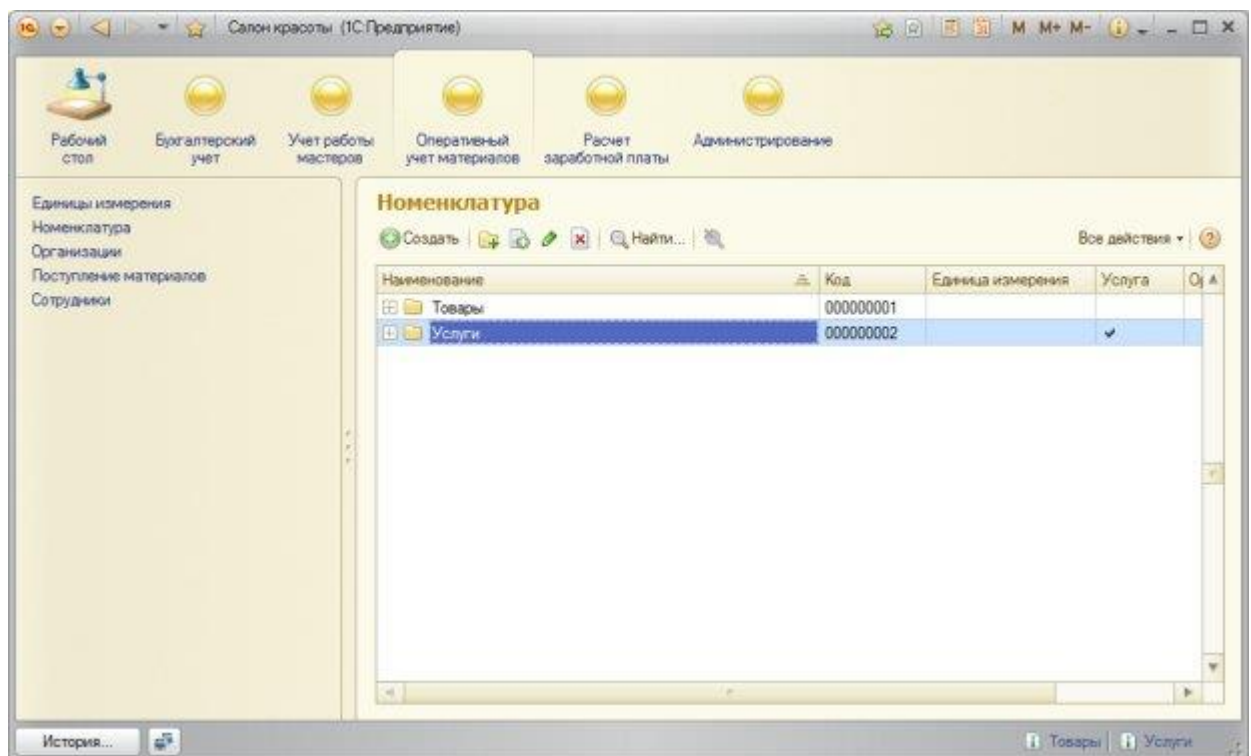
Перейдем в модуль объекта (кнопка **Модуль объекта** на закладке **Прочие** окна редактирования объекта), из списка процедур и выберем **ОбработкаЗаполнения**, рис. 4.3.



**Рис. 4.3.** Процедура обработки заполнения справочника

Процедура будет исполняться на стороне сервера, причем, вызываться она будет при различных способах создания элемента справочника – например, при интерактивном создании пользователем, при копировании, при программном создании. Параметр процедуры **СтандартнаяОбработка** позволяет включать или отключать стандартную обработку процесса заполнения реквизитов, параметр **ДанныеЗаполнения** содержит данные, которые система использует для заполнения элемента.

В режиме 1С:Предприятие откроем справочник **Номенклатура**, создадим две группы – **Товары** – флаг **Услуги** в этой группе не устанавливаем, и **Услуги** – флаг установлен, рис. 4.4.



**Рис. 4.4.** Две группы в справочнике Номенклатура

В модуль формы справочника Номенклатура добавим следующий код:

```
Процедура ОбработкаЗаполнения(ДанныеЗаполнения, СтандартнаяОбработка)  
Если ДанныеЗаполнения<>Неопределено Тогда  
    Если ДанныеЗаполнения.Свойство("Родитель") Тогда  
        ДанныеЗаполнения.Вставить("Услуга", ДанныеЗаполнения.Родитель.Услуга);  
    КонецЕсли;  
КонецЕсли;  
КонецПроцедуры
```

Рассмотрим теперь пример работы с подчиненными справочниками.

## Подчиненные справочники

Создадим новый справочник, назовем его **Контрагенты**.

Добавим его в подсистемы **БухгалтерскийУчет** и **ОперативныйУчетМатериалов**.

Справочник будет иерархическим, с иерархией групп и элементов.

В состав реквизитов справочника добавим следующие (рис. 4.12.):

**Имя:** ПолноеНаименование, тип – Строка, длина 100.

**Имя:** КонтактныеСведения, тип – Строка, длина 100

**Рис. 4.12.** Справочник Контрагенты

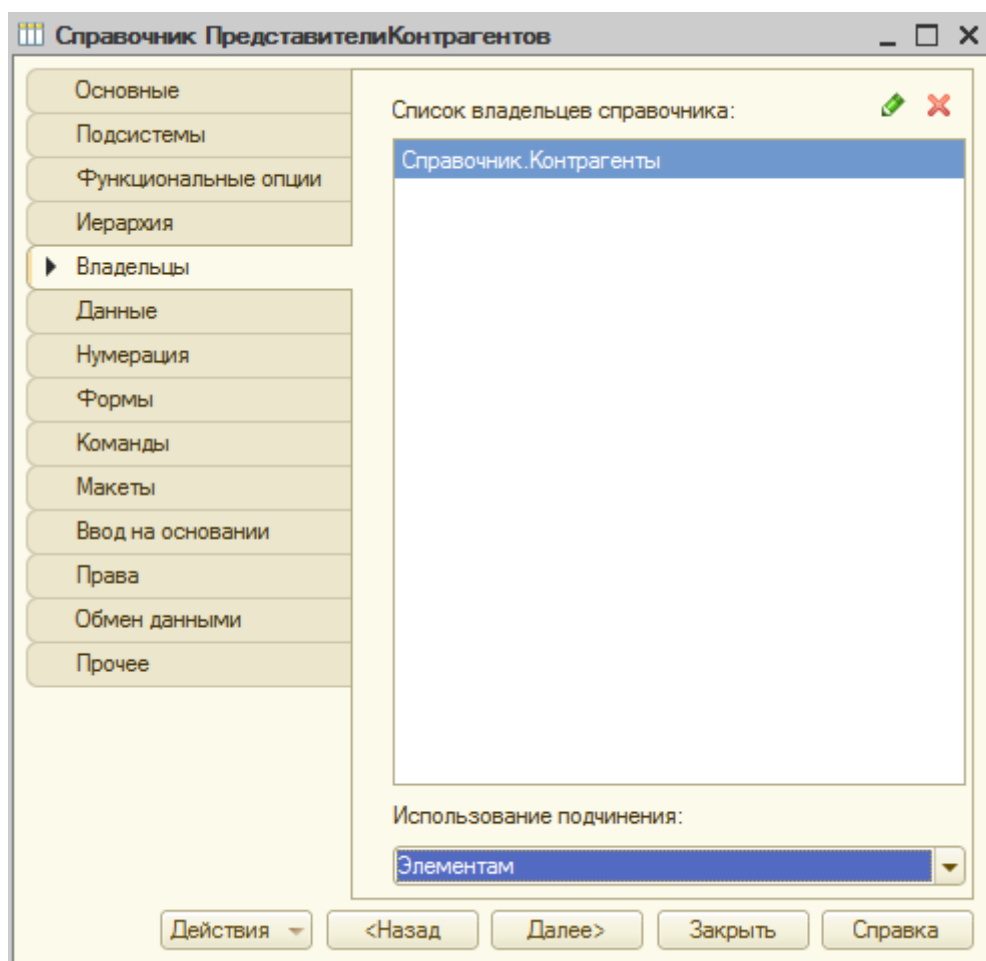
Создадим еще один справочник. Назовем его **ПредставителиКонтрагентов**. Главная черта этого справочника – то, что он подчинен справочнику **Контрагенты**. Для настройки подчинения используется вкладка окна настройки объекта конфигурации **Владельцы**. Здесь мы должны добавить в **Список владельцев справочника** справочники-владельцы, в нашем случае – справочник **Контрагенты**. После того, как владелец добавлен в этот список, мы можем настроить для него параметр **Использование подчинения**. Здесь возможны три варианта:

**Элементам** – элементы подчиненного справочника подчинены элементам справочника-владельца. В нашем случае выбор этого параметра означает, что в качестве "владельца" представителя контрагента выступает сам контрагент.

**Группам** – подчинение группам справочника-владельца. Нас это не устроит – так как группы справочника будут содержать контрагентов, сгруппированных по достаточно общим показателям (Покупатели, Заказчики и т.д.), и сопоставление одного контактного лица нескольким разным, например, покупателям, смысла не имеет. В другой ситуации эта установка могла бы быть вполне оправданной.

**Группам и элементам** – подчинение как группам, так и элементам справочника-владельца.

Мы укажем в параметре **Использование подчинения** вариант **Элементам**, рис. 4.13.



**Рис. 4.13.** Настройка подчинения

Добавим справочник в состав подсистем **БухгалтерскийУчет** и **ОперативныйУчетМатериалов**.

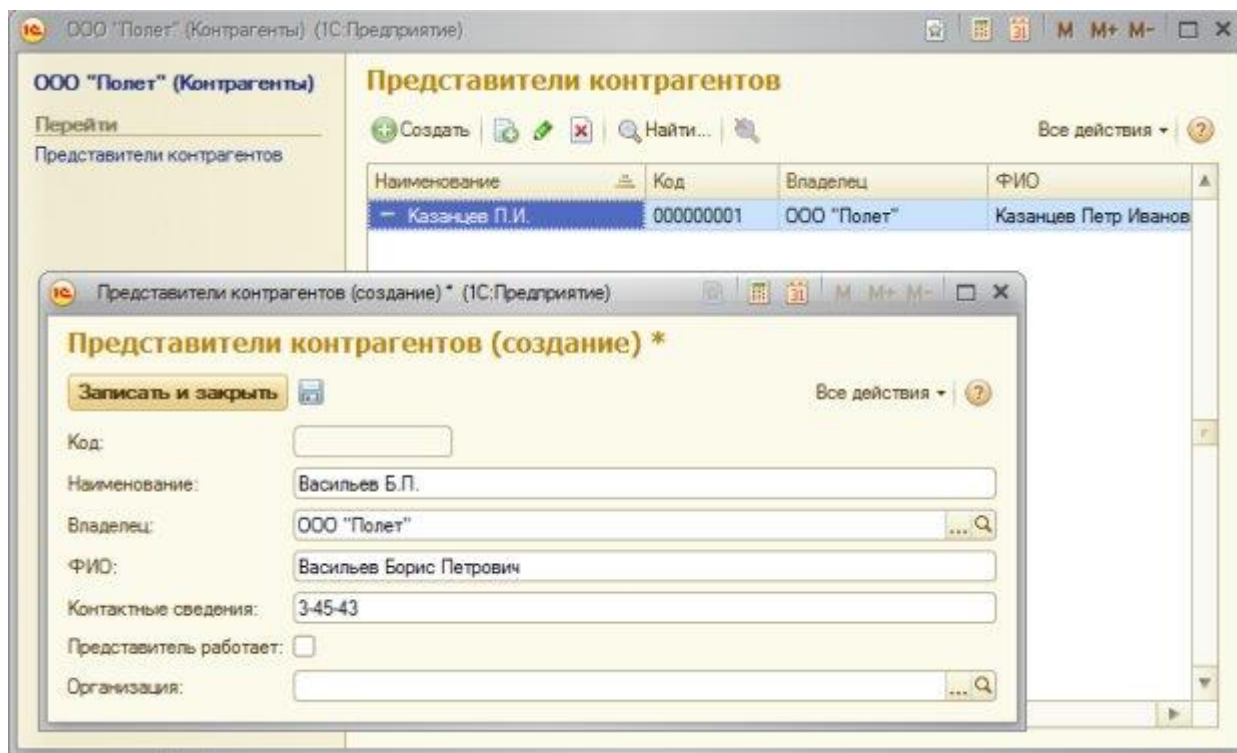
В состав реквизитов справочника добавим следующие:

**Имя:** ФИО, тип – Строка, длина 100

**Имя:** КонтактныеСведения, тип – Строка, длина 100.

**Имя:** ПредставительРаботает, тип – Булево.

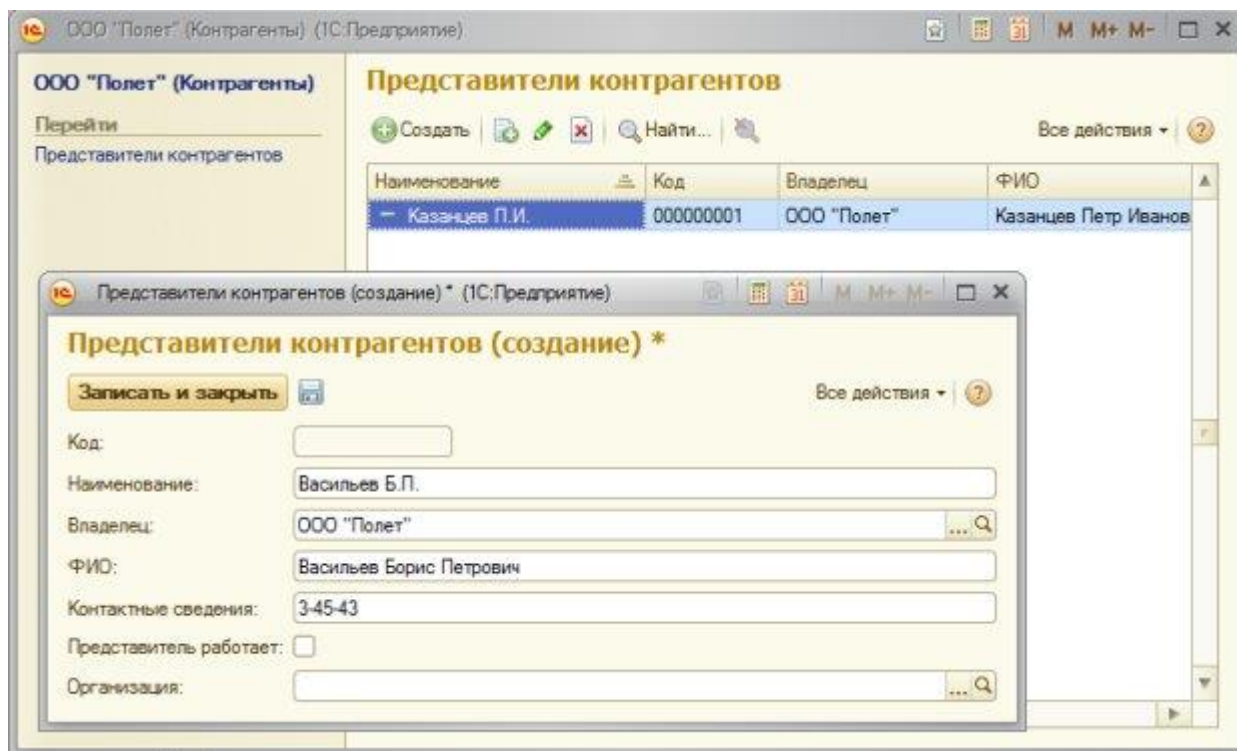
Посмотрим теперь, как выглядит работа с созданными справочниками в режиме **1С:Предприятие**. Особенность здесь заключается в том, что, открывая карточку контрагента, в левой ее части мы видим область **Перейти**, где можно найти ссылку для перехода в справочник **ПредставителиКонтрагентов**, рис. 4.14.



**Рис. 4.14.** Форма элемента справочника Контрагенты

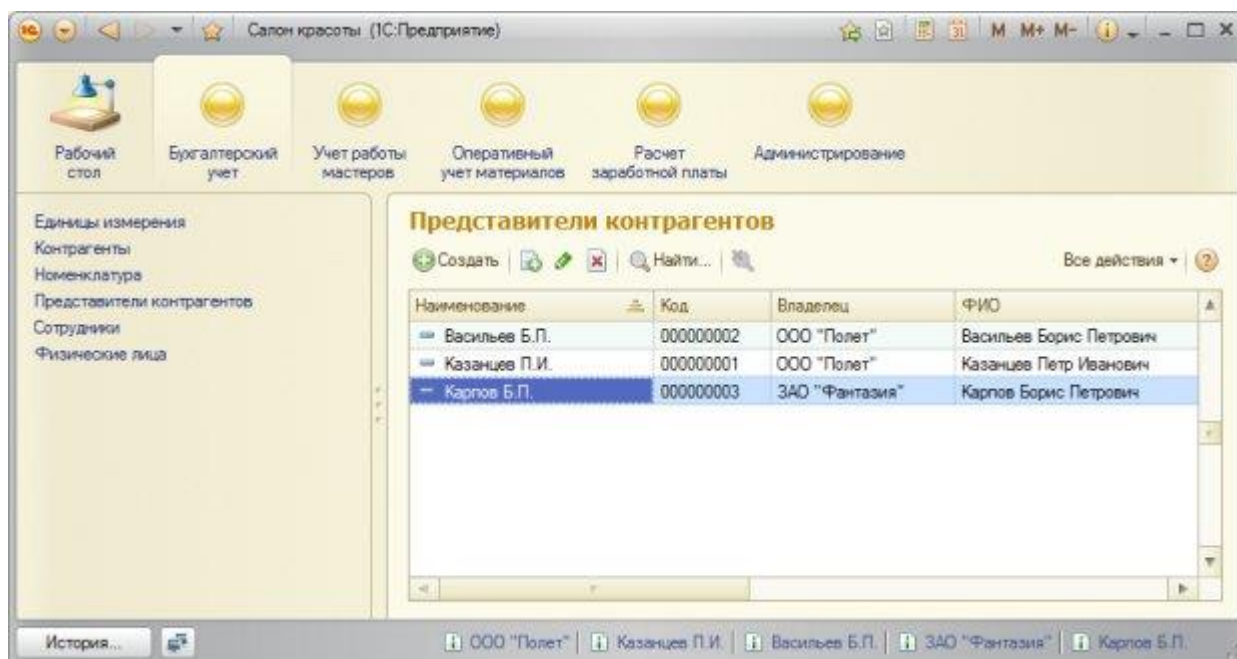
При переходе в этот справочник мы будем видеть в открывшемся окне лишь тех представителей, которые относятся к контрагенту, с которым мы в данный момент работаем. При создании новой записи о представителе он автоматически будет "привязываться" к тому контрагенту (поле владелец будет заполнено должным образом), из формы элемента которого мы перешли в справочник **ПредставителиКонтрагентов**. В форме списка справочника будет отображаться ссылка для перехода к форме элемента справочника-владельца, рис. 4.15.





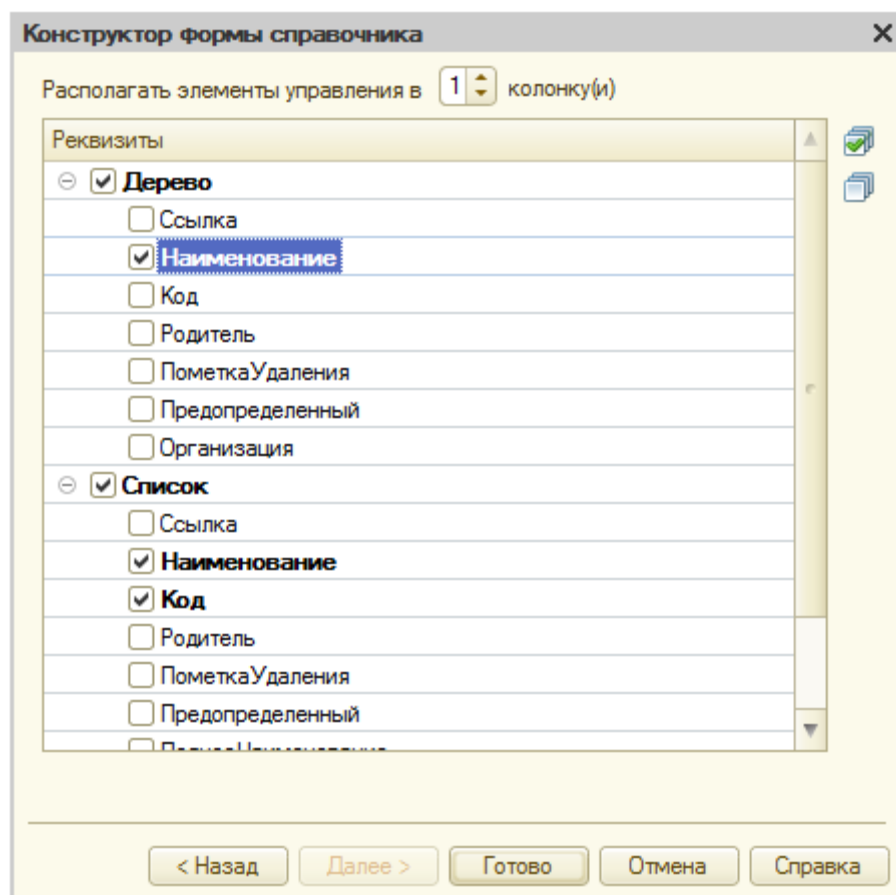
**Рис. 4.15.** Формы списка и элемента справочника ПредставителиКонтрагентов

Мы можем создавать элементы справочника **ПредставителиКонтрагентов** и непосредственно перейдя в него, тогда нам придется самостоятельно указывать его владельца – элемент справочника **Контрагенты**. При переходе в подчиненный справочник не из формы элемента справочника-владельца, мы можем просматривать все его элементы, рис. 4.16.



**Рис. 4.16.** Просмотр формы списка справочника ПредставителиКонтрагентов

Перейдем в окно редактирования объекта конфигурации справочника **Контрагенты**, перейдем на его закладку **Формы**, создадим новую форму списка. При работе с конструктором форм можно заметить, что на закладке управления реквизитами присутствуют два элемента – **Дерево** и **Список**. Список мы с вами уже видели, а элемент **Дерево** характерен для иерархических справочников, он позволяет облегчить навигацию по большим справочникам, выводя их иерархическую структуру в дополнение к списку. Установим флаг в поле **Дерево**, из списка реквизитов, отображаемых в дереве элементов, выберем **Наименование**, рис. 4.17.



**Рис. 4.17.** Конструктор формы справочника Контрагенты

Вот как будет выглядеть форма списка справочника в режиме 1С:Предприятие, 4.18.

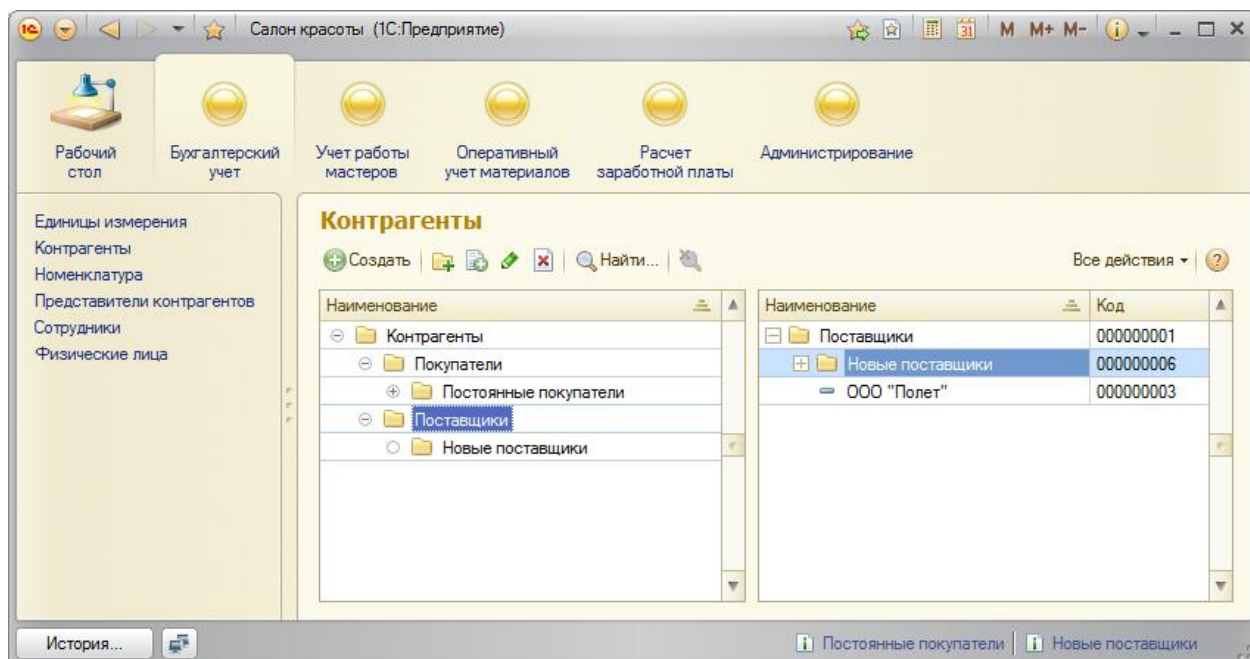


Рис. 4.18. Форма справочника со списком и деревом элементов

## Проверка заполнения реквизита справочника, фильтрация

Расширим справочник **Контрагенты**, добавим в состав его реквизитов еще один – назовем его **ОсновноеКонтактноеЛицо**, тип – **СправочникСсылка.ПредставителиКонтрагентов**. Смысл этого поля заключается в хранении ссылки на представителя контрагента, который является "основным" для данного контрагента. Если нужно связаться с контрагентом, можно открыть его карточку и тут же увидеть, какой представитель является основным. Создадим форму элемента справочника **Контрагенты** и посмотрим на нее, попытавшись установить новое поле – **Основное контактное лицо**, рис. 4.18.

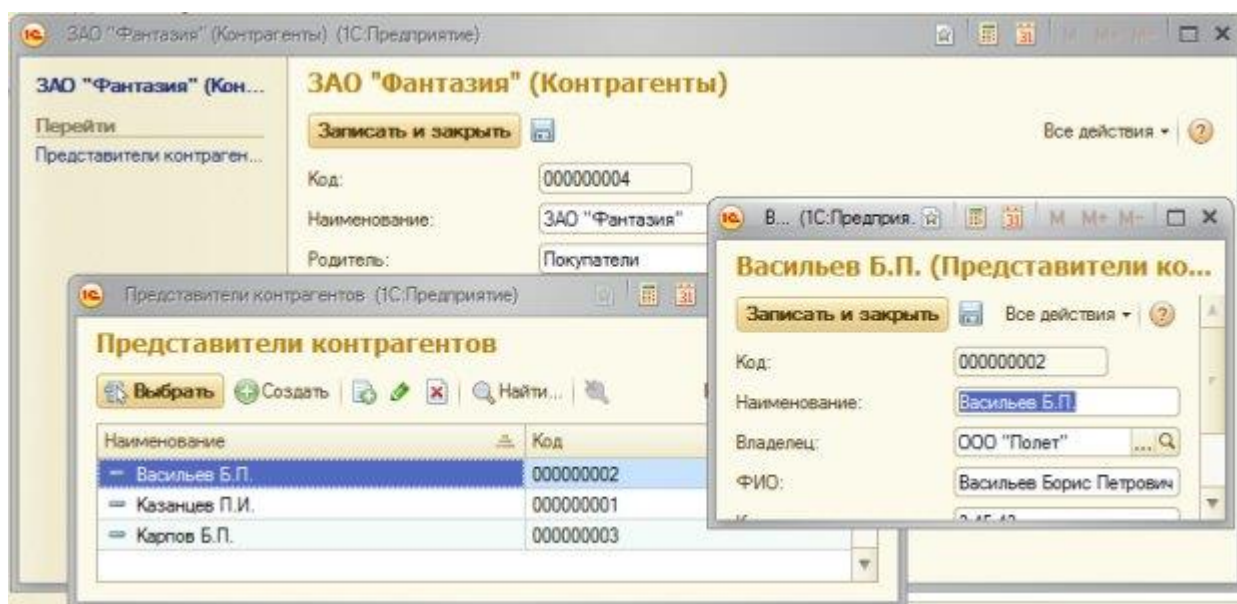


Рис. 4.19. Попытка заполнения реквизита Основное контактное лицо

Видно, что при попытке подбора элемента в данное поле нам показывают не только те элементы справочника **ПредставителиКонтрагентов**, владельцем которых является редактируемый элемент, но и все остальные. Так работать неудобно – это значит, что нам нужно настроить фильтрацию выводимых элементов. Для того, чтобы это сделать, удобнее всего будет воспользоваться свойством **Связи параметров выбора** реквизита **ОсновноеКонтактноеЛицо**. Для открытия палитры свойств реквизита мы можем сделать двойной щелчок по реквизиту в окне редактирования *формы*.

В открывшейся палитре свойств найдем свойство **Связи параметров выбора** и нажмем на кнопку с тремя точками около этого поля. Появится окно **Связи параметров выбора**, рис. 4.20.

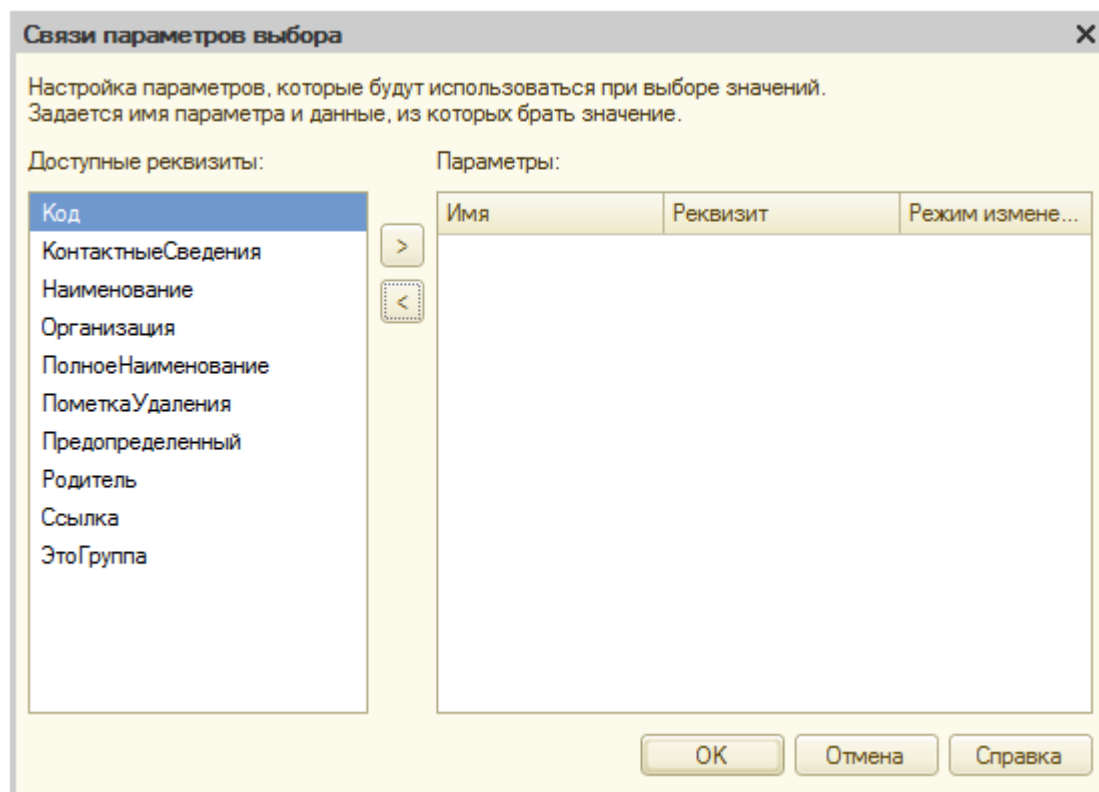
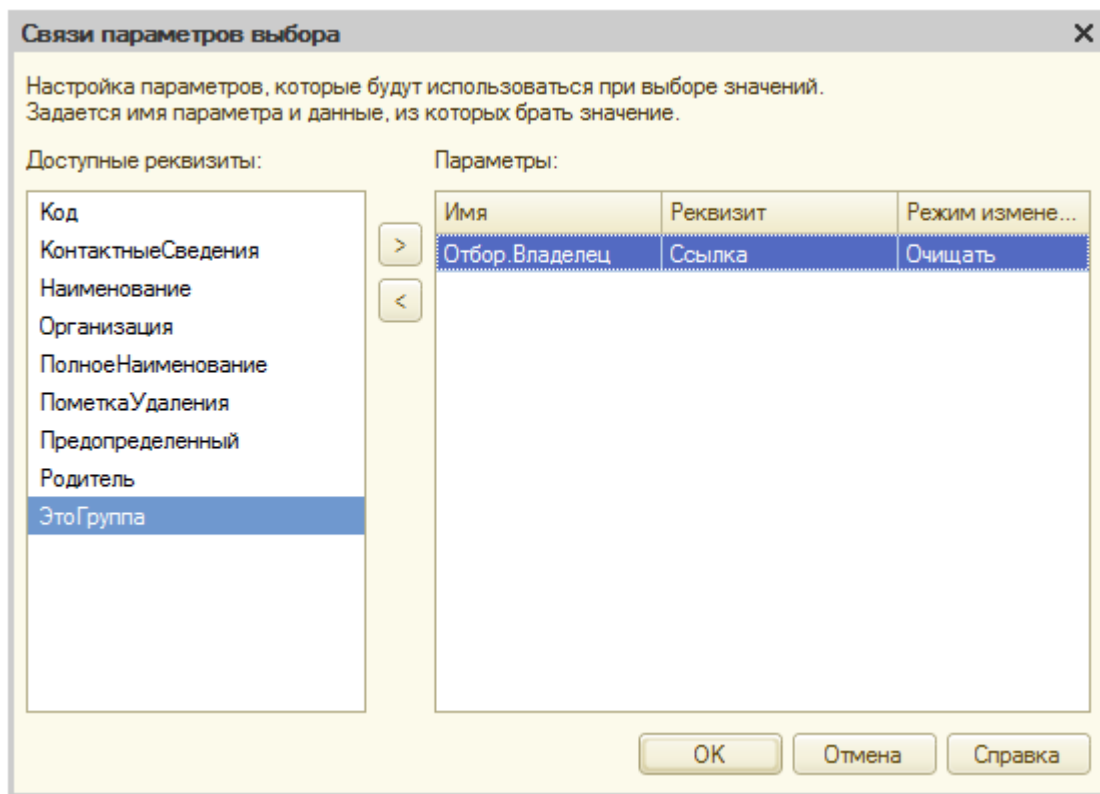


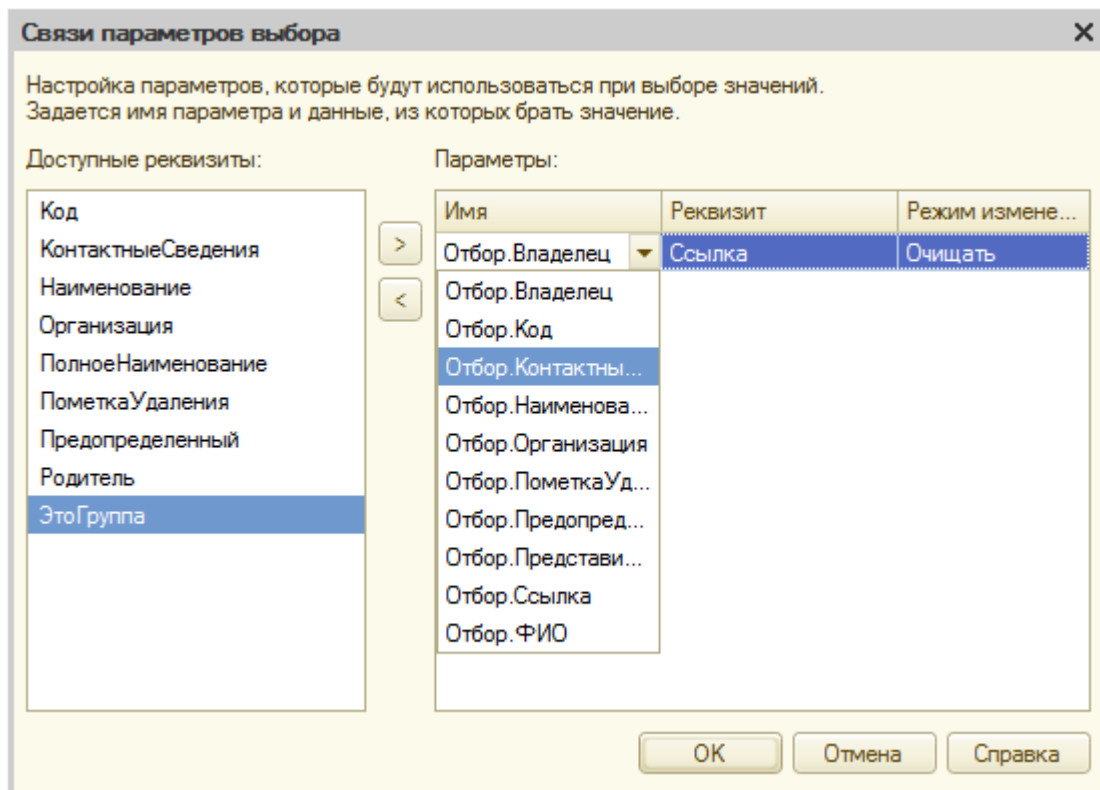
Рис. 4.20. Окно Связи параметров выбора

В левой части окна можно видеть доступные реквизиты (это реквизиты открытого элемента справочника **Контрагенты**), в правом – параметры, влияющие на отбор элементов в появляющемся окне выбора элементов при заполнении поля представителя контрагента. Выделим реквизит **Ссылка** и нажмем на кнопку **Добавить выбранный реквизит в параметры выбора** (она находится между полями). Окно примет следующий вид, рис. 4.21.



**Рис. 4.21.** Окно Связи параметров выбора с настроенным параметром

В данном случае в строке области **Параметры** отображается как раз то, что нам нужно – нам нужно, чтобы отбор в раскрывающемся списке происходил по владельцу, а именно – по текущему открытому элементу справочника **Контрагенты**, на который и указывает реквизит **Ссылка**. В поле имя можно выбрать другие варианты отбора, рис. 4.21.





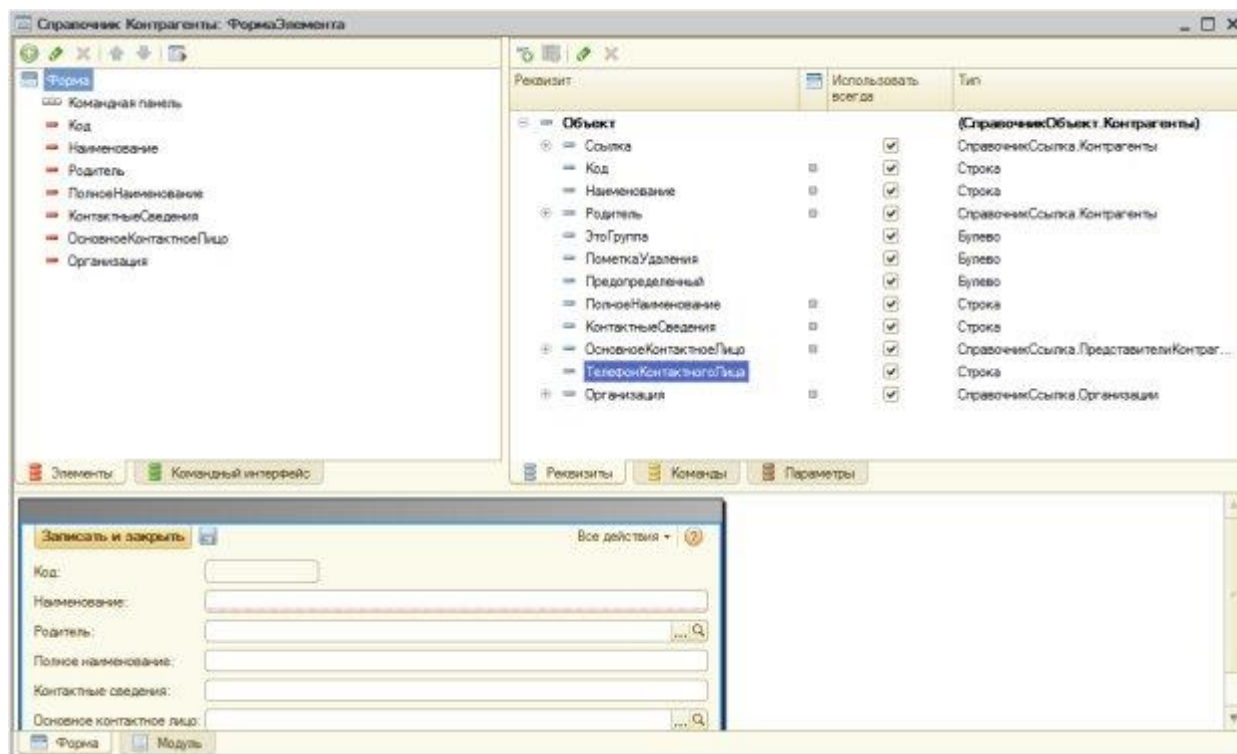
**Рис. 4.22.** Настройки в окне Связи параметров выбора

После того, как эта настройка выполнена, мы можем нажать **ОК** в окне **Связи параметров выбора** и проверить функциональность решения – при заполнении поля **ОсновноеКонтактноеЛицо** список выбора ограничивается подчиненными элементами.

Добавим в справочник **Контрагенты** еще один реквизит – **ТелефонКонтактногоЛица**. Зададим тип – **Строка**, длина – **100**. Этот реквизит соответствует реквизиту **КонтактныеСведения** справочника **ПредставителиКонтрагентов**. Он нужен нам исключительно для удобства – для того, чтобы, когда в форме контрагента указано основное контактное лицо, пользователю не пришлось бы, для поиска телефона контактного лица, заглядывать в его карточку.

После добавления реквизита в справочник **Контрагенты**, запустим режим **1С:Предприятие** и откроем форму одного из элементов этого справочника. Если присмотреться к этой форме на данном этапе работы, окажется, что реквизита **ТелефонКонтактногоЛица** на ней не наблюдается. Все дело в том, что, создав собственную форму элемента для справочника, мы отказываемся от автоматического механизма создания форм, который, если бы не наша, самостоятельно созданная ранее форма, автоматически построил бы форму с новым реквизитом.

Добавим элемент управления для реквизита **ТелефонКонтактногоЛица** на форму. Откроем форму элемента справочника **Контрагенты** для редактирования и перетащим реквизит **ТелефонКонтактногоЛица** с вкладки **Реквизиты** на вкладку **Элементы**, рис. 4.23.



**Рис. 4.23.** Реквизит **ТелефонКонтактногоЛица** нужно переместить со вкладки **Реквизиты** на вкладку **Элементы**

В справочнике **ПредставителиКонтрагентов** есть реквизит, который указывает на то, что представитель контрагента работает в организации-контрагенте – это реквизит логического типа **ПредставительРаботает**. Нам нужно реализовать следующий функционал. Если пользователь, заполняя карточку элемента справочника **Контрагенты** выбирает в качестве реквизита **ОсновноеКонтактноеЛицо** сотрудника, у которого флаг **ПредставительРаботает** не установлен – мы предупреждаем пользователя об этом, выводя сообщение.

Для этого нам понадобится перехватить событие изменения поля **ОсновноеКонтактноеЛицо**, после чего проверить, установлен ли у выбранного контактного лица флаг **ПредставительРаботает**, и если такой флаг не установлен – вывести сообщение пользователю.

Из контекстного меню элемента формы **ОсновноеКонтактноеЛицо** выберем событие **ПриИзменении**, откроется редактор кода, в котором уже будет создана пустая клиентская процедура для перехвата этого события. К этому моменту реквизит **ОсновноеКонтактноеЛицо** уже будет содержать выбранного представителя контрагента. Нам понадобится серверная процедура, которая обратится к реквизиту этого представителя **ПредставительРаботает** и вернет нам его значение. После того, как мы получим с сервера сведения о том, работает ли представитель, мы примем решение – выводить ли пользователю сообщение или нет.

Все это реализовано с помощью нижеприведенного кода:

```
&НаКлиенте
Процедура ОсновноеКонтактноеЛицоПриИзменении (Элемент)
Если НЕ ПроверитьЗаполнениеРеквизита ( ) Тогда
    Сообщить ("Выбранное контактное лицо, "+Объект.ОсновноеКонтактноеЛицо+", не
    работает у контрагента.");
КонецЕсли;
КонецПроцедуры

&НаСервере
Функция ПроверитьЗаполнениеРеквизита ( )
    Возврат (Объект.ОсновноеКонтактноеЛицо.ПредставительРаботает) ;
КонецФункции
```

Кроме того, зададим автоматический механизм переноса в реквизит **ТелефонКонтактногоЛица** сведений из элемента справочника **ПредставителиКонтрагентов**, который указан в поле **ОсновноеКонтактноеЛицо**. В частности, дополним обработчик события для поля **ОсновноеКонтактноеЛицо** **ПриИзменении()**:

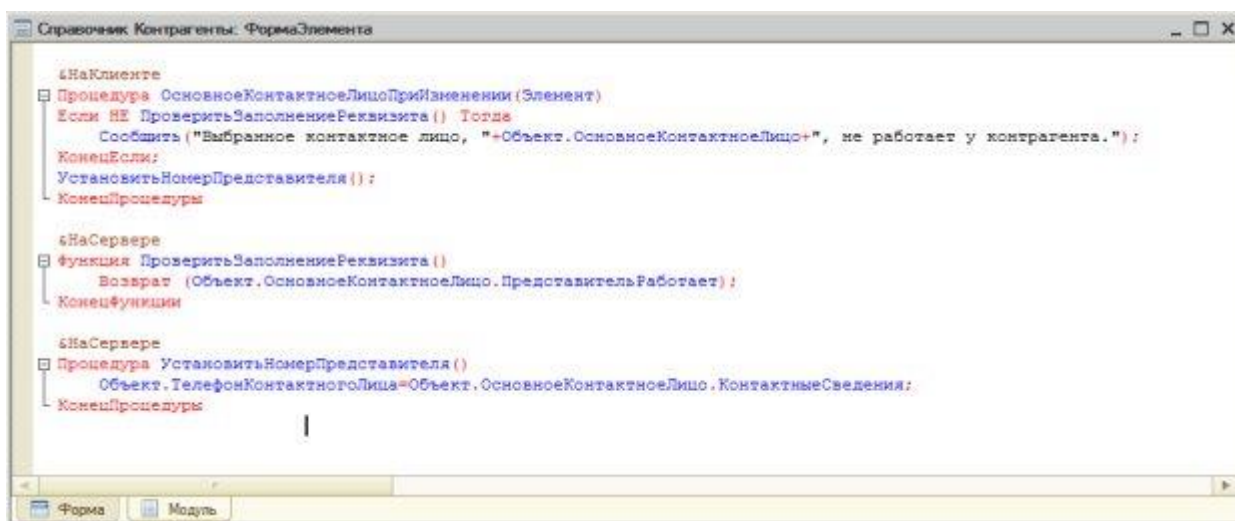
Подготовим серверную процедуру, которая будет работать с реквизитами объекта, она будет иметь следующий вид:

```
&НаСервере
Процедура УстановитьНомерПредставителя ( )

Объект.ТелефонКонтактногоЛица=Объект.ОсновноеКонтактноеЛицо.КонтактныеСведени
я;
КонецПроцедуры
```

Вызовем эту процедуру в уже существующем обработчике **ПриИзменении**, код модуля приобретет вид, показанный на рис. 4.24.

Кроме того, в окне его свойств отредактируем свойство **Вид** – выберем его значение **Поле надписи**. Пользователь не будет ничего в это поле вводить самостоятельно, поэтому поле надписи нас вполне устроит.



**Рис. 4.24.** Код модуля формы элемента справочника Контрагенты

При выборе "неподходящего" представителя окно элемента справочника Контрагенты примет следующий вид, рис. 4.25.



**Рис. 4.25.** Сообщение о выборе неподходящего контактного лица

Предположим, нам нужно, чтобы полное наименование организации было бы не короче 5-ти символов.

В предыдущей лекции, работая с формами, мы рассматривали проверку заполнения реквизитов справочника на стороне клиента. Такой подход возможен, но если говорить о производительности решения, предполагая работу конфигурации на сравнительно маломощном клиенте, лучше выполнять подобные проверки на стороне сервера. Как мы уже говорили, для проверки заполненности реквизита можно использовать стандартное свойство **Проверка заполнения**. Но если вам нужны более сложные проверки, если их достаточно много (например, при проверке производится манипулирование с табличными частями, получение каких-то дополнительных данных из базы, при этом алгоритмы достаточно сложны), лучше всего для проверки заполнения реквизитов воспользоваться



процедурой **ОбработкаПроверкиЗаполнения ()**, которая объявляется на уровне модуля объекта.

Объявим эту процедуру, открыв модуль объекта (закладка **Прочее** окна редактирования свойств объекта, кнопка **Модуль объекта**) выбором из списка **Процедуры и функции панели инструментов** модуль процедуры **ОбработкаПроверкиЗаполнения()**.

Эта процедура работает на сервере, мы можем напрямую обращаться к реквизитам объекта.

```
Процедура ОбработкаПроверкиЗаполнения(Отказ, ПроверяемыеРеквизиты)
  Если СтрДлина(ПолноеНаименование) < 5 Тогда
    Отказ=Истина;
    Сообщить ("Полное наименование организации должно быть не короче 5-ти
символов");
  КонецЕсли;
КонецПроцедуры
```

Передаваемый в процедуру параметр **Отказ** можно установить в значение **Истина** для того, чтобы показать, что проверка не пройдена. **ПроверяемыеРеквизиты** – это массив, он содержит реквизиты для автоматической проверки, в частности, там находятся те реквизиты, для которых включена автоматическая проверка заполнения. По умолчанию это – реквизит **Наименование**. В этом можно убедиться, просмотрев содержимое переменной в отладчике.