

Лабораторная работа №2

ПЛАН

По дисциплине: Корпоративные информационные системы
Тема занятия: Подсистемы. Справочники. Документы.
Механизм основных форм
Цель занятия: Создание подсистем, справочников и документов
информационной базы, изучение механизмов основных форм
Количество часов 2

СОДЕРЖАНИЕ РАБОТЫ

Теоретическая часть

Подсистемы

Подсистемы - это основные элементы для построения интерфейса 1С:Предприятия. Поэтому первое, с чего следует начинать разработку конфигурации, - это проектирование состава подсистем.

При этом перед разработчиком стоит важная и ответственная задача - тщательно продумать состав подсистем, и затем аккуратно и осмысленно привязать к подсистемам те объекты конфигурации, которые он будет создавать.



Рис. 2.1. Подсистемы

Справочники

Объект конфигурации Справочник предназначен для работы со списками данных. Свойства и структура этих списков описываются в объектах конфигурации Справочник, на основе которых платформа создает в базе данных таблицы для хранения информации из этих справочников.

Справочник состоит из элементов. Пользователь в процессе работы может самостоятельно добавлять новые элементы в справочник.

В базе данных каждый элемент справочника представляет собой отдельную запись в основной таблице, хранящей информацию из этого справочника.

Каждый элемент справочника, как правило, содержит некоторую дополнительную информацию, которая подробнее описывает этот элемент. Поскольку эти объекты конфигурации логически связаны с объектом Справочник, они называются подчиненными этому объекту.

В БД справочник хранится в виде таблицы, в строках которой расположены элементы списка, а каждому реквизиту (стандартному или созданному разработчиком) в этой таблице соответствует отдельный столбец.

Система скрывает от разработчика всю «техническую» часть, связанную с хранением данных: в базе данных для справочника создаются несколько таблиц, эти таблицы связываются по уникальному полю (**Ссылка**), поля таблиц имеют определенные типы и т.д. Все это система делает сама. Нам лишь нужно добавить в объект конфигурации Справочник подчиненный ему объект **«Табличная часть»**.

Для удобства использования элементы справочника могут быть сгруппированы по какому-либо принципу. Группировка называется иерархией, выделяют два вида иерархии:

- иерархия групп и элементов (*в качестве родителя выступает группа элементов справочника*)
- иерархия элементов (родитель один из элементов справочника)

Элементы одного справочника могут быть подчинены элементам или группам другого справочника. 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. Объект конфигурации Справочник позволяет описать любое количество таких элементов справочника. Они называются предопределенными элементами справочника.

Предопределенные элементы отличаются от обычных тем, что они создаются в конфигураторе и что пользователь не может их удалить. Все остальные действия с ними он делать может, в том числе и переименовывать. В интерфейсе предопределенные элементы справочника помечены специальной пиктограммой.

Предопределенные элементы

Выделяют следующие 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.

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.

Основная конфигурация и конфигурация базы данных

Конфигурация, предназначенная для разработчика, называется Основная конфигурация. 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. Хранилище конфигурации содержит конфигурацию, предназначенную для групповой разработки. Она хранится не в виде единой конфигурации, а в виде отдельных объектов в разрезе версий конфигурации. Таким образом, мы можем получить из хранилища конфигурацию любой версии - для этого она «собирается» из объектов нужной версии.

Теперь представьте, что между всеми этими видами конфигураций существует возможность сравнения и обновления. В этом случае очень легко запутаться, и название Основная конфигурация как нельзя лучше отражает конечную цель всех изменений.

*Если эти изменения не затрагивают структуру базы данных (например, если не нужно изменять таблицы, если поменялся только программный код в каком-то модуле), то обновить конфигурацию базы данных можно не прерывая работы пользователей - **динамическое обновление**.*

Но если изменения касаются структуры базы данных, например, добавился новый реквизит у справочника или изменился тип существующего реквизита, то тогда нужно 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.

Формы справочника

В зависимости от того, какие действия мы хотим выполнять со справочником, нам требуется изображать справочник в «разном виде».

Система может самостоятельно сгенерировать все формы, которые нужны для представления данных, содержащихся в справочнике. Причем система «знает», какие именно формы нужно использовать в каких ситуациях.

Вообще говоря, для отображения справочника в различных ситуациях требуется максимум пять форм для справочника.

Таблица 3.1. Формы справочника

В контекстном меню и в палитре свойств (Рис.3.3)	В конструкторе форм (Рис. 3.2)	На закладке формы (Формы...) (Рис.3.1)
Форма объекта	Форма элемента справочника	Элемента
Форма группы	Форма группы справочника	Группы
Форма списка	Форма списка справочника	Списка
Форма для выбора	Форма выбора справочника	Выбора
Форма для выбора группы	Форма выбора группы справочника	Выбора группы

Обратите внимание, что в различных местах конфигуратора одни и те же формы называются немного по-разному. Дело в том, что в контекстном меню и палитре свойств отображаются свойства объектов конфигурации. Они одинаковые для

всех объектов конфигурации. А в конструкторе форм и на закладке формы отображаются представления этих свойств в более понятном виде.

Рис.3.1 Формы

Рис.3.2 Форма элемента справочника

Рис.3.3 Основная форма объекта

«Форма элемента» используется для редактирования или создания элемента справочника.

«Форма группы» используется для редактирования или создания группы справочника. Группа, как правило, содержит гораздо меньше информации, чем сам элемент справочника. Поэтому для нее нужна отдельная форма, отличная от формы элемента.

«Форма списка» используется для отображения списка элементов справочника.

«Форма выбора» используется для того, чтобы в поле некоторой формы выбрать один из элементов справочника. При этом форма выбора проще, чем форма списка, так как в форме списка может показываться много реквизитов. А при выборе элемента (в документе, например), нам нужно знать только наименование. Поэтому можно для выбора использовать отдельную, более простую форму.

«Форма выбора группы» используется, когда в поле некоторой формы нужно выбрать не просто элемент справочника, а одну из его групп. При этом форма выбора группы проще, чем форма выбора элемента, так как группа, как правило, содержит гораздо меньше информации, чем сам элемент справочника.

Любая форма может быть описана в конфигураторе. Для создания такого описания существует подчиненный объект конфигурации Форма, она подчинена одному из прикладных объектов, но может существовать и самостоятельно.

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

Документы

Объект конфигурации Документ предназначен для описания информации о совершенных хозяйственных операциях или о событиях, произошедших в жизни организации вообще. Как правило, в работе любой фирмы используются такие документы, как приходные накладные, приказы о приеме на работу, платежные поручения, счета и т.д. Свойства и структура этих документов описываются в объектах конфигурации Документ, на основе которых платформа создает в базе данных таблицы для хранения информации из этих документов.

Логика работы документов отличается от логики работы других объектов конфигурации. 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, вступят в силу и состояние учета будет изменено.

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

Каждый документ содержит информацию, которая подробнее описывает этот документ. 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 разработчик создает самостоятельно, однако у каждого объекта Документ по умолчанию существуют стандартные реквизиты. Два наиболее важных из них - это Дата и Номер. Для описания конкретной информации служат табличные части объекта Документ. В этом случае в базе данных будут созданы дополнительные таблицы для хранения табличных частей, подчиненных конкретному документу.

Формы документа

Для «визуализации» документа существует несколько основных форм, которые имеют несколько вариантов названий (табл. 4.1).

В контекстном меню и в палитре свойств	В конструкторе форм	На закладке формы
Форма объекта	Форма документа	Документа
Форма списка	Форма списка документа	Списка
Форма для выбора	Форма выбора документа	Выбора

Таблица 4.1. Основные формы документа.

Типы данных. Типообразующие объекты конфигурации

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.

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.

Объекты конфигурации, которые могут образовывать новые типы данных, называются типообразующими. Эти типы данных не поддерживаются платформой изначально и существуют только в конкретном прикладном решении.

Это небольшое отступление было необходимо потому, что уже при создании первого документа мы столкнемся с использованием типов данных СправочникСсылка.Склады и СправочникСсылка.Номенклатура, которые появились в нашей конфигурации в результате создания объектов конфигурации Справочник Склады и Номенклатура.

Механизм основных форм

Ранее мы создали форму документа Приходная накладная и назначили эту форму основной.

У всех прикладных объектов конфигурации существует некоторое количество основных форм. Они служат для отображения данных объекта в том или ином виде.

Если разработчик не назначит в качестве основных форм объекта свои собственные, система будет генерировать необходимые формы объекта самостоятельно, в те моменты, когда к ним происходит обращение.

Наличие такого механизма позволяет разработчику не тратить время на создание форм для тестирования своей разработки, а воспользоваться тем, что платформа создаст по умолчанию.

Создание этих форм происходит динамически, в процессе работы системы. Форма создается в тот момент, когда к ней происходит обращение. Причем не важно, интерактивное это обращение или программное.

Также примечательным фактом является то, что состав основных форм, определенных для объекта конфигурации, может не совпадать с перечнем тех форм, которые вообще можно создать для данного объекта, используя конструктор формы.

Например, для большинства регистров в конфигураторе можно задать основную форму списка. Однако если открыть конструктор форм для регистра, вы увидите, что кроме формы списка предлагается создать и форму набора записей регистра, которая отсутствует в перечне основных форм.

Обработчики событий

При работе с событиями в платформе 1С:Предприятие 8 следует различать два типа событий: события, связанные с формой и ее элементами, и все остальные.

Разница заключается в том, что обработчики событий, связанных с формой и ее элементами, - назначаемые, а обработчики всех остальных событий - фиксированные.

Фиксированный обработчик события должен иметь имя, совпадающее с именем события. Только в этом случае он будет вызываться при возникновении соответствующего события.

Назначаемый обработчик может иметь произвольное имя. Если имя процедуры совпадает с именем события формы или ее элемента, этого совсем недостаточно для вызова процедуры обработки события с таким именем. Требуется явное назначение процедуры обработчиком этого события в палитре свойств, в соответствующем событии.

Назначение обработчика может выполняться интерактивно, при работе с формой в конфигураторе, или программно, используя методы формы и ее элементов - УстановитьДействие().

Модули

Виды модулей:

- *Модуль управляемого приложения.* Модуль управляемого приложения выполняется при старте системы 1С:Предприятие в режимах тонкого клиента и веб-клиента.

В нем возможно объявление переменных, а также объявление и описание процедур и функций, которые будут доступны в любом модуле конфигурации (кроме модуля внешнего соединения). Их доступность также обеспечивается для неглобальных общих модулей с установленным свойством Клиент (управляемое приложение). В контексте модуля управляемого приложения доступны экспортируемые процедуры и функции общих модулей. Чтобы открыть модуль управляемого приложения, нужно выделить корень дерева объектов конфигурации и вызвать из контекстного меню команду Открыть модуль управляемого приложения

- *Общие модули.* В общих модулях хранятся процедуры и функции, которые вызываются из других модулей системы. Сам по себе общий модуль не исполняется. Исполняются отдельные его процедуры/ функции в момент их вызова из других модулей. Чтобы открыть общий модуль, нужно раскрыть ветвь Общие в дереве объектов конфигурации, затем раскрыть ветвь Общие модули и дважды щелкнуть мышью на нужном модуле.
- *Модули объектов.* Модули объектов - это, например, модуль элемента справочника или модуль документа. Эти модули вызываются тогда, когда либо программно создается этот объект средствами встроенного языка, например, методами Соз-датьЭлемент() у менеджеров справочников или СоздатьДокумент() менеджеров документов, либо когда пользователь создает новый элемент справочника или документ интерактивно. Чтобы открыть модуль объекта, нужно в окне редактирования объекта конфигурации перейти на закладку Прочее и нажать кнопку Модуль объекта. Или, выделив нужный объект в дереве объектов конфигурации, вызвать из контекстного меню команду Открыть модуль объекта.

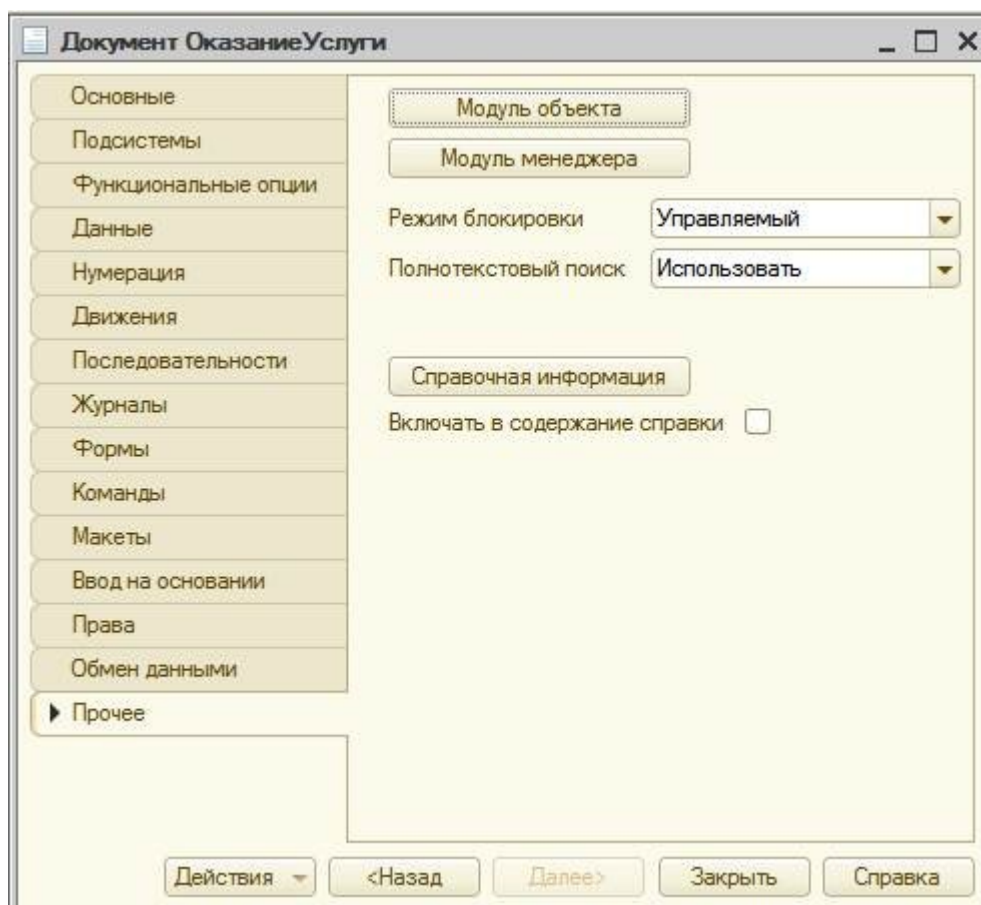
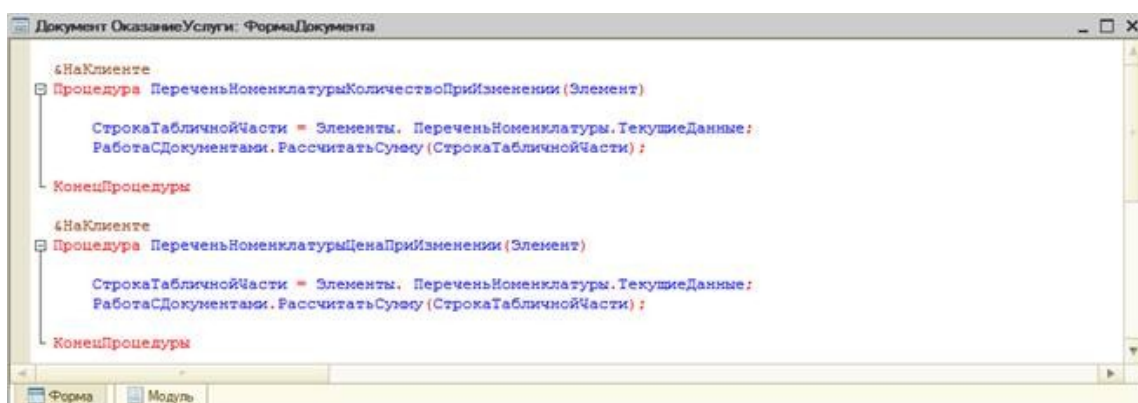


Рис.5.1. Открытие модуля объекта

- **Модули форм.** Каждая форма, определенная в конфигурации, имеет свой собственный модуль. Этот модуль выполняется при создании объекта УправляемаяФорма встроенного языка. Этот объект создается системой в режиме 1С предприятие в тот момент, когда мы программно (методами ПолучитьФорму() или ОткрытьФорму()). Чтобы открыть модуль формы, нужно открыть нужный объект конфигурации Форма объекта и в окне редактора форм перейти на закладку Модуль .



Листинг 5.1. Открытие модуля формы

- **Модуль сеанса.** Модулем сеанса называется модуль, который автоматически выполняется при старте системы 1С:Предприятие в момент загрузки конфигурации. Модуль сеанса предназначен для инициализации

параметров сеанса и обработки действий, связанных с сеансом работы. Модуль сеанса не содержит экспортируемых процедур и функций и может использовать процедуры из общих модулей конфигурации. Чтобы открыть модуль сеанса, нужно выделить корень дерева объектов конфигурации (строку Фамилия) и вызвать из контекстного меню команду Открыть модуль сеанса.

- *Модуль внешнего соединения* предназначен для размещения в нем текстов функций и процедур, которые могут вызываться в сессии внешнего соединения. Чтобы открыть модуль сеанса, нужно выделить корень дерева объектов конфигурации (строку ПособиеДляНачинающих) и вызвать из контекстного меню команду Открыть модуль внешнего соединения.
- *Модуль менеджеров*. Для каждого прикладного объекта существует менеджер, предназначенный для управления этим объектом как объектом конфигурации. С помощью менеджера можно создавать объекты, работать с формами и макетами. Модуль менеджера позволяет расширить функциональность менеджеров, предоставляемых системой, за счет написания процедур и функций на встроенном языке. Чтобы открыть модуль менеджера, нужно в окне редактирования объекта конфигурации перейти на закладку Прочее и нажать кнопку Модуль менеджера.
- *Модуль команды*. Как в самой конфигурации, так и у многих прикладных объектов могут существовать подчиненные объекты конфигурации - Команды. У каждой команды существует модуль команды, в котором можно написать predeterminedную процедуру ОбработкаКоманды() для выполнения этой команды. Чтобы открыть модуль команды, подчиненной некоторому объекту конфигурации, нужно в окне редактирования объекта конфигурации перейти на закладку Команды и дважды щелкнуть мышью на нужной команде. Или, выделив нужную команду в дереве объектов конфигурации, вызвать из контекстного меню команду Открыть модуль команды.

Контекст модуля формы

Каждый модуль связан с остальной частью конфигурации. Эта связь называется контекстом модуля.

Контекст модуля определяет набор доступных во время выполнения модуля объектов, переменных, процедур и функций.

Контекст модуля формы образуется:

- локальным контекстом самого модуля формы;
- реквизитами формы, которой принадлежит модуль;
- свойствами и методами объекта УправляемаяФорма встроенного языка;
- свойствами и методами расширения формы, определяемого типом того объекта, данные которого содержатся в основном реквизите формы;
- глобальным контекстом, в том числе неглобальными общими модулями и экспортируемыми функциями и процедурами глобальных общих модулей;
- экспортируемыми переменными, процедурами и функциями модуля управляемого приложения;

Форма как программный объект

Помимо того что форма;внутри; своего модуля предоставляет доступ к различным частям конфигурации, она также доступна из других частей конфигурации как программный объект.

При этом помимо стандартных свойств и методов объекта встроенного языка УправляемаяФорма, у нее могут существовать и другие свойства и методы, определенные разработчиком.

```
&НаКлиенте
□ Процедура МатериалыКоличествоПриИзменении (Элемент)
    СтрокаТабличнойЧасти = Элементы.Материалы.ТекущиеДанные;
    РаботаСДокументами.РассчитатьСумму (СтрокаТабличнойЧасти);
КонецПроцедуры
```

Листинг 5.2.

Например, если в модуле формы ФормаДокумента документа ПриходнаяНакладная описана экспортируемая процедура, то может быть использован следующий вызов этой процедуры:

```
форма=ПолучитФорму ("Документ.ПриходнаяНакладная.форма.формаДокумента");
форма.МатериалыКоличествоПриИзменении();|
```

Листинг 5.3.

Как понять работу кода на встроенном языке

Теперь мы покажем два способа, как самому разобраться с множеством незнакомых свойств и методов объектов конфигурации, чтобы в будущем самостоятельно изучать фрагменты кода или создавать свои собственные процедуры на встроенном языке.

Синтакс-помощник — инструмент, созданный для помощи разработчику, содержащий описание всех программных объектов, которые использует система, их методов, свойств, событий и пр.

Чтобы открыть синтакс-помощник, нужно нажать соответствующую кнопку на панели инструментов конфигуратора или выполнить команду главного меню Справка > Синтакс-помощник, или с помощью горячей клавиши (Ctrl + F1).

Анализ кода с помощью синтакс-помощника:

Пользоваться синтакс-помощником удобно в тех случаях, когда нужно разобраться в уже написанном незнакомом коде.

Первый способ - найти нужный раздел в содержании и спускаться вниз «по дереву», раскрывая нужные подразделы, свойства, ссылки и т.п.

Второй способ - воспользоваться контекстной помощью синтакс-помощника. Для этого нужно открыть программный модуль, установить курсор на интересующую вас конструкцию встроенного языка и нажать Ctrl+ F1.

Есть также еще одна полезная возможность использования синтакс-помощника. Можно ограничить состав объектов, которые будут отображаться в нем. Так как мы находимся на клиенте, в форме, имеет смысл ограничиться только объектами встроенного языка, доступными в режимах Тонкий клиент и Веб-клиент. Для этого нужно выполнить команду главного меню Сервис > Параметры или нажать кнопку Открыть режим настройки параметров, находящуюся над окном описания объектов синтакс-помощника. На закладке Справка окна Параметры можно снять или поставить отметку у нужных режимов исполнения.

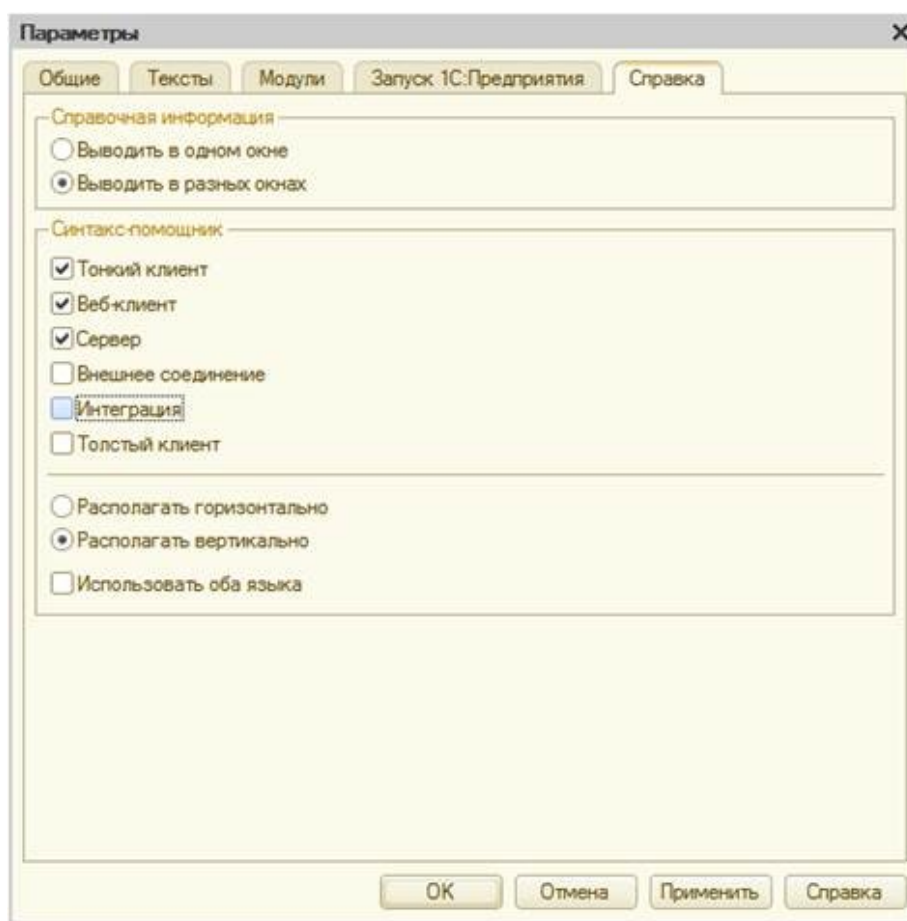


Рис. 5.2. Параметры

Анализ кода с помощью отладчика

Пользоваться отладчиком наиболее удобно в тех случаях, когда нужно написать какой-то собственный код. Можно просто остановиться в конкретном месте программы и посмотреть, какие же свойства здесь доступны или какие программные объекты здесь используются.

Отладчик - вспомогательный инструмент, облегчающий разработку и отладку программных модулей системы 1С:Предприятие. Отладчик предоставляет следующие возможности:

- пошаговое выполнение модуля,
- расстановка точек останова,
- прерывание и продолжение выполнения модуля,
- возможность отладки нескольких модулей одновременно,
- вычисление выражений для анализа состояния переменных,
- просмотр стека вызовов процедур и функций,
- возможность остановки по возникновению ошибки,
- возможность редактирования модуля в процессе отладки.

Объекты

Как правило, термин объект употребляется в одном из трех контекстов:

- конфигурация,
- база данных,
- встроенный язык.

Говоря о конфигурации, термином объект конфигурации мы обозначаем некоторую совокупность описания данных и алгоритмов работы с этими данными. Например, в конфигурации может существовать объект Справочник Сотрудники.

Когда мы говорим о базе данных, термином *объект* мы обозначаем всего лишь некий элемент такой информационной структуры. Характерной особенностью такого элемента является то, что на него (как на совокупность данных) существует ссылка, которая может являться значением какого-либо поля другой информационной структуры.

Если же мы начинаем говорить о встроенном языке и о том, каким образом средствами встроенного языка работать со справочниками, то термином *объект* мы обозначаем тип данных, позволяющий получить доступ к данным и обладающий набором свойств и методов.

Сервер и клиенты

Система 1С:Предприятие поддерживает два варианта работы системы: файловый и клиент-серверный.

Файловый вариант работы с информационной базой рассчитан на персональную работу одного пользователя или работу небольшого количества пользователей в локальной сети. В этом варианте все данные информационной базы (конфигурация, база данных, административная информация) располагаются в одном файле.



Рис. 5.3. Файловый вариант

Основное назначение файлового варианта - быстро и легко установить систему и работать с ней. Например, чтобы что-то посмотреть или доработать дома или на ноутбуке. В файловом варианте тоже можно вести реальную учетную работу, но при этом нужно понимать, что он не предоставляет абсолютно всех тех же возможностей по масштабируемости, защите данных, какие имеет клиент-серверный вариант.

Клиент-серверный вариант предназначен для использования в рабочих группах или в масштабе предприятия - это основной вариант для работы в многопользовательской среде с большим объемом данных. Он предоставляет абсолютно все возможности по масштабируемости, администрированию и защите данных. Однако он требует значительных усилий по установке и администрированию.

Система 1С:Предприятие изначально рассчитана на клиент-серверный вариант работы. Хотя сейчас вы разрабатываете свою учебную конфигурацию в файловом варианте работы, она будет работать и в клиент-серверном варианте без ваших дополнительных усилий.

Прикладные решения разрабатываются один раз и одинаково работают, что в одном, что в другом варианте. То есть переход с одного варианта на другой не требует переделки конфигурации.



Рис. 5.4. Клиент-серверный вариант

Клиент-серверная архитектура разделяет всю работающую систему на три различные части, определенным образом взаимодействующие между собой, - Клиент, Сервер 1С: Предприятия и Сервер баз данных.

Клиентское приложение - это программа, часть системы 1С: Предприятие. Основное ее назначение - организация пользовательского интерфейса, отображение данных с возможностью их изменения. Кроме этого, клиентское приложение может исполнять код на встроенном языке. Такой подход позволяет клиентскому приложению быть очень «легким», не требовать много ресурсов, «путешествовать» по Интернету и работать даже в среде веб-браузеров.

Сервер баз данных - это тоже программа. Она уже не является частью системы 1С:Предприятие, это специализированная программа, поставляемая сторонними производителями. Ее основное назначение - это организация и ведение баз данных - структурированных организованных наборов данных, описывающих характеристики каких-либо физических или виртуальных систем.

В настоящее время система 1С:Предприятие может работать со следующими серверами баз данных:

- **MS SQL Server;**
- **Posteg SQL;**
- **IBM DB2;**
- **Oracle Database.**

Практическая часть

Подсистемы

Каждый объект конфигурации может быть включен в одну или сразу несколько подсистем, в составе которых он будет отображаться.

Сейчас мы создадим в нашей конфигурации пять новых объектов конфигурации Подсистема, которые будут иметь имена: «**Бухгалтерия**», «**РасчетЗарплаты**», «**УчетМатериалов**», «**ОказаниеУслуг**» и «**Предприятие**». Чтобы это сделать, выполним следующие действия.

Добавление подсистемы в режиме «Конфигуратор»

Чтобы создать новые подсистемы, раскроем ветвь «**Общие**» в дереве объектов конфигурации, нажав на **+** слева от нее.

Затем выделим ветвь «**Подсистемы**», вызовем ее контекстное меню и выберем пункт «**Добавить**» или нажмем соответствующую кнопку в командной панели окна конфигурации.

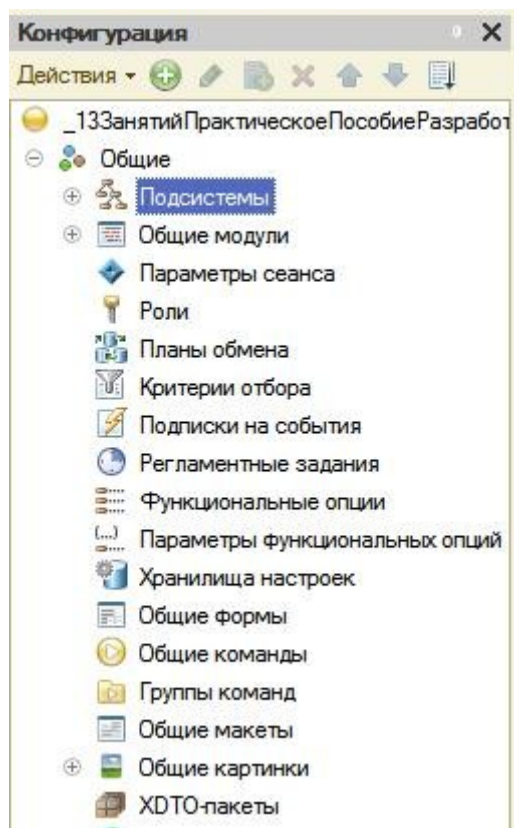


Рис. 2.2. Дерево объектов конфигурации

После этого система откроет окно редактирования объекта конфигурации.

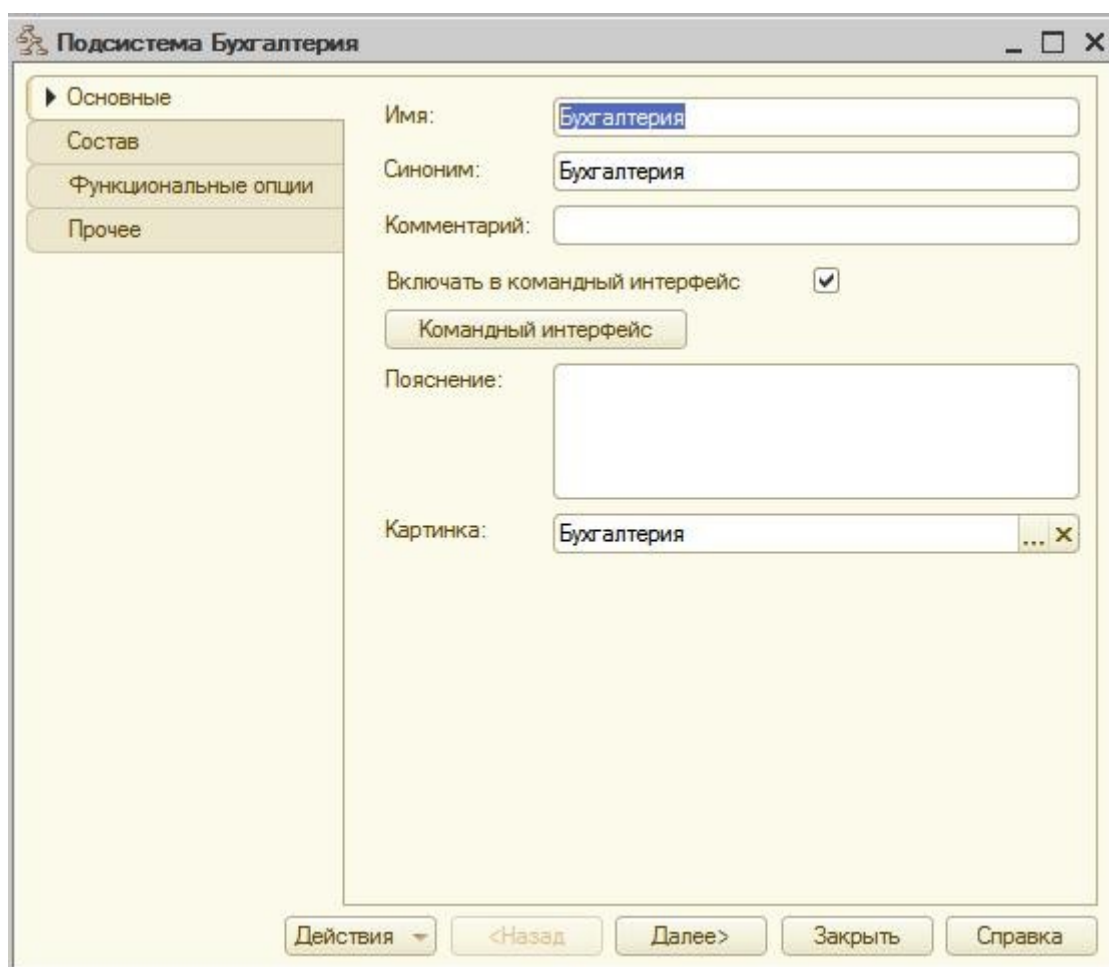


Рис. 2.3. Подсистема Бухгалтерия

Зададим имя подсистемы – «**Бухгалтерия**». На основании имени платформа автоматически создаст синоним – «**Бухгалтерия**».

Имя и синоним объекта конфигурации

Имя является основным свойством любого объекта конфигурации. При создании нового объекта система автоматически присваивает ему некоторое имя.

Свойство **Синоним** также есть у любого объекта конфигурации. Оно предназначено для хранения «альтернативного» наименования объекта конфигурации, которое будет использовано в элементах интерфейса нашей программы, то есть будет показано пользователю. Поэтому на синоним практически нет никаких ограничений.

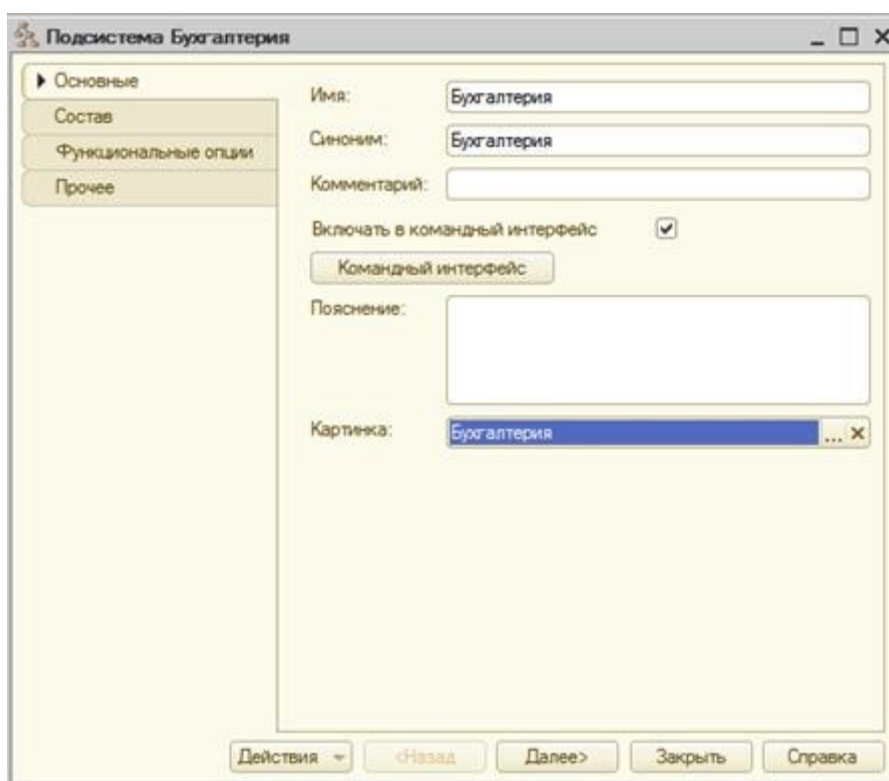


Рис. 2.4. Подсистема Бухгалтерия

В целях усовершенствования интерфейса приложения мы можем также задать картинку для отображения подсистемы.

Система создаст объект конфигурации Общая картинка и откроет окно редактирования его свойств.

Картинке можно присвоить свое имя, а так же выбрать свое изображение с локального диска. Для просмотра изображений поставим флажок **«Просмотр»**. Выбранная нами картинка появится в окне редактирования общей картинки. Закроем окно редактирования объекта конфигурации

Итак, мы вернулись в окно редактирования объекта конфигурации Подсистема Бухгалтерия. Мы видим, что выбранная нами одноименная картинка установилась в качестве картинки для подсистемы.

Таким образом, в интерфейсе 1С Предприятия в качестве названия раздела будет показан синоним подсистемы, и над ним будет выводиться указанная картинка.

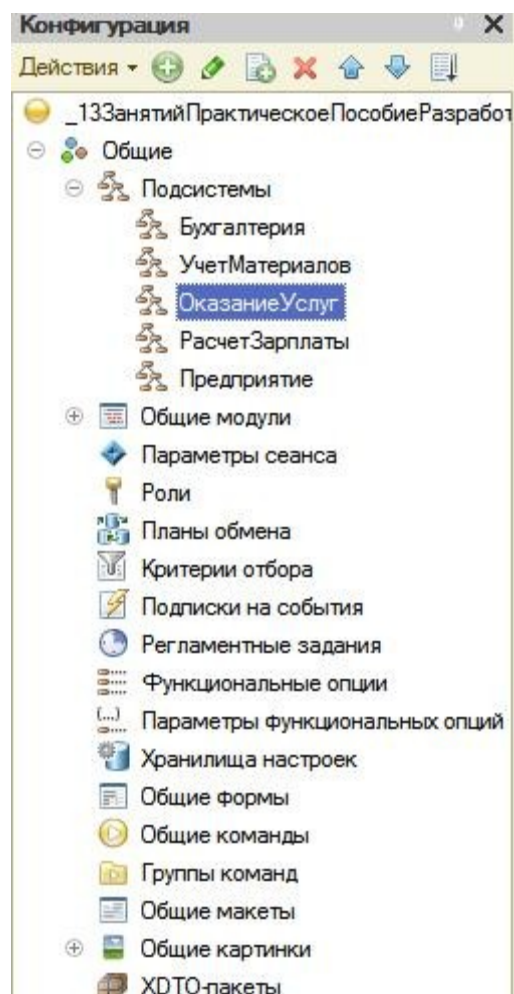


Рис. 2.5. Дерево объектов конфигурации

Снова выделим ветвь «Подсистемы», нажмем кнопку «Добавить» в дереве объектов конфигурации и создадим подсистемы с именами «УчетМатериалов» и «ОказаниеУслуг». Установим для них картинки.

Теперь воспользуемся другим способом для добавления подсистем. Вызовем контекстное меню одной из созданных подсистем. Выберем в нем пункт «Добавить». Он разбивается на два подпункта. Выбор подпункта «Подсистема» позволяет добавить подсистему того же уровня иерархии, что и выделенная. Выбор подпункта «Подчиненная Подсистема» позволяет добавить подсистему, подчиненную выделенной. Поскольку в нашей конфигурации не планируется сложной многоуровневой структуры, выберем первый вариант и добавим подсистему «РасчетЗарплаты». Установим для нее в качестве картинки общую картинку «Зарплата», добавив ее из файла «Зарплата».

В заключение добавим подсистему Предприятие для доступа к административным и сервисным функциям.

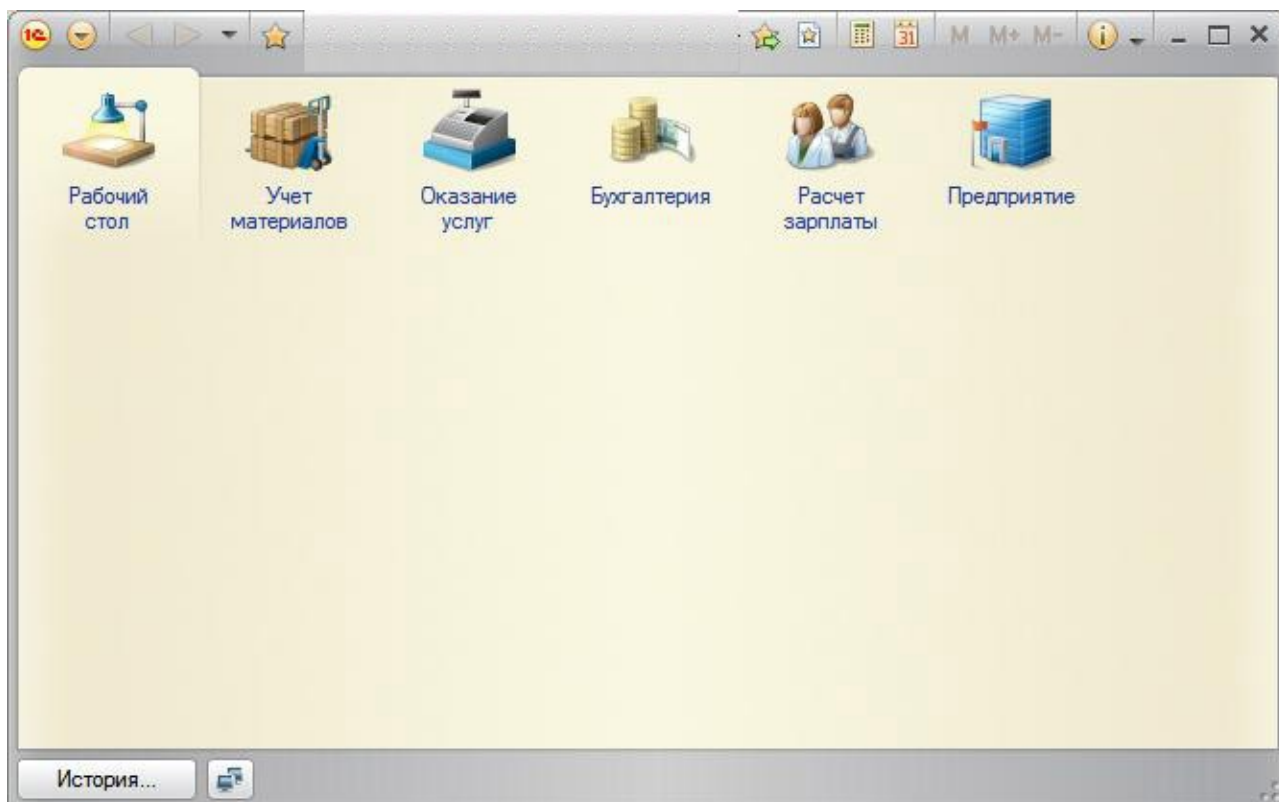


Рис. 2.6. Подсистемы

Панель разделов прикладного решения в режиме 1С:Предприятие

Запустим 1С Предприятие в режиме отладки и увидим результат наших изменений. Вид разрабатываемого нами приложения изменился. Сразу под главным меню располагается панель разделов приложения, где и отражены созданные нами подсистемы. Причем все разделы выводятся с выбранными в их свойствах картинками. Разделы представлены в форме гиперссылок, нажав на которые пользователь может открыть связанные с ними документы, справочники, отчеты и т. п.

Обратите внимание, что раздел **Рабочий стол** формируется платформой по умолчанию. Он предназначен для размещения наиболее часто используемых пользователем документов, отчетов и т. п.

Порядок разделов в режиме Конфигуратор

Однако порядок расположения подсистем нас не совсем устраивает. Изменим его. Закроем приложение и вернемся в конфигуратор. Выделим корень дерева объектов конфигурации «**Фамилия**», нажатием правой кнопки мыши вызовем контекстное меню и выберем пункт «**Открыть командный интерфейс конфигурации**»

В открывшемся окне «**Командный интерфейс**» вы увидите список созданных вами подсистем (разделов приложения). Расположим сначала подсистемы, отражающие производственную деятельность нашей фирмы: «**Учет материалов**» и «**Оказание услуг**», затем бухгалтерскую деятельность и расчет зарплаты сотрудников: «**Бухгалтерия**» и «**Расчет зарплаты**», а затем подсистему «**Предприятие**».

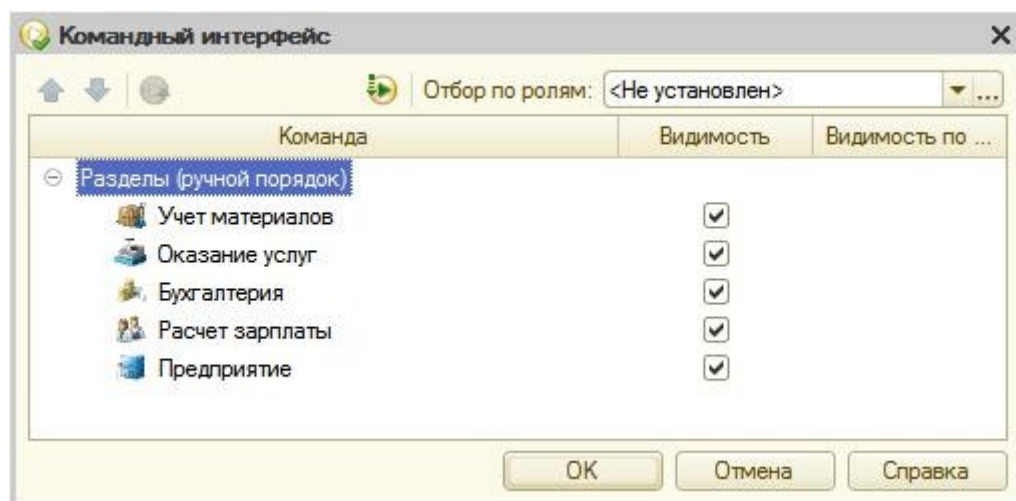


Рис. 2.7. Командный интерфейс

В режиме 1С:Предприятие

Запустим 1С Предприятие в режиме отладки и увидим, что порядок расположения подсистем в панели разделов приложения изменился так, как мы его задали.

Примечание:

После успешного завершения каждого занятия рекомендуется сохранять конфигурацию, выполнив команду главного меню **Администрирование > Выгрузить информационную базу**. Для загрузки БД выполните: **Администрирование > Загрузить информационную базу**.

Окно редактирования объекта конфигурации и палитра свойств

На первый взгляд окно редактирования объекта и палитра свойств дублируют друг друга. *Окно редактирования объекта конфигурации предназначено в первую очередь для быстрого создания новых объектов.* Окно редактирования объекта помогает быстро создать незнакомый объект конфигурации и обеспечивает удобный доступ к нужным свойствам.

Что же касается палитры свойств, то она предоставляет одну абсолютно незаменимую возможность. Дело в том, что она не привязана по своей структуре к какому-то конкретному виду объектов конфигурации. Ее содержимое меняется в зависимости от того, какой объект является текущим. За счет этого она может «запоминать», какое свойство объекта в ней выбрано, и при переходе в дереве к другому объекту будет подсвечивать у себя все то же свойство, но уже другого объекта.

Справочники

Для начала нам понадобится список сотрудников предприятия, которые будут оказывать услуги. Затем нам будет нужен список клиентов, с которыми работает наше предприятие. После этого нам понадобится перечень услуг, которые может оказывать наше предприятие, и список материалов, которые могут быть израсходованы.

Кроме этого, нам потребуется список складов, на которых могут находиться материалы.

1.Создание простого справочника

Создадим справочник, в котором будут храниться наименования наших клиентов

- **В режиме Конфигуратор:**

Откроем в конфигураторе конфигурацию, выделим в дереве объектов конфигурации ветвь «**Справочники**» и нажмем кнопку «**Добавить**» в командной панели окна конфигурации.

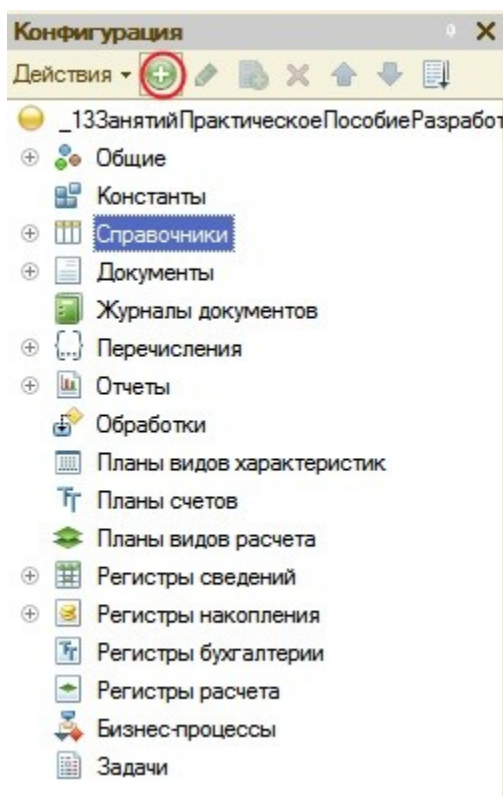


Рис. 3.4. Дерево объектов конфигурации

В открывшемся окне редактирования объекта конфигурации зададим имя справочника - **Клиенты**. На основании имени платформа автоматически создаст синоним - **Клиенты**. Напомним, что свойство **Синоним** служит для представления объекта в интерфейсе нашей программы.

Также у разработчика есть возможность установки дополнительных свойств, определяющих пользовательское представление объектов. Эти свойства задавать не обязательно.

- **Представления объекта в конфигурации**

Представление объекта определяет название объекта в единственном числе и используется в названии стандартной команды, например, команды создания объекта - **Клиент: создать**. Представление объекта нужно задавать тогда, когда синоним объекта конфигурации задан во множественном числе, или когда он описывает множество объектов. Потому что в интерфейсе автоматически

формируются команды открытия списка справочника и команды создания нового элемента справочника.

Рис. 3.5. Справочник Клиенты

Представление объекта используется для того, чтобы описать, как будет выглядеть в интерфейсе команда добавления нового клиента. Также оно будет использовано в заголовке формы и в представлении ссылки на клиента.

Расширенное представление объекта определяет заголовок формы объекта, например формы для создания нового элемента справочника. Если это свойство не задано, то вместо него используется свойство **Представление объекта**.

Представление списка определяет название списка объектов и используется в названии стандартной команды, например, команды открытия списка объектов - **Клиенты: открыть**. Представление списка нужно задавать тогда, когда синоним задан в единственном числе.

Расширенное представление списка определяет заголовок формы списка, например формы списка справочника. Если это свойство не задано, то вместо него используется свойство **Представление списка**.

В лабораторной работе зададим два свойства **Представление объекта - Клиент** и **Представление списка - Клиенты**. Последнее можно было и не задавать, так как синоним справочника совпадает со свойством **Представление списка**.

- **Принадлежность объекта к подсистемам**

Нажмем кнопку **Далее** и перейдем на закладку **Подсистемы**. На этой закладке определяется, в каких подсистемах будет отображаться данный справочник.

В списке подсистем мы видим подсистемы, созданные ранее при определении структуры приложения. Логично предположить, что список клиентов должен быть доступен в разделе **Оказание услуг**. Бухгалтерская отчетность, формируемая в разделе **Бухгалтерия**, также может быть представлена в разрезе клиентов. Поэтому отметим в списке подсистемы **Бухгалтерия** и **ОказаниеУслуг**.

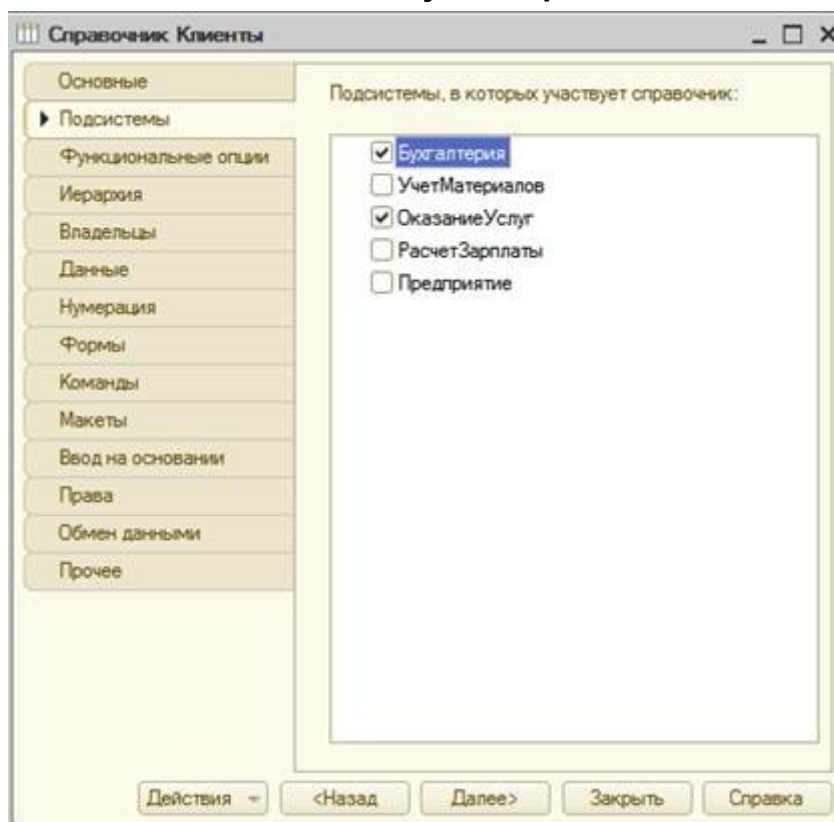


Рис. 3.6. Справочник Клиенты

- **Код и наименование справочника**

В окне редактирования объекта конфигурации Справочник нажмем на вкладку **Данные**. Здесь для нас представляют интерес длина кода и длина наименования. Длина кода - количество элементов, содержащихся в справочнике.. 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.

Зададим длину кода - 9 символов, а длине наименования увеличим длину до 50.

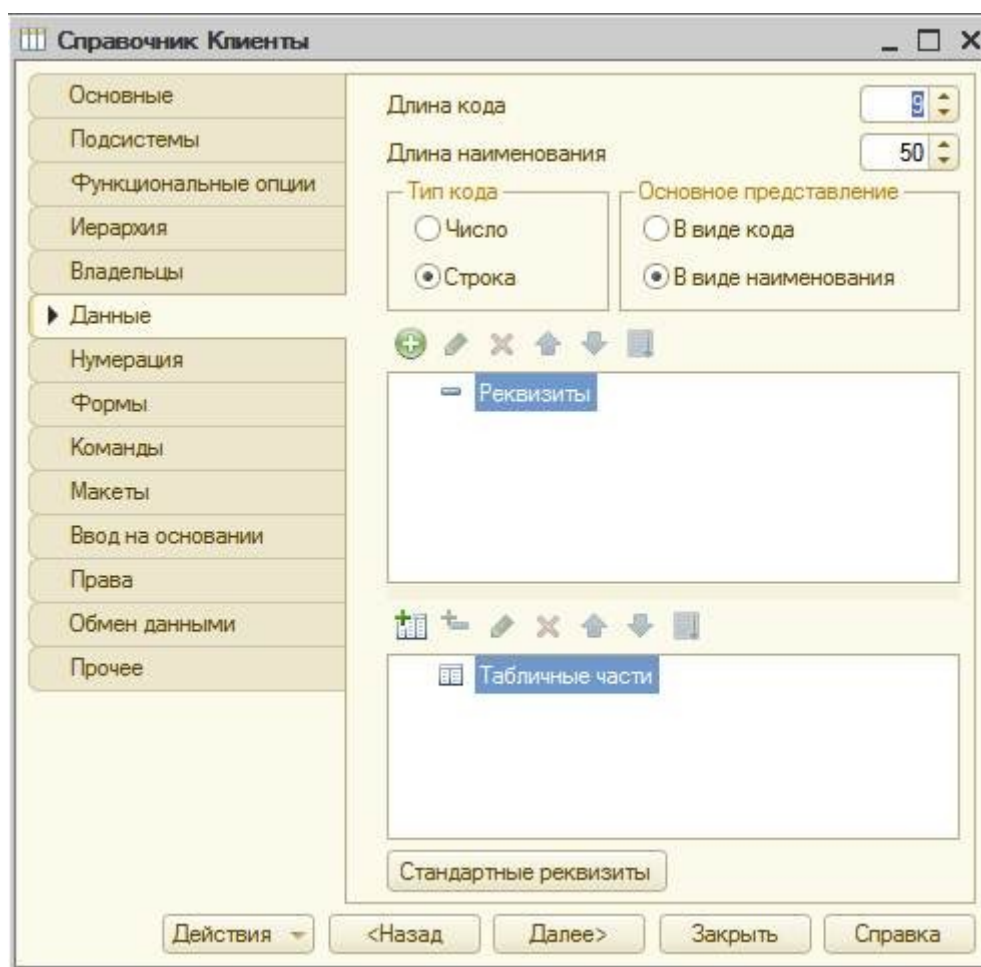


Рис. 3.7. Справочник Клиенты

- **Команда добавления нового элемента**

Прежде чем запускать 1С предприятие, настроим интерфейс приложения, чтобы было удобнее вводить новые элементы справочника.

Команда для открытия списка справочника, как и команда для создания его новых элементов, добавляется в интерфейс тех разделов (подсистем), в которых будет отображаться справочник. Но команда создания новых элементов по умолчанию невидима в интерфейсе приложения.

Сделаем доступной в панели действий раздела **ОказаниеУслуг** стандартную команду для создания новых клиентов.

Для этого в дереве объектов конфигурации выделим ветвь **Подсистемы**, вызовем ее контекстное меню и выберем пункт **Все подсистемы**, выделим подсистему **ОказаниеУслуг**. В списке **Командный интерфейс** отразятся все команды выбранной подсистемы. В группу **Панель действий.Создать** добавилась команда **Клиент: создать** для создания нового элемента справочника, включим видимость у этой команды.

*Для подсистемы **Бухгалтерия** никаких команд добавлять в панель действий не будем, так как это определяется прикладной логикой работы.*

Закроем окно редактирования справочника **Клиенты** и запустим 1С предприятие в режиме отладки. Ответим утвердительно на запрос конфигуратора об обновлении конфигурации и нажмем кнопку **Принять**

- **В режиме 1С: Предприятие. Панель навигации и панель разделов**

Перед нами откроется окно системы в режиме 1С:Предприятие. Мы видим, что если перейти в раздел **Оказание услуг** или **Бухгалтерия**, то слева в вертикальной области окна появится панель навигации (см рис).

Панель навигации отображает структуру выбранного раздела, предназначена для быстрого перехода к различным спискам в пределах выбранного раздела программы.

Также в разделе **Оказание услуг** появилась панель действий. Панель действий содержит команды, которые соответствуют текущему разделу, выбранному в панели разделов.

Сейчас в панели действий раздела **Оказание услуг** в группе **Создать** доступна команда для создания элементов нашего первого справочника **Клиенты**. Этой командой мы и воспользуемся для создания новых элементов справочника, не открывая при этом списка клиентов.

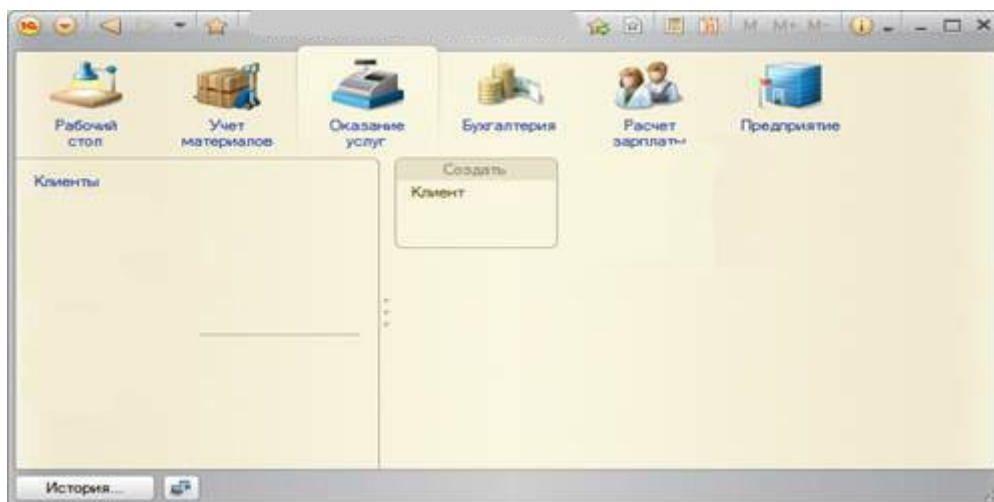


Рис. 3.8. Панель навигации и панель разделов

- **Создание элементов справочника**

Пока наш справочник пуст, поэтому добавим в него несколько элементов. Для этого выполним команду **Клиент** в панели действий раздела **Оказание услуг**. Перед нами откроется форма для создания элемента справочника.

Внесем последовательно в справочник 3-х новых клиентов. Код вносить не нужно, так как он генерируется автоматически. После последовательного добавления клиентов нажмем **Записать и закрыть**.

Чтобы открыть существующий элемент справочника для редактирования, нужно дважды щелкнуть на нем мышью. А кроме этого, нажав на ссылку в

информационной панели, мы тоже можем открыть для редактирования один из последних измененных элементов справочника.

2. Создание справочника с табличной частью

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

Этот справочник будет устроен несколько сложнее, чем справочник **Клиенты**. Дело в том, что в нем мы будем хранить не только фамилию, имя и отчество сотрудника, но и информацию о его прошлой трудовой деятельности.

- **В режиме Конфигуратор**

Добавим новый объект конфигурации Справочник. Назовем его **Сотрудники**.

На основании имени платформа автоматически заполнит его синоним.

Зададим **Представление объекта** как **Сотрудник**. **Представление списка** устанавливать не будем, а **Расширенное представление списка** зададим как **Список сотрудников**.

Нажмем кнопку **Далее** и перейдем на закладку **Подсистемы**.

По логике нашей конфигурации список сотрудников должен быть доступен в разделах **Оказание услуги** и **Расчет зарплаты**.

Перейдем на закладку **Данные**. Оставим по умолчанию длину и тип кода, длину наименования справочника зададим равной 50 символам.

- **Табличная часть**

Наша задача - создать справочник, имеющий табличную часть. Поэтому добавим в справочник новую табличную часть с именем **ТрудоваяДеятельность**.

Для этого нажмем кнопку **Добавить табличную часть** над списком табличных частей справочника. Зададим имя – **ТрудоваяДеятельность**.

После чего добавим реквизиты табличной части, для этого нажмем на кнопку **Добавить реквизит**. Добавим следующие реквизиты:

- **Организация** - тип **Строка**, длина 100;
- **НачалоРаботы** - тип **Дата**, состав даты - **Дата**;
- **ОкончаниеРаботы** - тип **Дата**, состав даты - **Дата**;
- **Должность** - тип **Строка**, длина 100.

В заключение отредактируем командный интерфейс, чтобы нам было удобнее вводить новые элементы справочника. Сделаем видимой в панели действий подсистемы **РасчетЗарплаты** стандартную команду для создания новых сотрудников.

Сделаем это в дереве объектов конфигурации в ветви **Подсистемы\Все подсистемы**:

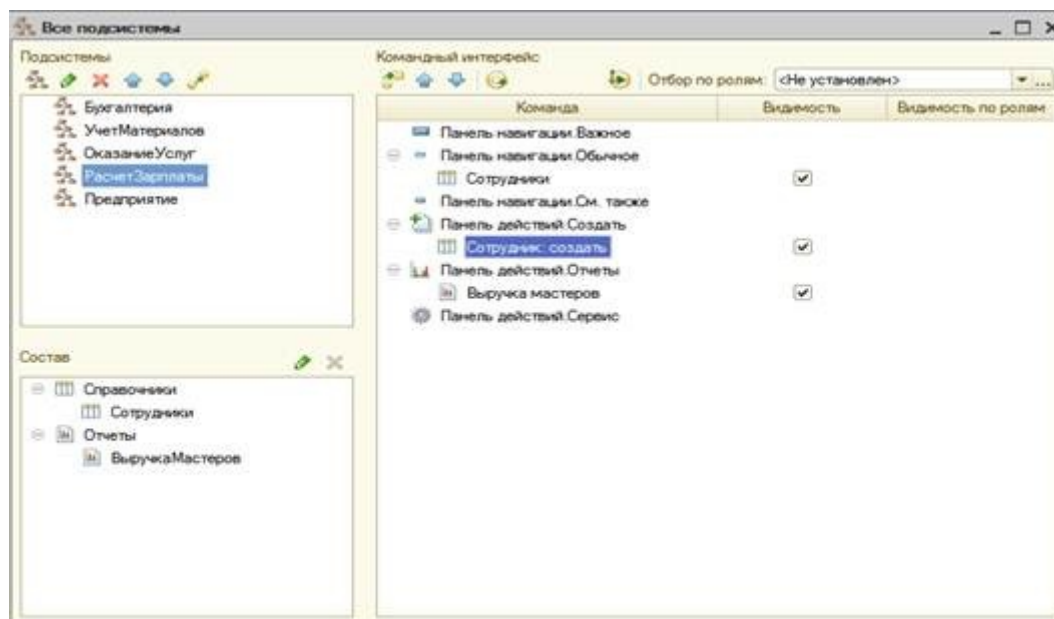


Рис. 3.9. Все подсистемы

Для подсистемы **ОказаниеУслуг** никаких команд добавлять в панель действий не будем, так как вряд ли понадобится пополнять список сотрудников в этом разделе. На этом создание справочника **Сотрудники** завершено.

Обновим конфигурацию, подтвердим обновление и запустим программу в режиме **Предприятие**.

- **Заполнение табличной части в режиме 1С: Предприятие**

В панели действий раздела **Расчет зарплаты** появилась команда **Сотрудник** для создания новых сотрудников. Этой командой мы и воспользуемся для создания новых элементов справочника, не открывая при этом списка сотрудников. Выполним команду **Сотрудник**. Перед нами откроется форма для создания элемента справочника - основная форма объекта. Заголовок этой формы определяется свойством **Представление объекта**.

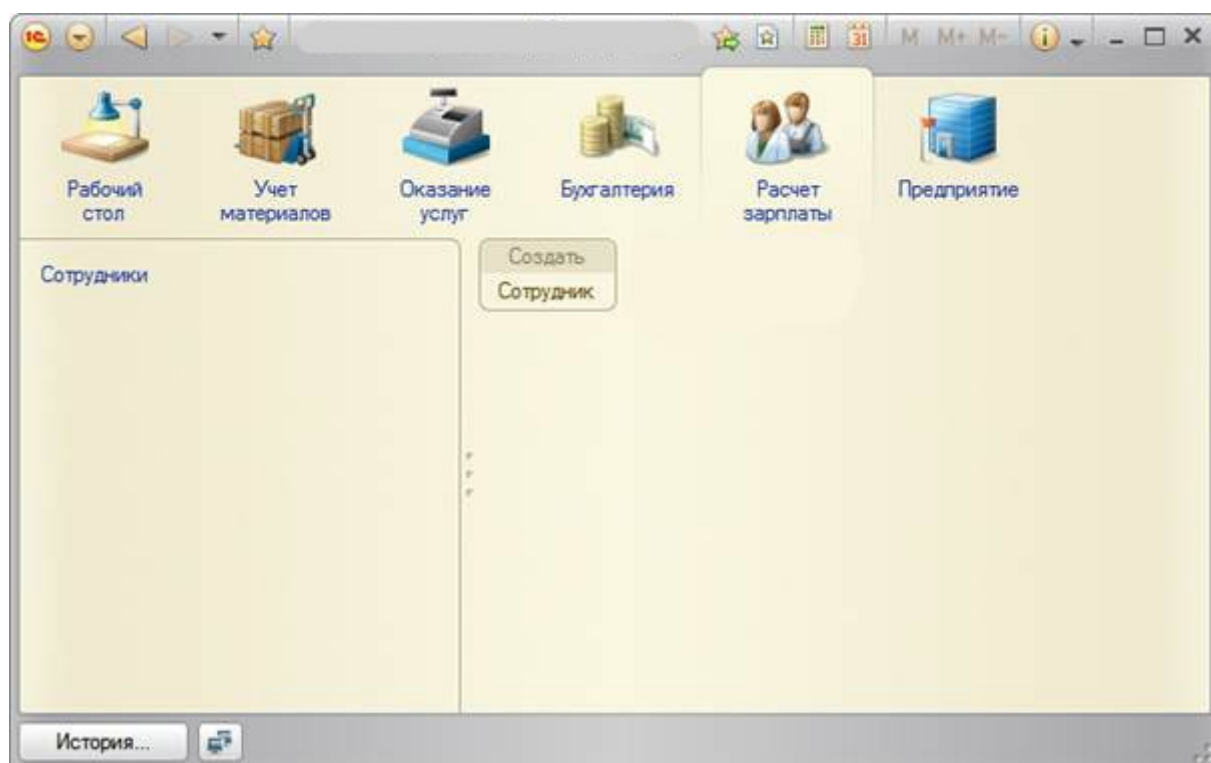


Рис. 3.10. Заполнение табличной части

Эта форма содержит табличную часть с реквизитами, которые мы описали в конфигураторе для этого справочника. Теперь приступим к созданию сотрудников, в этом вам поможет ваша фантазия (для корректной работы заполните все требующиеся поля таблицы)

Сотрудник (создание) *

Записать и закрыть

Код:

Наименование:

+ Добавить

N	Организация	Начало работы	Окончание работы	Должность
1	ЗАО "НПО СпецСв..."	16.12.2008	17.02.2012	Инженер

Рис. 3.11. Заполнение элемента справочника "Сотрудники"

3. Создание иерархического справочника

Справочник **Номенклатура** будет содержать информацию об услугах, которые оказывает наше предприятие, и о тех материалах, которые при этом могут быть использованы.

Этот справочник не будет сложным. Единственная особенность, которой он будет обладать, - наличие иерархической структуры. Для того чтобы справочником было удобно пользоваться, мы сгруппируем услуги в одну группу, а материалы - в другую.

- **В режиме Конфигуратор**

Создадим новый объект конфигурации Справочник и назовем его **Номенклатура**. На основании имени платформа автоматически заполнит его синоним.

Поскольку понятие **Номенклатура** не имеет единственного числа, больше никаких свойств, определяющих представление объекта в интерфейсе приложения, задавать не будем. Вместо **Представления объекта** и **Представления списка** будет использоваться **Синоним объекта - Номенклатура**.

- На закладке **Подсистемы**: По логике нашей конфигурации список номенклатуры должен быть доступен в разделах **Учет материалов**, **Оказание услуг** и **Бухгалтерия**, поэтому отметим в списке подсистем эти подсистемы.

- На закладке **Иерархия**: установим флажок **Иерархический справочник**.

- На закладке **Данные**: оставим по умолчанию длину и тип кода, длину наименования справочника зададим равной 100 символам.

- В дереве объектов конфигурации (**Подсистемы\Все подсистемы**): в открывшемся окне слева в списке **Подсистемы** выделим подсистему **УчетМатериалов**. В группе **Панель действий.Создать** включим видимость у команды **Номенклатура: создать**.

Выделив в списке подсистем **ОказаниеУслуг**, сделаем те же действия.

А для подсистемы **Бухгалтерия** никаких команд добавлять в панель действий не будем, так как вряд ли понадобится пополнять список номенклатуры в этом разделе.

Теперь заполним справочник **Номенклатура**. В процессе заполнения вы научитесь создавать группы и переносить элементы из одной группы в другую.

- **В режиме 1С:Предприятие**

В открывшемся окне 1С:Предприятия мы видим, что в панели навигации разделов **Учет материалов**, **Оказание услуг** и **Бухгалтерия** появилась команда **Номенклатура**.

Название этой команды определяется синонимом объекта, так как других представлений мы для этого справочника не задавали.

Выполним команду **Номенклатура** в панели навигации раздела **Учет материалов**. Справа от панели навигации в рабочей области окна приложения откроется основная форма списка.

- Создадим две группы в корне справочника: **Материалы** и **Услуги**.
Для этого нажмем кнопку Создать группу в командной панели формы списка.

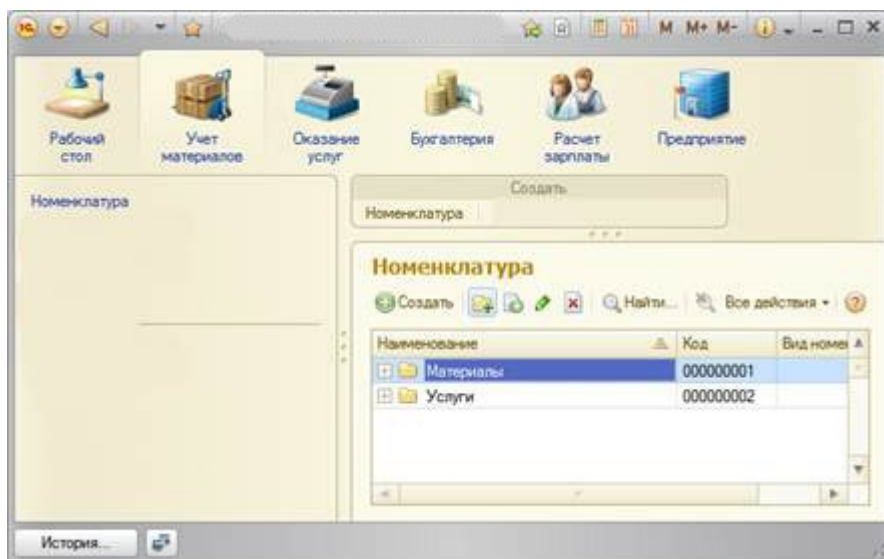


Рис. 3.12. Создание групп в корне справочника

А) Зададим наименование групп: **Материалы**, **Услуги**. Поля Родитель и Код заполнять не будем.

Б) Раскроем группу **Материалы** (нажатием на +) и создадим в ней пять элементов. После этого раскроем группу **Услуги** и тоже создадим в ней несколько элементов (не более 3).

- *Перенос элементов в другие группы*

Разнесите услуги по двум смысловым группам: Для этого в группе Услуги необходимо создать еще две подчиненных группы, которые по смыслу относятся к предоставляемым услугам.

*Для того чтобы переместить услуги в соответствующие группы, в окне списка установим курсор на ту услугу, которую мы хотим переместить, и выполним команду **Все действия > Переместить в группу**. В открывшемся окне выберем новую группу.*

4. Создание справочника с predetermined elements

В заключение мы создадим справочник **Склады**. Этот справочник будет включать в себя один predetermined element - склад **Основной**, на который будут поступать все материалы. *Наша задача - создать справочник, содержащий predetermined elements.*

- **В режиме Конфигуратор**

Откроем конфигуратор и создадим новый объект конфигурации Справочник с именем **Склады**. На основании имени платформа автоматически заполнит его синоним.

- На вкладке **Основные**: Зададим **Представление объекта** - Склад. **Представления списка** - Склады.

- На вкладке **Подсистемы**: Отметим в списке подсистем подсистемы **Оказание услуг** и **Учет материалов**.

- На вкладке **Поле ввода**: установим флажок в поле **Быстрый выбор**.

Свойство **Быстрый выбор** как раз позволяет выбирать элементы не из отдельной формы, а из небольшого выпадающего списка, заполненного элементами этого справочника. Этот вариант наиболее удобен для списка складов, так как их, вероятно, будет немного.

- На вкладке **Прочее**: нажмем кнопку **Предопределенные**.

Система откроет список предопределенных элементов справочника.

Сейчас он пуст, поэтому нажмем кнопку **Добавить** и создадим предопределенный элемент с именем **Основной**.

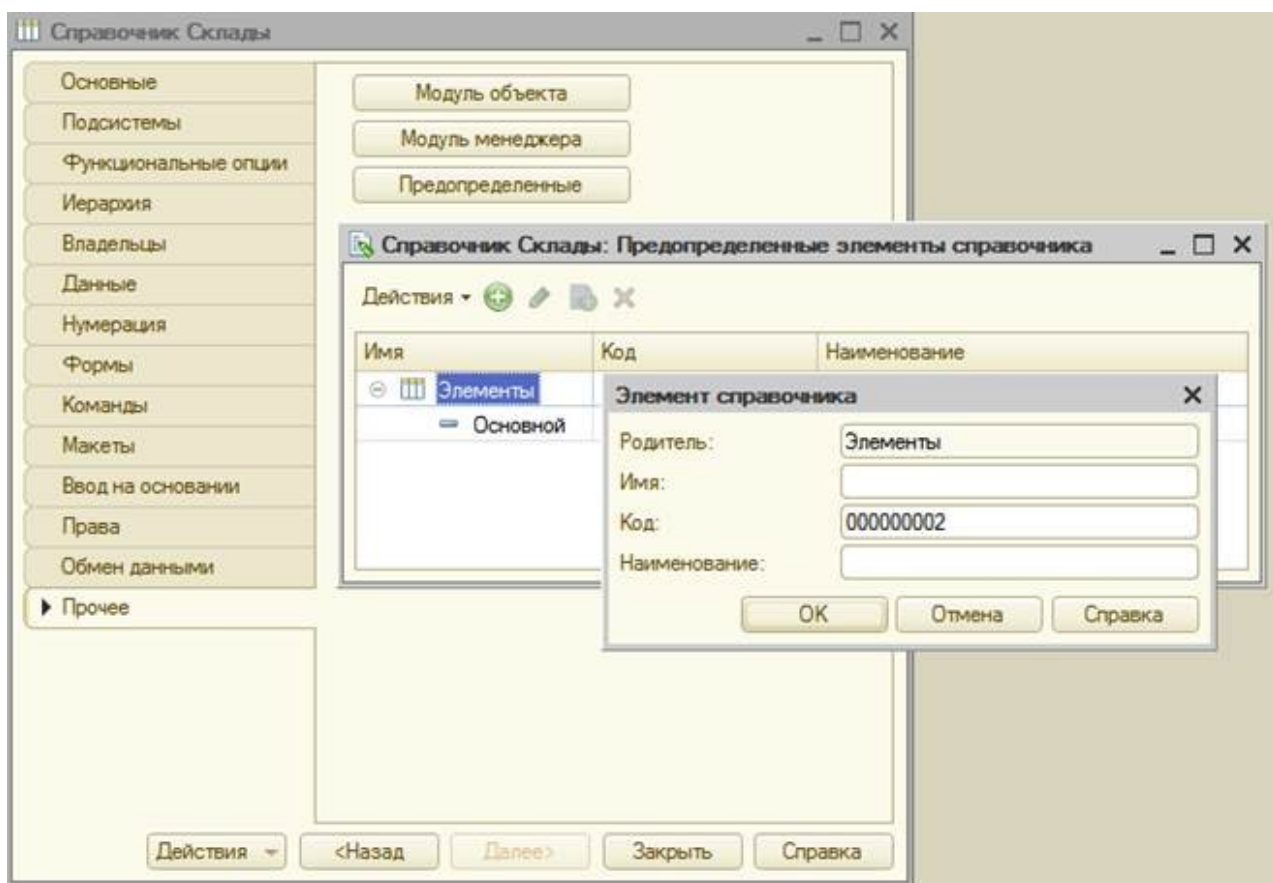


Рис. 3.13. Создание справочника с предопределенными элементами

- В дереве объектов конфигурации (**Подсистемы\Все подсистемы**): в открывшемся окне слева в списке **Подсистемы** выделим подсистему **УчетМатериалов**. В группе **Панель действий.Создать** включим видимость у команды **Склад: создать**.

- **В режиме 1С: Предприятие**

В открывшемся окне 1С:Предприятия мы видим, что в панели действий раздела **Учет материалов** появилась команда **Склад** для создания новых складов. Также в панели навигации разделов **Оказание услуг** и **Учет материалов** появилась команда **Склады** для открытия списка складов.

- *Выполним команду **Склады** в панели навигации раздела **Учет материалов**.*

В списке складов уже есть один элемент с наименованием **Основной**. Это предопределенный элемент, который мы создали в конфигураторе.

Выполнив команду **Склад** в панели действий, добавим в справочник еще один склад, который назовем **Розничный**.

Документы

Создание Документа «Приходная накладная»

Для отражения событий, происходящих на предприятии, в базе данных мы создадим два документа: **Приходная накладная** и **Оказание услуги**.

Документ **Приходная накладная** будет фиксировать факт поступления в нашу организацию необходимых материалов, а документ **Оказание услуги** - фиксировать оказание услуг и расход материалов, которые используются при оказании этих услуг.

- **В режиме конфигуратор**

Откроем конфигуратор и добавим новый объект конфигурации Документ.

- *На закладке **Основные***: зададим имя документа - ПриходнаяНакладная. На основании имени платформа автоматически заполнит его синоним. Так же зададим Представление списка, как Приходные накладные.

- *На закладке **Подсистемы***: список приходных накладных должен быть доступен в разделах Учет материалов и Бухгалтерия.

- *На закладке **Данные***: создадим реквизит документа с именем Склад (нажать на «+»); далее выберем для реквизита ссылочный тип данных СправочникСсылка.Склады. Этот тип стал доступен в конфигурации после создания объекта конфигурации Справочник Склады. В качестве значения свойства **Значение заполнения** выберем предопределенный элемент справочника Склады - Основной.

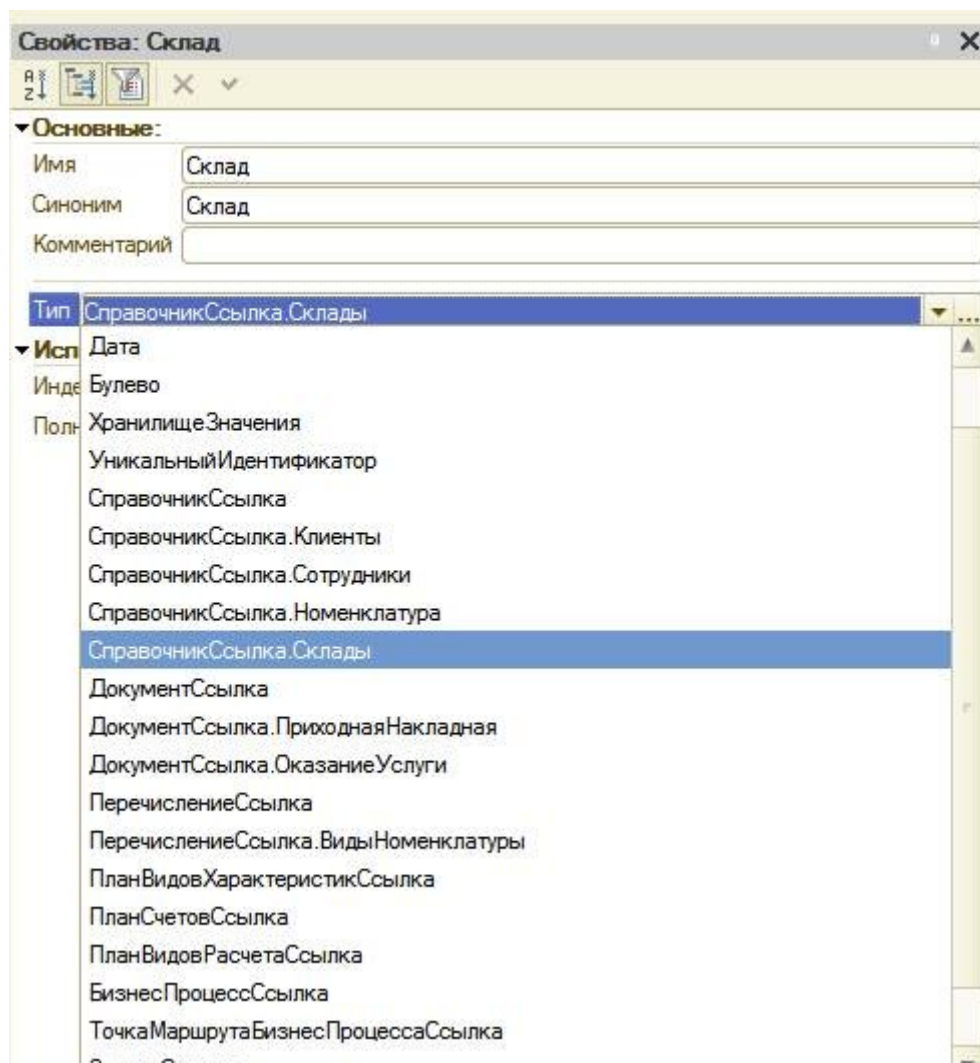


Рис. 4.1. Справочник Склады

После этого добавим в документ табличную часть с именем Материалы. Для этого нажмем кнопку Добавить табличную часть над списком табличных частей документа. Кроме имени табличной части установим свойство Проверка заполнения в значение Выдавать ошибку.

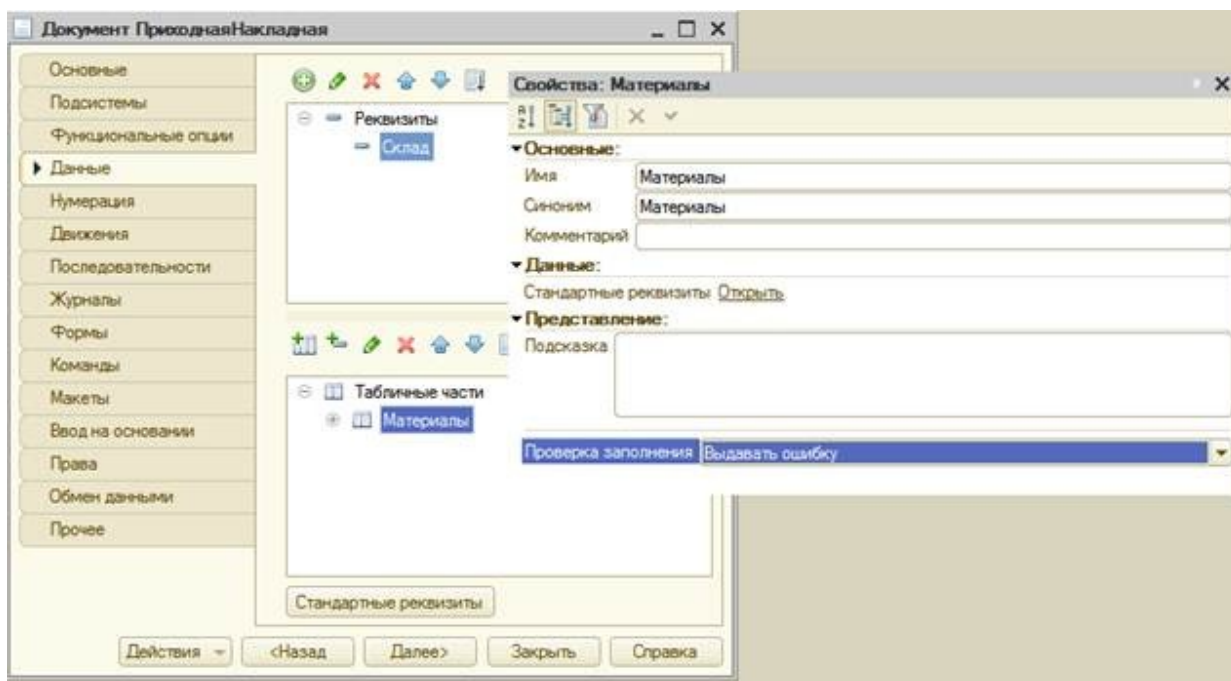


Рис. 4.2. Свойства: Материалы

Создадим реквизиты табличной части Материалы. Для этого нажмем кнопку Добавить реквизит в разделе описания табличных частей документа:

- Материал, 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.

Для каждого реквизита табличной части также установим свойство Проверка заполнения в значение Выдавать ошибку.

- На закладке Нумерация: убедимся, что свойство Авто-нумерация включено.
- В дереве объектов конфигурации выделим ветвь Подсистемы, вызовем ее контекстное меню и выберем пункт Все подсистемы. В открывшемся окне в списке Подсистемы выделим подсистему УчетМатериалов: включим видимость у команды Приходная накладная: создать, в группе Панель действий.Создать.

- В режиме 1С:Предприятие

Запустим 1С:Предприятие в режиме отладки и протестируем получившийся документ. В открывшемся окне 1С:Предприятия мы видим, что в панели навигации разделов Бухгалтерия и Учет материалов появилась команда Приходные накладные для открытия списка приходных накладных.

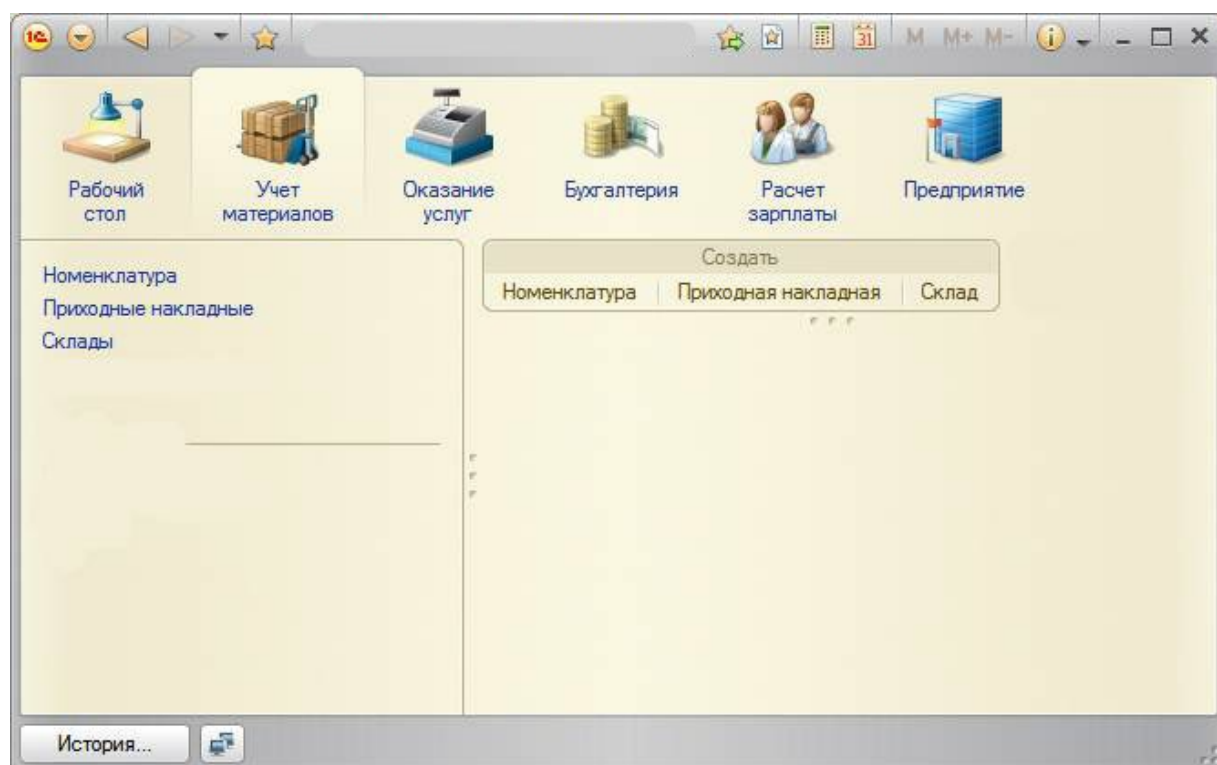


Рис. 4.3. В режиме 1С:Предприятие

Пока в БД нет ни одного документа **Приходная накладная**, поэтому выполним команду **Приходная накладная** в панели действий раздела **Учет материалов** и создадим новую приходную накладную.

Система автоматически подставит текущую дату создания документа и нулевое время, так как документ еще не проведен. В качестве времени документа при оперативном проведении ему присваивается оперативная отметка времени.

Поле **Номер** не заполнено, но система сама сгенерирует для нового документа уникальный номер, так как свойство **Автономерация** для документа включено по умолчанию. Новый номер будет сохранен в момент записи документа. *Осталось только заполнить табличную часть приходной накладной материалами (добавьте 3 любых записи в накладную). Так же система может вводить данные с клавиатуры и предлагать вам материалы по совпадающим буквам (создайте еще одну накладную таким образом)* После чего нажмем Провести и закрыть.

Документ будет сохранен и проведен, ему будет присвоен автоматически сгенерированный системой номер и текущее время проведения документа.

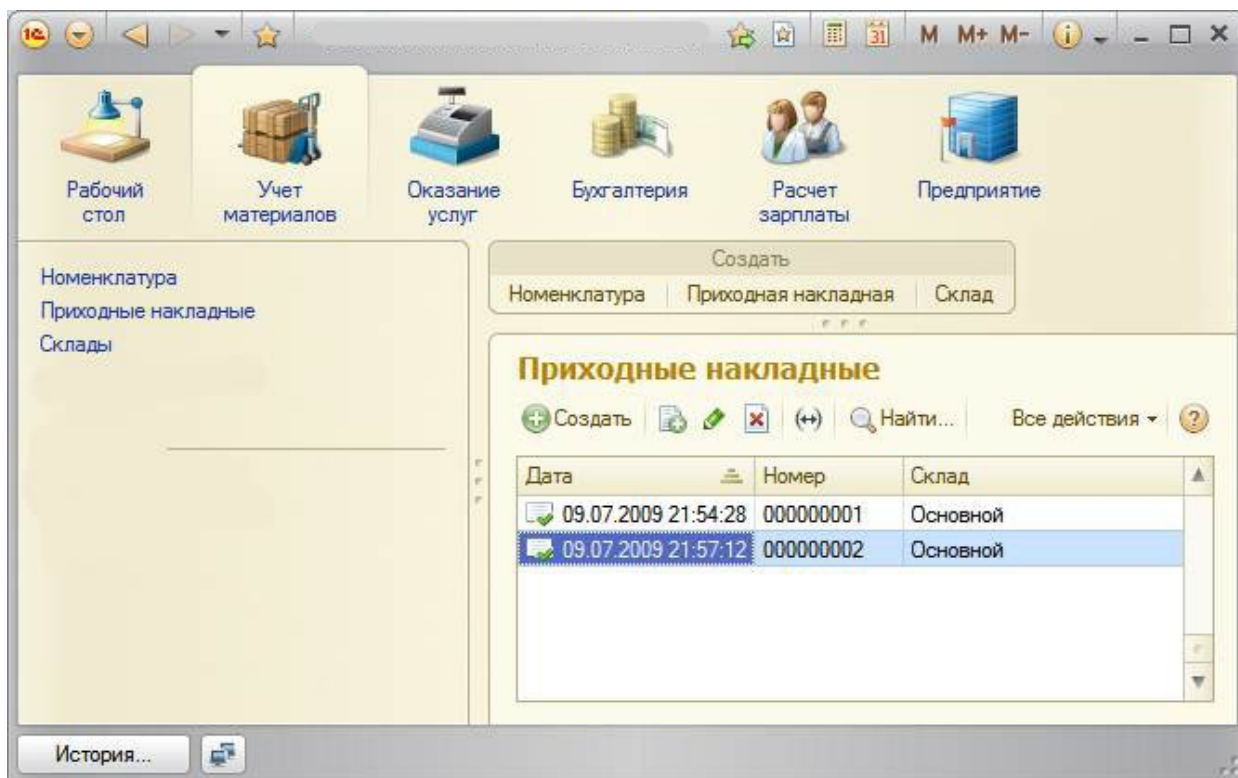


Рис. 4.4. Приходные накладные

Автоматический пересчет суммы в строках документа

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.

Это совсем не сложно, и для этого нам потребуется сначала создать собственную форму документа, а затем воспользоваться возможностями встроенного языка.

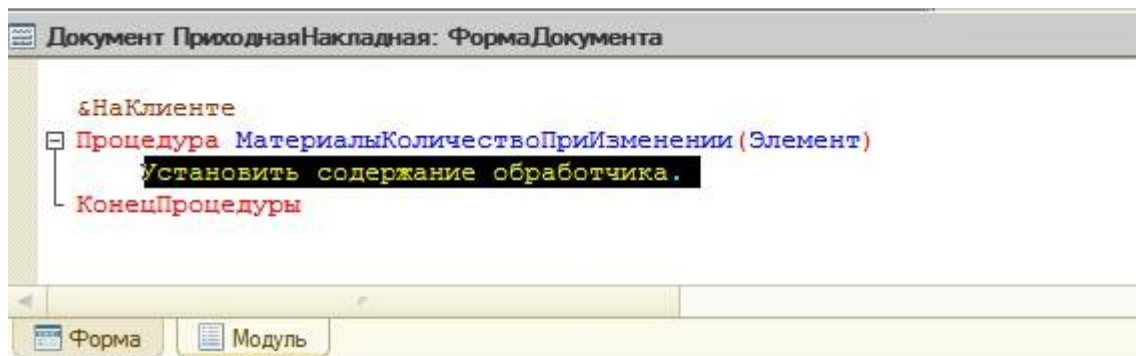
- **В режиме конфигуратор**

Откроем окно редактирования объекта конфигурации Документ

ПриходнаяНакладная, далее пройдем на вкладку **Формы**: чтобы создать форму документа, нажмем кнопку открытия со значком лупы в поле ввода или кнопку **Добавить** над списком форм. Система вызовет полезный инструмент разработчика - **конструктор форм**, в нем выберем тип формы **Форма документа** и нажмем кнопку **Готово**. *Обращаем внимание, что в дереве объектов конфигурации у объекта конфигурации Документ ПриходнаяНакладная появилась форма ФормаДокумента, а на экране открылось окно редактора форм, содержащее эту форму.* Разработчик может через палитру свойств изменить свойства элемента, которые повлияют на его отображение в форме. Он может также изменить структуру элементов формы - создать новое поле, группу полей, добавить табличную часть, связав эти элементы с данными формы.

Используя встроенный язык, разработчик может «вклинить» в эти события и описать собственный алгоритм того, что должно происходить при наступлении этого события.

- Дважды щелкнем на элементе формы **Материалы** **Количество** или правой кнопкой мыши откроем для него палитру свойств: прокрутив список до конца, мы увидим перечень событий, которые могут быть связаны с этим полем. Найдем в списке событий событие **ПриИзменении**, и нажмем на кнопку открытия (значок лупы), система создаст шаблон процедуры обработчика этого события в модуле нашей формы и откроет закладку **Модуль** редактора формы.



Листинг 4.1. Форма документа

Модуль - это «хранилище» для текста программы на встроенном языке. В модуль формы, в процедуру **МатериалыКоличествоПриИзменении()**, добавим следующий текст:

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

Таким образом, в результате выполнения первой строки процедуры обработчика переменная **СтрокаТабличнойЧасти** будет содержать объект **ДанныеФормыСтруктура**. Этот объект содержит данные, находящиеся в текущей строке табличной части документа (**Элементы.Материалы.ТекущиеДанные**). Получив этот объект, мы можем обратиться к данным конкретной колонки табличной части, указав имя колонки в качестве свойства объекта.

Одна процедура для обработки нескольких событий

Теперь хотелось бы и для поля **Цена** и **Сумма** сделать то же самое.

Поэтому лучше будет поместить расчет суммы в некоторое «общедоступное» место, 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.

- **В режиме конфигуратор**

Добавим объект конфигурации **Общий модуль**. Для этого раскроем ветвь **Общие** в дереве объектов конфигурации, нажав на + слева от нее. Затем выделим ветвь **Общие модули** и нажмем кнопку **Добавить** в командной панели окна конфигурации:

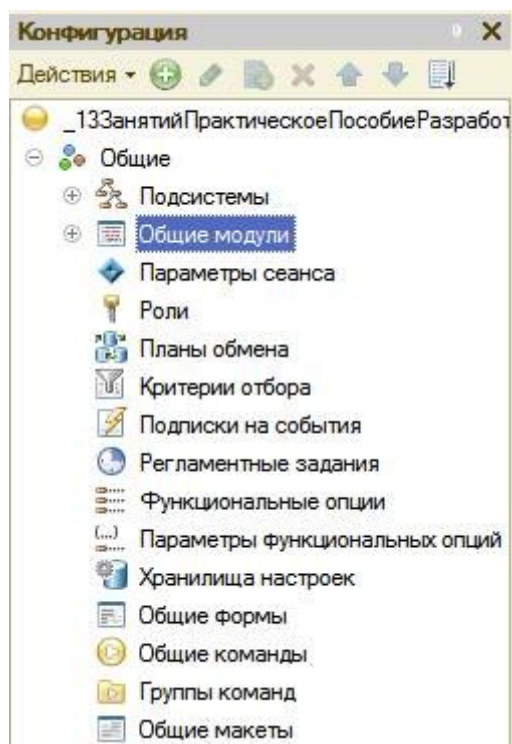


Рис. 4.5. Общие модули

Назовем его РаботаСДокументами и установим в его свойствах флажок Клиент (управляемое приложение), а флажок Сервер снимем. Это означает, что экземпляры этого модуля будут скомпилированы в контексте тонкого клиента и в контексте веб-клиента

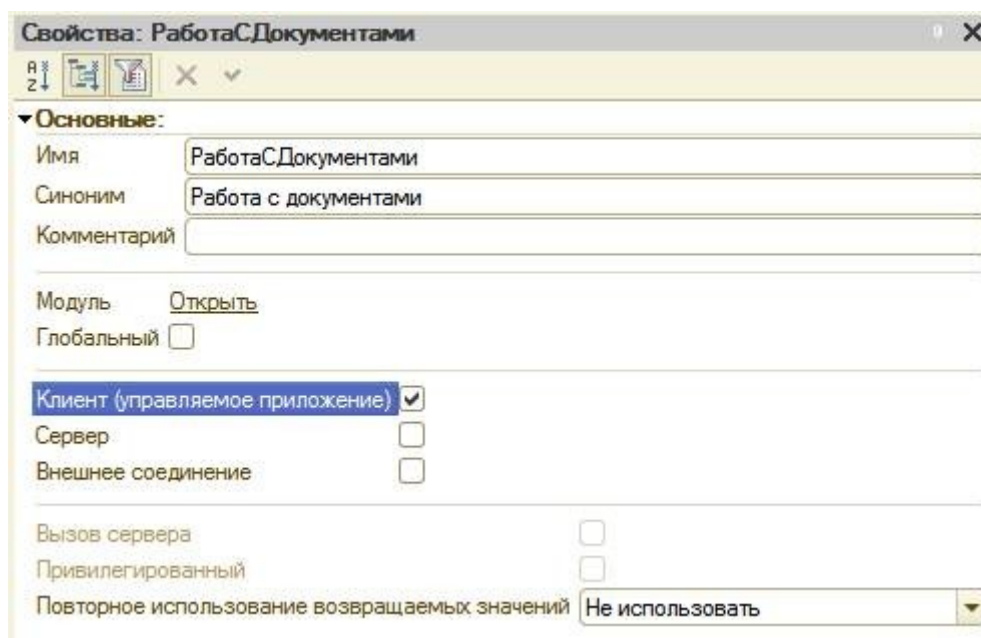


Рис. 4.6. Свойства: РаботаСДокументами

Внесем в общий модуль следующий текст:

Процедура РассчитатьСумму(строкаТабличнойЧасти) Экспорт
СтрокаТабличнойЧасти.Сумма =
СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена;

КонецПроцедуры

Изменим текст обработчика. Для этого вернитесь в модуль формы документа, выберите поле КоличествоПриИзменении.

&НаКлиенте

Процедура МатериалыКоличествоПриИзменении (Элемент)

СтрокаТабличнойЧасти= Элементы.Материалы.ТекущиеДанные;

РаботаСДокументами.РассчитатьСумму(СтрокаТабличнойЧасти);

КонецПроцедуры

Создадим обработчик события ПриИзменении для поля табличной части МатериалыЦена так же, как мы делали это для поля МатериалыКоличество, и повторим в нем вызов процедуры РассчитатьСумму из общего модуля.

- Листинг процедуры «МатериалыЦенаПриИзменении()»

&НаКлиенте

Процедура МатериалыЦенаПриИзменении (Элемент)

СтрокаТабличнойЧасти= Элементы.Материалы.ТекущиеДанные;

РаботаСДокументами.РассчитатьСумму(СтрокаТабличнойЧасти);

КонецПроцедуры

- **В режиме 1С: Предприятие:**

Запустим **1С: Предприятие** в режиме отладки и убедимся, что теперь сумма в строках табличной части документов ПриходнаяНакладная пересчитывается как при изменении количества, так и при изменении цены.

Документ «Оказание услуги»

- **В режиме конфигуратор**

Добавим новый объект конфигурации Документ и назовем его ОказаниеУслуги.

- На закладке **Основные**: Представление объекта задавать не будем, вместо него будет использоваться Синоним объекта. Представление списка зададим как Оказание услуг.

- На закладке **Подсистемы**: отметим, что документ будет доступен в подсистемах Оказание услуг и Бухгалтерия.

- На закладке **Данные**:

создадим реквизиты документа:

- Склад, тип СправочникСсылка.Склады. Выберем для свойства Значение заполнения predetermined элемент Основной справочника Склады.
- Клиент, тип СправочникСсылка.Клиенты. Установим свойство Проверка заполнения в значение Выдавать ошибку.
- Мастер, тип СправочникСсылка.Сотрудники. Установим свойство Проверка заполнения в значение Выдавать ошибку.

Создадим 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;

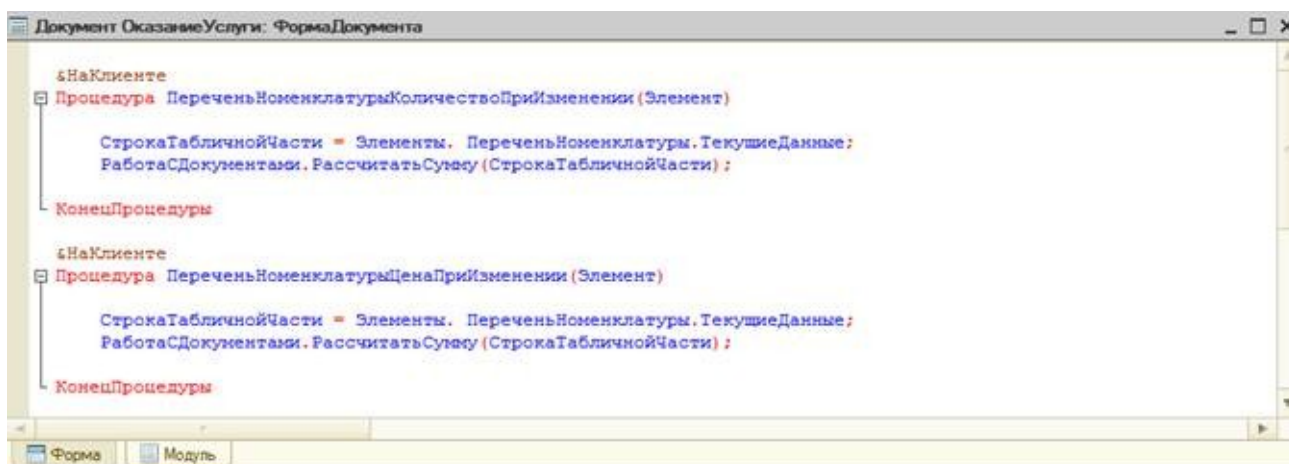
Установим для табличной части в целом и для каждого ее реквизита свойство Проверка заполнения в значение Выдавать ошибку.

- На закладке *Формы*: создадим основную форму документа:

Для поля ПереченьНоменклатурыКоличество создадим обработчик события ПриИзменении, в котором будем вызывать процедуру РассчитатьСумму из общего модуля РаботаСДокументами.

При этом откроется модуль формы с шаблоном обработчика события ПереченьНоменклатурыКоличествоПриИзменении, который мы пока заполнять не будем, а перейдем в окно элементов формы на закладку *Форма* и аналогичным образом создадим обработчик события ПереченьНоменклатурыЦенаПриИзменении для поля ПереченьНоменклатурыЦена.

Далее модуль формы документа ОказаниеУслуги нужно заполнить следующим образом:



Листинг 4.2. Модуль формы документа «ОказаниеУслуги»

В заключение отредактируем командный интерфейс, чтобы в подсистеме Оказание услуг была доступна команда создания новых документов.

Для разнообразия воспользуемся другим способом. Откроем окно редактирования объекта конфигурации Подсистема ОказаниеУслуг и нажмем кнопку Командный интерфейс.

В открывшемся окне отразятся все команды выбранной подсистемы.

В группе Панель действий.Создать включим видимость у команды Оказание услуги: создать

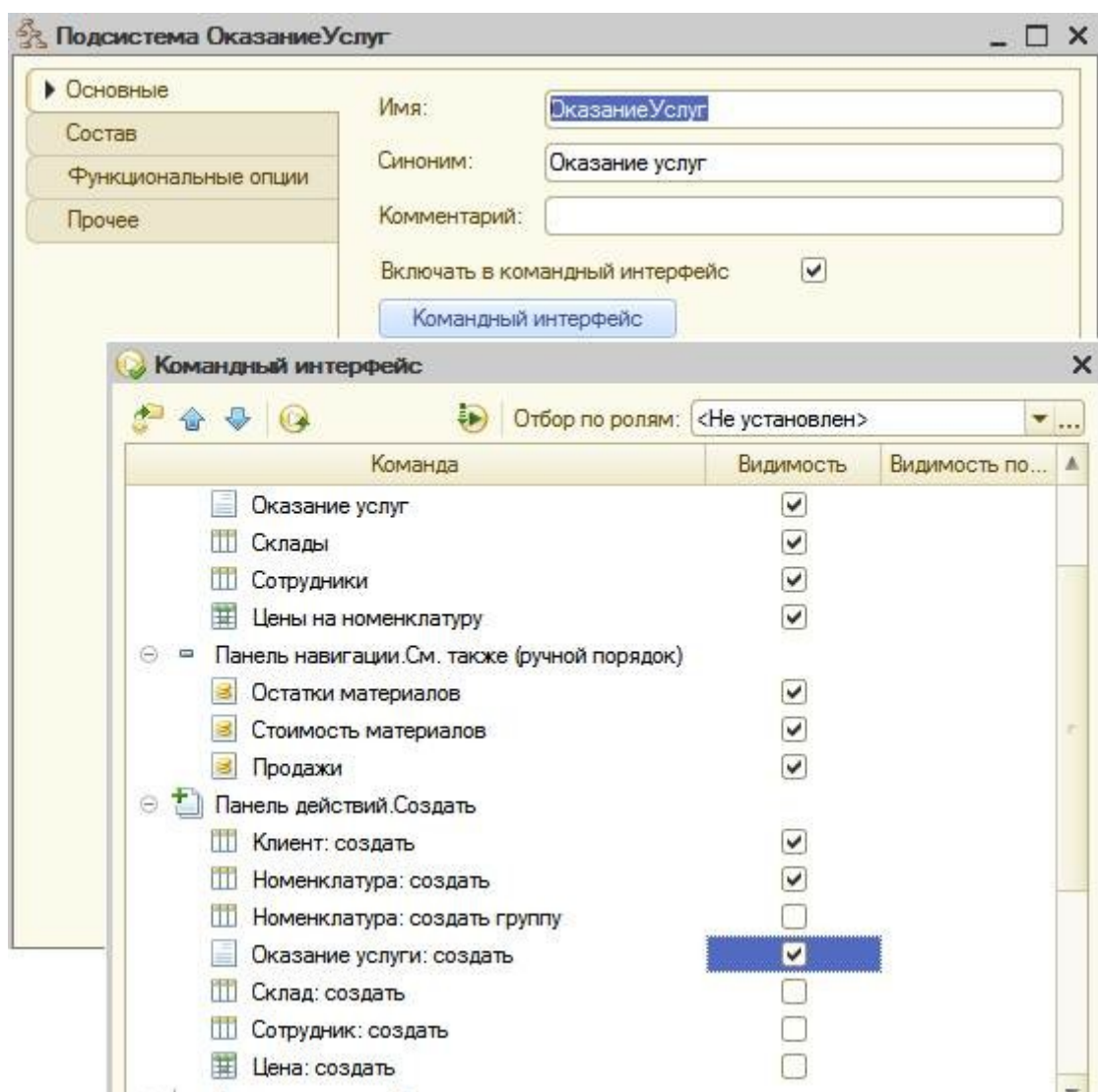


Рис. 4.7. Видимость у команды Оказание услуги: создать

В режиме 1С:Предприятие

В панели действий раздела Оказание услуг вызовем команду создания документа Оказание услуги и заполним его следующим образом:

Контрольные вопросы

1. Для чего используется объект конфигурации Подсистема.
2. Как описать логическую структуру конфигурации при помощи объектов Подсистема.
3. Как управлять порядком вывода и отображением подсистем в конфигурации.
4. Что такое окно редактирования объекта конфигурации и в чем его отличие от палитры свойств.
5. Для чего предназначен объект конфигурации Справочник.
6. Каковы характерные особенности справочника.
7. Для чего используются реквизиты и табличные части справочника.
8. Зачем нужны иерархические справочники и что такое родитель.
9. Зачем нужны подчиненные справочники и что такое владелец.
10. Какие основные формы существуют у справочника.
11. Что такое predetermined элементы.
12. Чем с точки зрения конфигурации отличаются обычные элементы справочника от predetermined элементов.
13. Как пользователь может отличить обычные элементы справочника от predetermined элементов.
14. Как создать объект конфигурации Справочник и описать его структуру.
15. Как добавить новые элементы в справочник.
16. Как создать группу справочника.
17. Как переместить элементы из одной группы справочника в другую.
18. Зачем нужна основная конфигурация и конфигурация базы данных.
19. Как изменить конфигурацию базы данных.
20. Как связаны объекты конфигурации и объекты базы данных.
21. Что такое подчиненные объекты конфигурации.
22. Зачем нужна проверка заполнения у реквизитов справочника.
23. Что такое быстрый выбор и когда его использовать.
24. Как отобразить справочник и определить его представление в различных разделах интерфейса приложения.
25. Как отобразить команды создания нового элемента справочника в интерфейсе подсистем.
26. Как редактировать командный интерфейс подсистем.
27. Какими характерными особенностями обладает документ.
28. Для чего предназначены реквизиты и табличные части документа.
29. Какие существуют основные формы документа.
30. Что такое проведение документа.
31. Как создать объект конфигурации Документ и описать его основную структуру.
32. Как создать новый документ и заполнить его данными.
33. Как создать собственную форму документа.
34. Что такое конструктор форм.
35. Что такое редактор форм.
36. Что такое элементы формы.
37. Что такое события и с чем они связаны.
38. Что такое обработчик события и как его создать.
39. Что такое модуль и для чего он нужен.
40. Зачем нужны общие модули.
41. Что такое типобразующие объекты.