Оглавление

Редакторы команд 3

Как использовать общий модуль 4

Свойства общих модулей 4

Параметры сеанса в 1С 8.3 и 8.2 6

Использование параметров сеанса в 1С 6

Роль 7

Редакторы ролей 8

Общий реквизит 9

Разделение данных 9

Хранение однотипных данных 9

План обмена 11

Назначение 11

Механизмы, реализуемые планом обмена 12

Критерии отбора 1С 14

Подписка на событие 17

Регламентное задание 18

Функциональные опции 20

Параметр функциональной опции 22

Определяемые типы в 1С 8.3 24

Что такое определяемые типы 24

Настройка 24

Хранилище настроек 25

Основные формы 26

Автогенерируемые формы 27

Связь формы с данными 27

Структура формы 28

Механика работы форм 32

Редактор форм 33

Общие команды и группы команд в 1С 8.3 34

Настройка и свойства общей команды в 1С 34

Группы команд 1С 35

Объекты 1С "Общие макеты" 36

Типы макетов 1С: 36

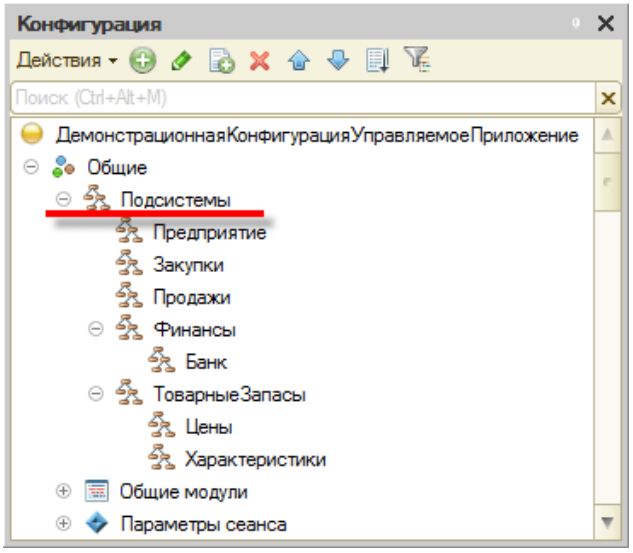
Общие картинки 37
XDTO пакет 40
Web-сервисы 41
Техническая реализация web-сервисов 42
HTTP-сервисы 45
Язык 51
Константа 54
Форма констант 54
Справочники 56
Структура справочника 56
Формы справочника 59
Форма списка 60
Форма элемента 60
Форма группы 61
Форма выбора, форма выбора группы 61
Макеты 62
Документы 63
Структура документа 63
Проведение документа 65
Конструктор движений 67
Формы документа 67
Форма списка 68
Форма документа 69
Форма выбора 69
Макеты документа 69
Журнал документов 71
Перечисления 73
Отчет 74
Выполнение отчетов в фоновом режиме 74
Обработка 75
План видов характеристик 76
Структура 76
Формы плана видов характеристик 78

Форма списка 79
Форма характеристики 80
План счетов 81
Структура 81
Формы плана счетов 83
Форма списка 84
Форма счета 85
Форма выбора 86
План видов расчета 87
Структура 87
Формы плана видов расчета 88
Форма списка 89
Форма вида расчета 89
Регистр сведений 91
Структура 91
Периодичность 92
Подчинение регистратору 93
Уникальность записей 94
Формы 95
Форма списка 95
Форма записи 95
Функциональные возможности регистра сведений 96
Регистр накопления 97
Структура 97
Связь с регистратором 99
Конструктор движений 100
Уникальность записей 100
Регистры остатков и регистры оборотов 100
Агрегаты 101
Форма списка и форма набора записей 101
Функциональные возможности регистра накопления 101
Регистр бухгалтерии 103
Структура 103

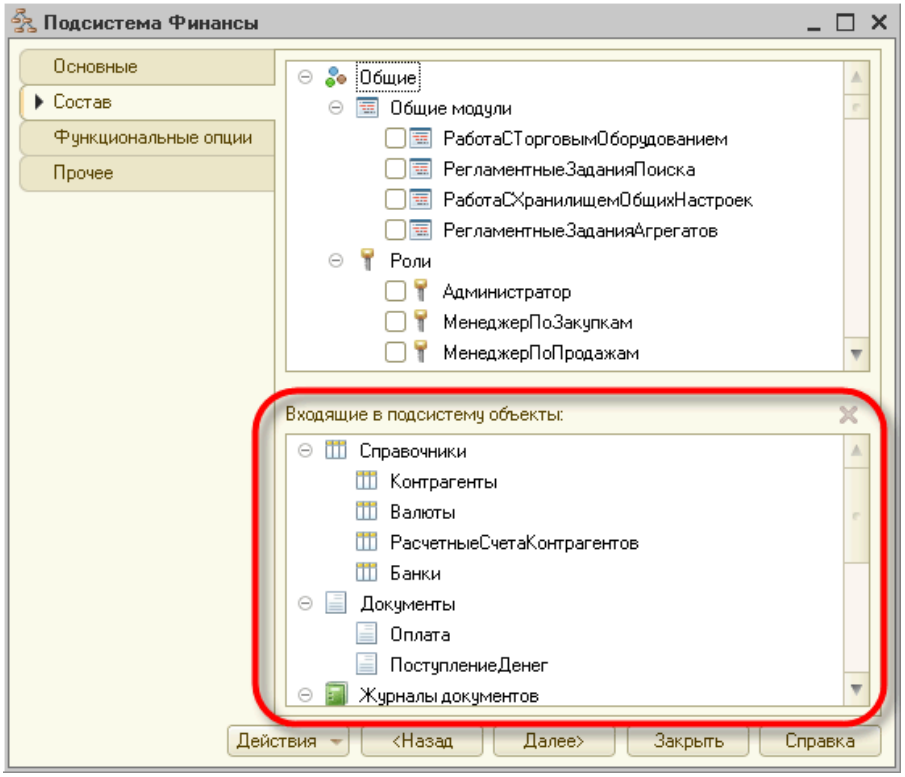
Связь с регистратором 105
Конструктор движений 106
Уникальность записей 106
Форма списка 106
Функциональные возможности регистра бухгалтерии 107
Регистр расчета 109
Структура 109
Связь с планом видов расчета 110
Периодичность 110
Подчинение регистратору 111
Связь с графиком времени 111
Перерасчеты 113
Уникальность записей 114
Механизмы, реализуемые регистром расчета 114
Вытеснение по периоду действия 114
Зависимость по базовому периоду 114
Формирование записей перерасчета 115
Функциональные возможности регистра расчета 115
Бизнес-процесс 117
Карта маршрута 117
Точка действия 118
Групповая и коллективная маршрутизация 118
Персональная и ролевая маршрутизация 118
Условная маршрутизация 119
Использование в прикладных решениях 120
Задача 121
Внешний источник данных 122

Подсистемы — это общие объекты конфигурации. На их основе платформа формирует командный интерфейс прикладного решения и визуально разделяет всю функциональность программы на крупные и мелкие блоки.

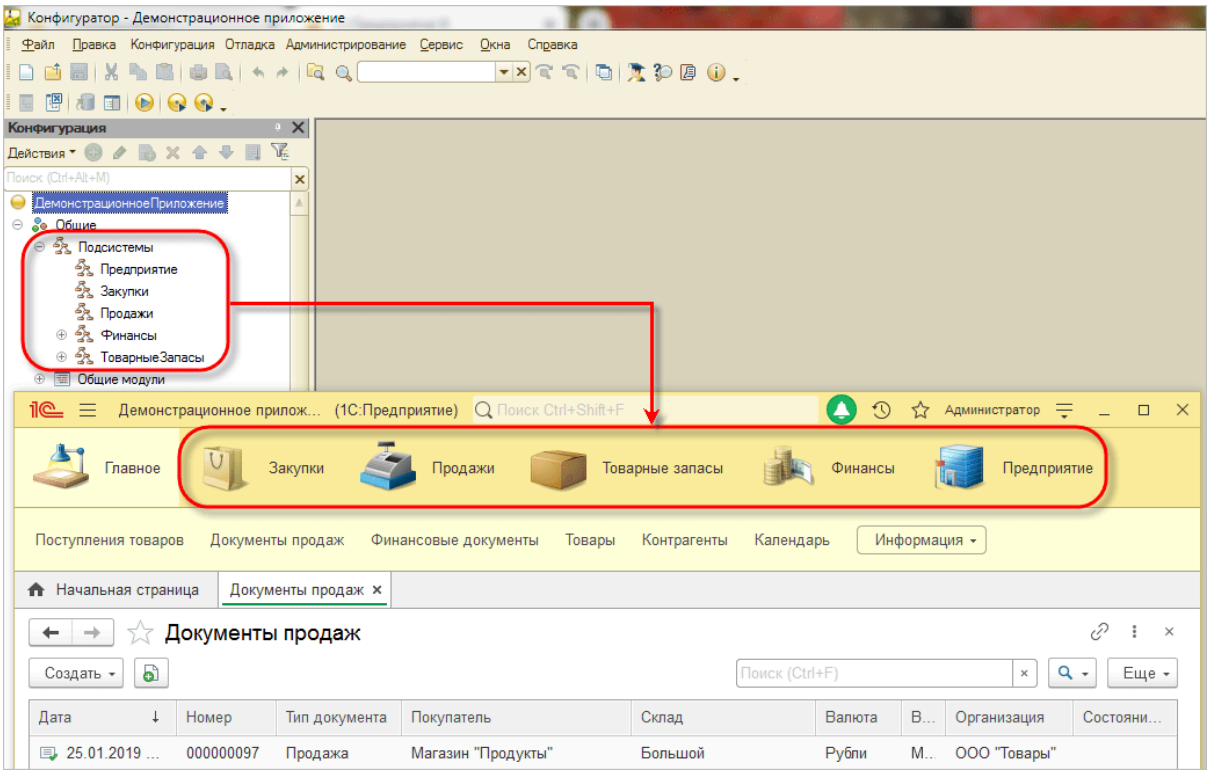
Подсистемы могут иметь иерархическую структуру, т. е. одна подсистема может включать в себя несколько других подсистем:



Каждый объект конфигурации можно включить в состав одной или нескольких подсистем. Таким образом, в терминах подсистем можно описать всю структуру прикладного решения:



Разделы прикладного решения, по которым пользователь осуществляет основную навигацию, однозначно соответствуют первому уровню подсистем конфигурации:

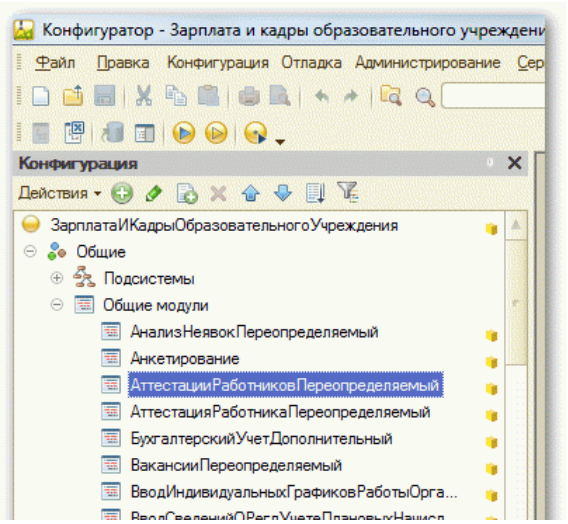


Редакторы команд

Для редактирования команд, отображаемых в тех или иных разделах интерфейса, платформа содержит целый набор различных редакторов:

- **Редактор командного интерфейса конфигурации**
  Этот редактор предназначен для настройки панели разделов. Он позволяет задать состав, порядок разделов и их видимость в разрезе ролей. Подробнее..
- **Редактор командного интерфейса основного раздела**
  Этот редактор предназначен для настройки команд основного раздела. Для панели функций текущего раздела он позволяет задать состав команд, их порядок и видимость команд в разрезе ролей. Подробнее…
- **Редактор командного интерфейса**
  Этот редактор предназначен для настройки команд какой-либо подсистемы. Он позволяет задать состав команд, их порядок и их видимость в разрезе ролей для панели навигации и панели действий выбранной подсистемы. Подробнее…
- **Редактор «Все подсистемы»**
  Этот редактор позволяет работать одновременно со всеми подсистемами конфигурации. Подробнее…

Общие модули 1С — объект метаданных конфигурации 1С 8.3, который хранит в себе программный код, который часто вызывается в конфигурации. Функцию/процедуру можно вызвать из любого места конфигурации (если она экспортная).



Как использовать общий модуль

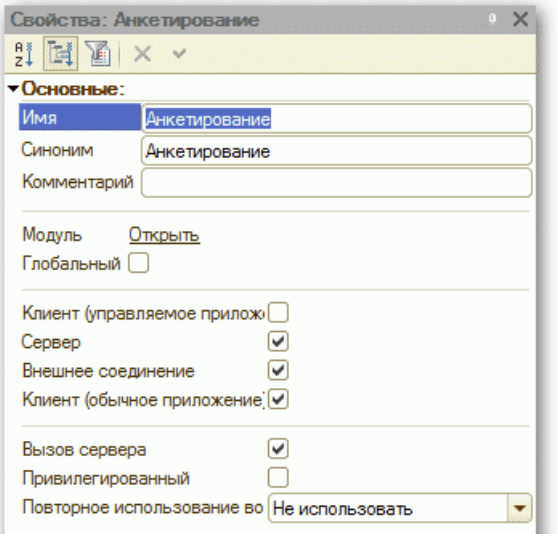
Хороший тон — поместить процедуру или функцию в общий модуль, если она вызывается в более чем одном месте. Во-первых, если процедура корректируется, её надо править только в одном месте. Во-вторых, этим достигается больший порядок в коде.

Типичный пример общего модуля — обработка проведения по какому-то регистру, получение количества разницы рабочих дней, пересчет курсов валют, пересчет количества/цены/суммы в табличной части и другие функции.

Свойства общих модулей

Одно из основных особенностей общих модулей от других модулей — нельзя объявлять общие переменные.

Рассмотрим подробнее палитру свойств общего модуля:



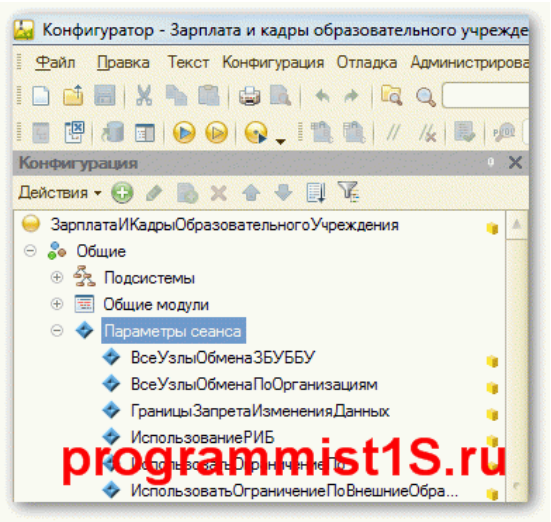
- **Глобальный** — если флаг установлен, функции и процедуры из этого модуля становятся доступны в глобальном контексте. Т.е. их можно вызвать в любом месте конфигурации, обращаясь без названия общего модуля. Однако

добавляется условие — название процедур и функций в этом общем модуле должны быть уникальны в рамках глобального контекста.

- **Сервер** — процедуры и функции данного общего модуля могут быть выполнены на сервере.
- **Внешнее соединение** — программные коды данного общего модуля могут быть выполнены при подключении внешним источником (например, COM).
- **Клиент (управляемое приложение)** — процедуры и функции данного общего модуля могут быть использованы в толстом клиенте в режиме управляемого приложения.
- **Клиент (обычное приложение)** — программные коды данного общего модуля могут быть использованы в толстом клиенте в режиме обычного приложения.
- **Вызов сервера** — флаг, разрешающий на клиенте использовать процедуры и функции из этого общего модуля.
- **Привилегированный** — если установлена Истина, в этом общем модуле будет отключена проверка прав доступа.
- **Повторное использование** — определяет настройки возвращаемых значений, если опция включена, то после первого выполнения система запомнит значение для данных входных параметров и будет возвращать уже готовое значение. Может принимать следующие значения: **не используется** — отключение, **на время вызова** — на время выполнения определенной процедуры, **на время сеанса** — пока пользователь не закрыл сеанс (программу).

Параметры сеанса в 1С 8.3 и 8.2

Параметры сеансы 1С 8.3 — переменная, в которой хранится значение нужного параметра на время сеанса пользователя. По сути, это некая глобальная переменная, привязанная к сеансу текущего пользователя.

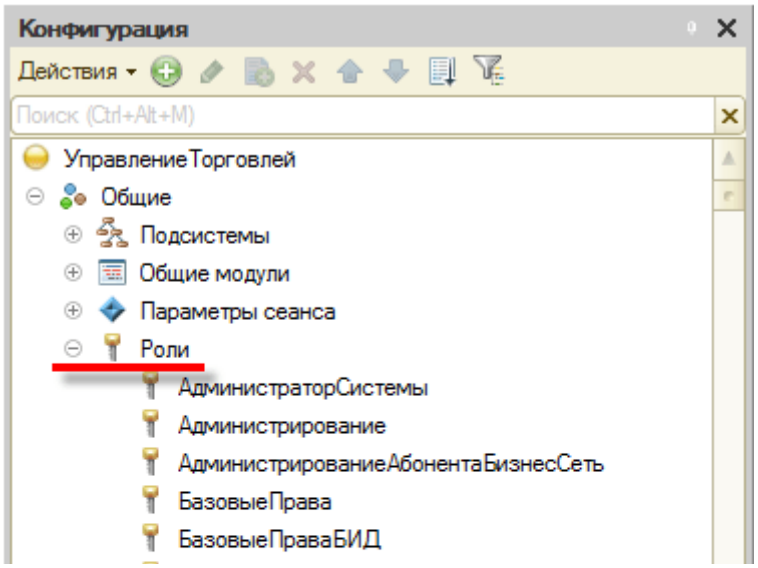


Использование параметров сеанса в 1С

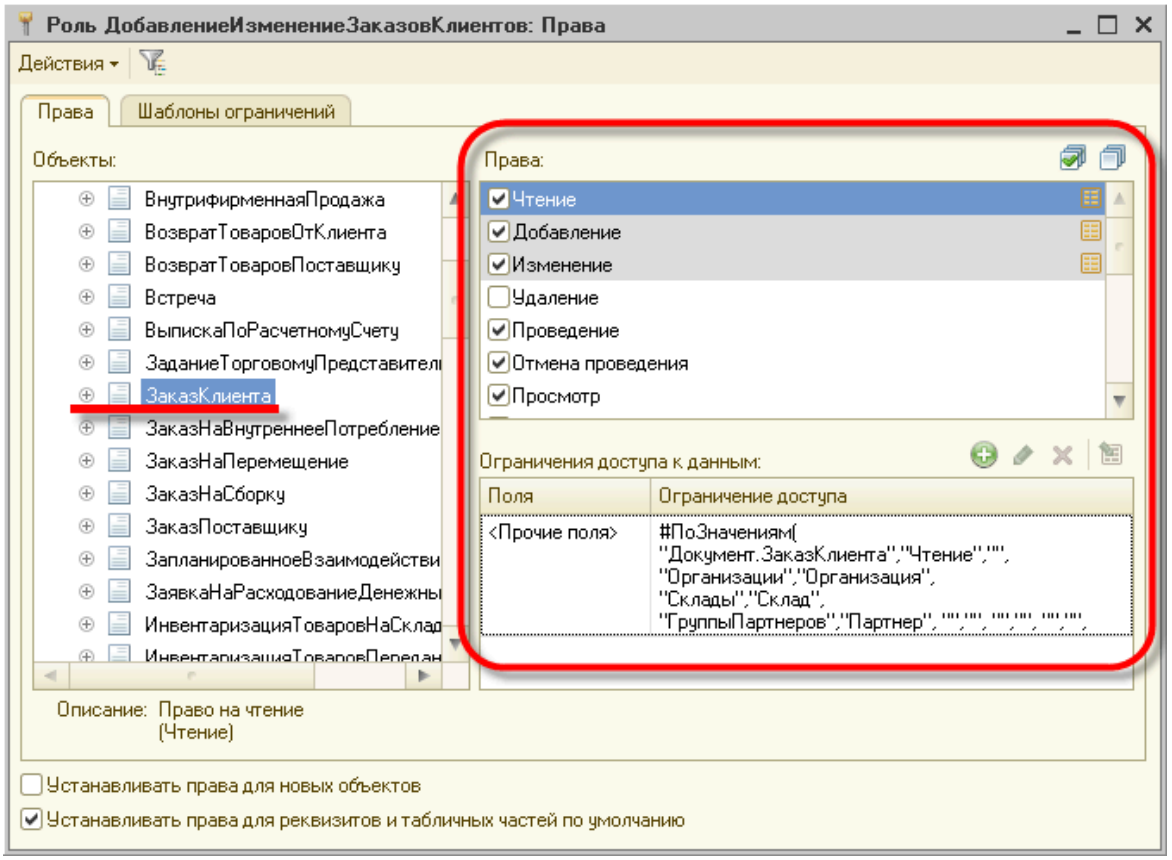
Параметры сеанса задаются только программно, универсального интерфейса для установки параметров сеанса в системе нет. Обычно они задаются при старте системы, в «Модуле сеанса». Если параметр не определен, во время обращения к нему будет вызвана ошибка.

Роль

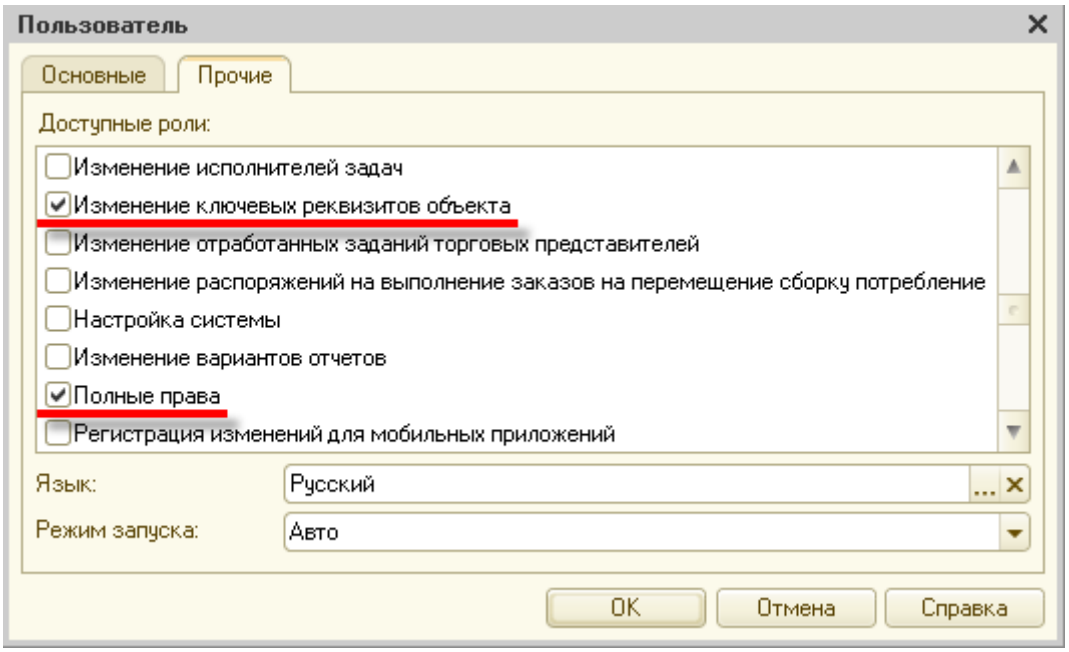
Роли — это общие объекты конфигурации. Они предназначены для реализации ограничения прав доступа в прикладных решениях. Роль в конфигурации может соответствовать должностям или видам деятельности различных групп пользователей, для работы которых предназначена данная конфигурация:



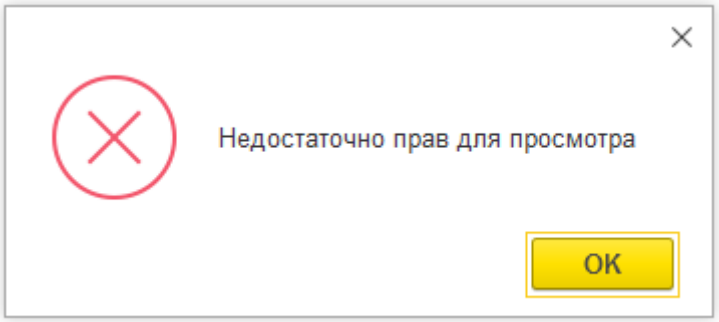
Роль определяет, какие действия, над какими объектами метаданных может выполнять пользователь, выступающий в этой роли:



В процессе ведения списка пользователей прикладного решения каждому пользователю ставится в соответствие одна или несколько ролей.



При попытке пользователя выполнить действие, на которое у него нет разрешения, действие выполнено не будет, а система выдаст окно предупреждения:



Редакторы ролей

Для редактирования состава ролей платформа содержит два редактора:

- **Редактор роли**
  Этот редактор позволяет состав прав для одной выбранной роли. По дробнее…

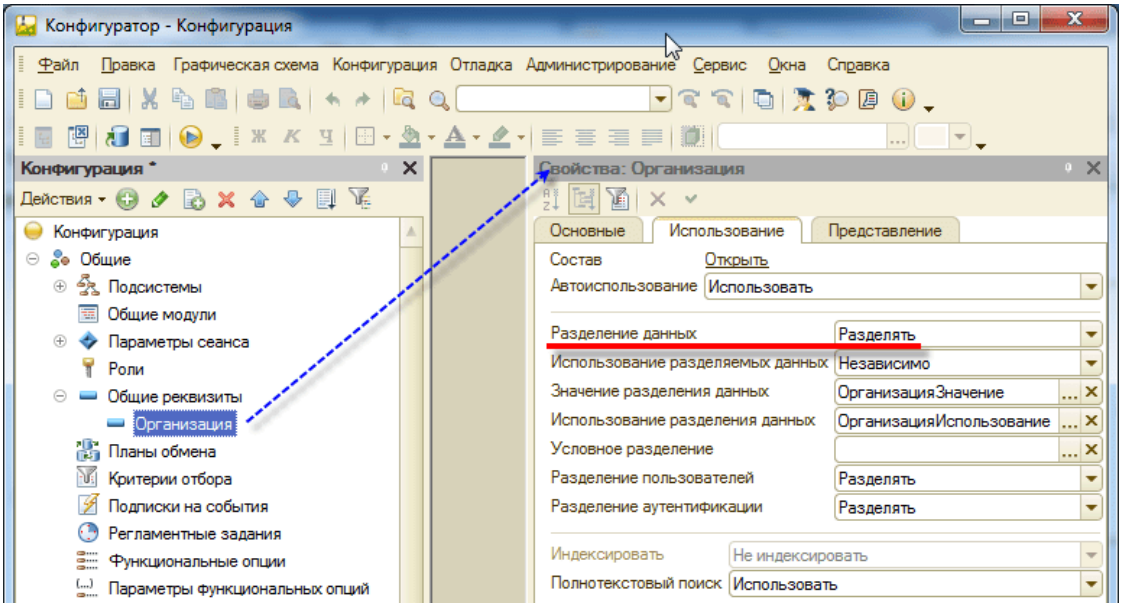
- **Редактор «Все роли»**
  Этот редактор позволяет изменять и анализировать состав прав сразу для нескольких или для всех ролей, существующих в прикладном решении.

Общий реквизит

Общие реквизиты — это общие объекты конфигурации. Они используются для двух различных целей.

Разделение данных

Во-первых, общие реквизиты используются в механизме разделения данных для того, чтобы идентифицировать независимые области данных, создаваемые внутри информационной базы. Для этого используется свойство общего реквизита **Разделение данных**.



Область данных — это множество данных, доступных в текущем сеансе на основании факта использования разделителей и на основании значений разделителей, определенных в конфигурации.

Данными, которые могут быть отнесены к той или иной области, являются:

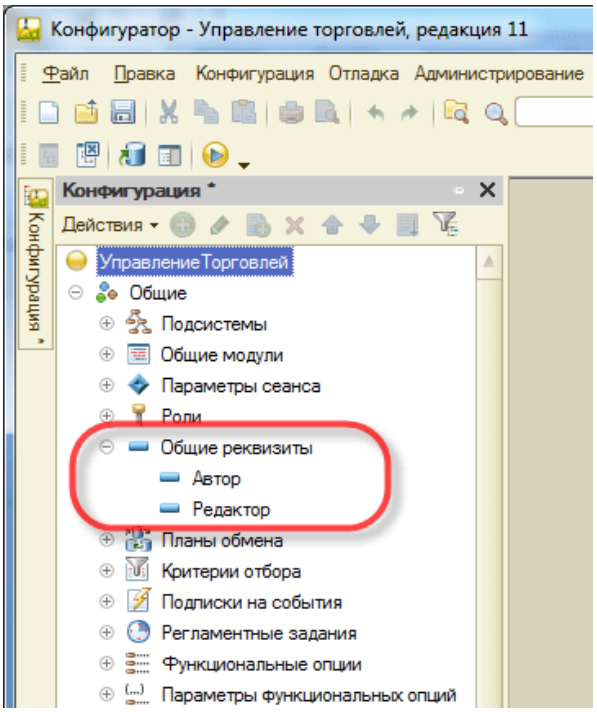
- данные базы данных;
- пользователи информационной базы;
- записи журнала регистрации;
- фоновые и регламентные задания;
- часовой пояс информационной базы;
- системные и пользовательские настройки;
- история;
- активные пользователи;
- другие данные.

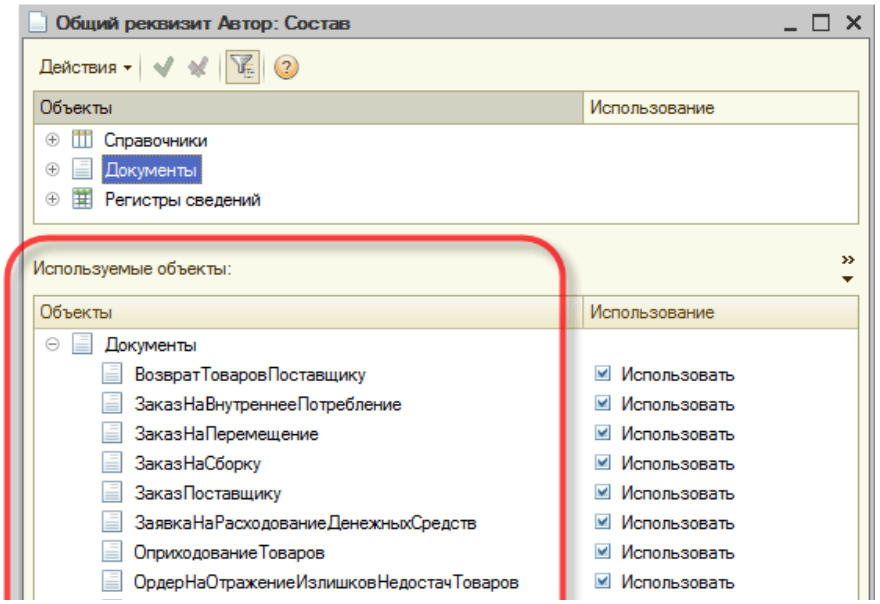
> Механизм разделения данных позволяет хранить данные нескольких независимых организаций в одной информационной базе. Это становится возможным благодаря тому, что общие реквизиты объектов конфигурации можно использовать не только как «одинаковый реквизит, который есть у всех объектов», но и как идентификатор того, что данные относятся к какой-то одной из нескольких независимых областей.

Хранение однотипных данных

Во-вторых, общие реквизиты используются для того, чтобы автоматически добавить нескольким или всем объектам конфигурации одинаковый реквизит с одним и тем же именем и одинаковым типом.

Например, если во всех документах требуется иметь реквизиты **Автор** и **Редактор**, нет необходимости создавать такие реквизиты в каждом документе. Достаточно создать два общих реквизита (**Автор, Редактор**) и указать, в каких документах они будут использоваться.





План обмена

Планы обмена — это общие объекты конфигурации. Они используются для реализации механизмов обмена данными.

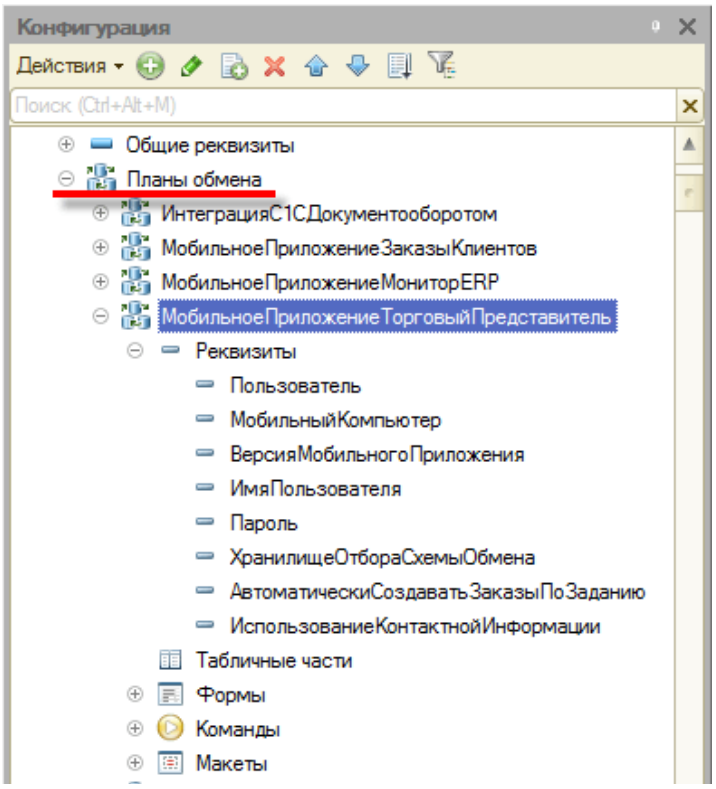
План обмена содержит информацию об узлах, которые могут участвовать в обмене данными, определяет состав данных, которыми будет производиться обмен, и указывает, следует ли задействовать механизм распределенной информационной базы при обмене.

В одном прикладном решении может существовать несколько планов обмена, каждый из которых может описывать свой порядок обмена данными. Например, если выполняется обмен данными с удаленными складами и удаленными офисами, то, скорее всего, будет существовать два плана обмена (один для обмена со складами, другой — для офисов), поскольку состав данных, которыми производится обмен со складами, будет значительно «уже», чем состав данных, предназначенных для обмена с офисами.

Назначение

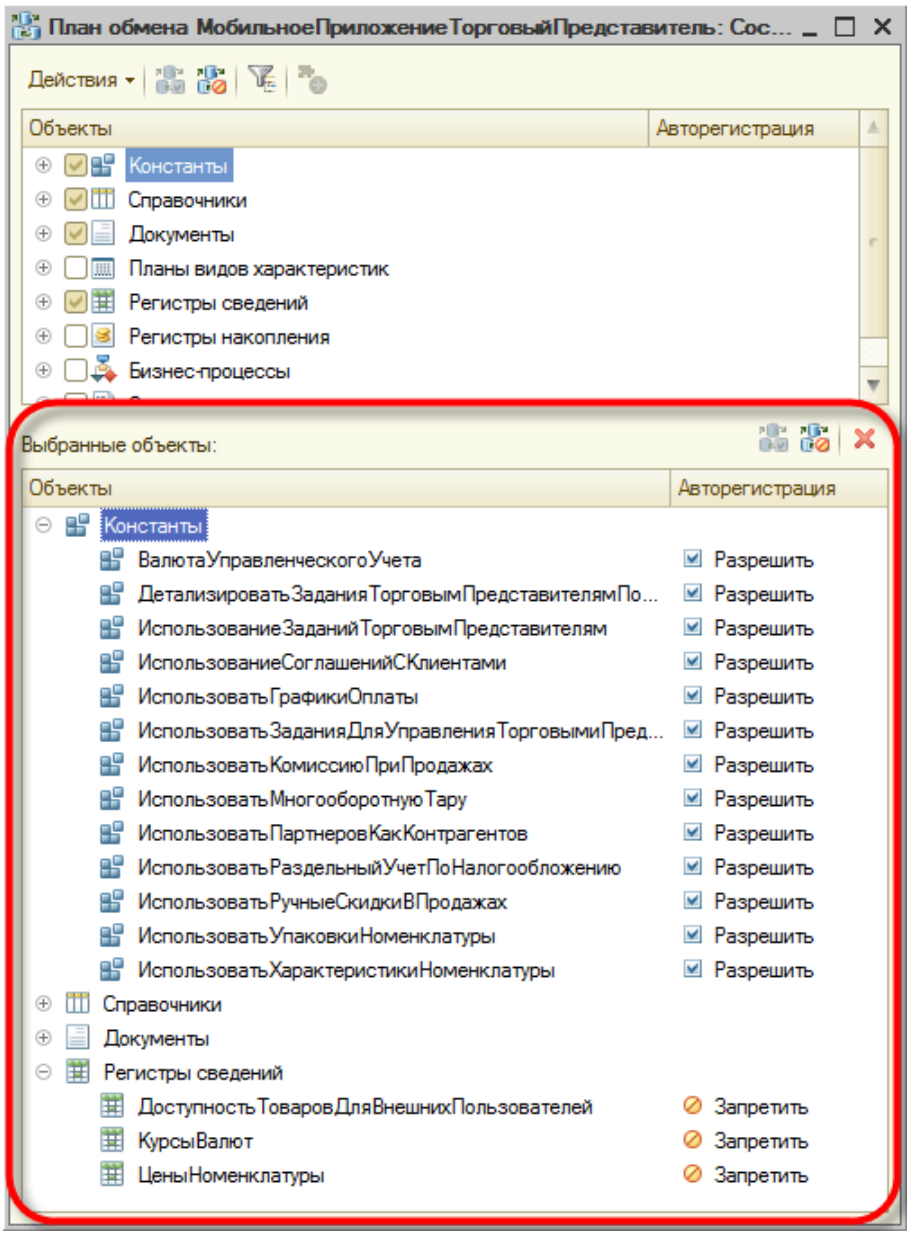
В плане обмена хранится список узлов, — участников обмена в распределенной информационной системе. В качестве узлов могут выступать информационные базы 1С:Предприятия 8, информационные базы 1С:Предприятия 7.7 или другие информационные системы, не основанные на 1С:Предприятии.

Для каждого узла можно задать код, наименование и необходимый перечень **реквизитов**, описывающих узел. Узел может иметь также несколько подчиненных табличных частей для хранения информации, связанной с этим узлом, несколько форм, для отображения информации, содержащейся в плане обмена, и т. д. Например, структура плана обмена **МобильноеПриложениеТорговыйПредставитель** может выглядеть следующим образом:



При создании плана обмена существует возможность указать, будет ли он задействовать механизмы распределенной информационной базы или нет.

Также в плане обмена указывается состав данных, которыми предполагается вести обмен:



Для каждого из объектов прикладного решения, которые могут участвовать в обмене, задается режим регистрации их изменений. Если разрешена авторегистрация, то система сама отслеживает выполняемые изменения, и формирует набор данных, которыми нужно обменяться. Если же разработчик запрещает авторегистрацию изменений, — он должен, средствами встроенного языка, самостоятельно выполнять регистрацию нужных ему изменений.

Механизмы, реализуемые планом обмена

- **Механизм распределенных информационных баз**
  Позволяет создавать в рамках конкретного плана обмена распределенную информационную

базу. Распределенная информационная база представляет собой иерархическую структуру, состоящую из отдельных информационных баз 1С:Предприятия — узлов распределенной информационной базы, между которыми организован обмен данными с целью синхронизации конфигурации и данных. Подробнее…

- **Служба регистрации изменений**
  Суть регистрации изменений состоит в том, чтобы иметь перечень измененных элементов данных которые должны быть переданы в очередном сообщении тому или иному узлу, с которым производится обмен данными. При каждом изменении данных регистрируется, что имеются изменения, которые предстоит передать во все узлы, с которыми поддерживается обмен этими данными. При получении подтверждения приема сообщения, в котором были отправлены изменения, записи регистрации изменений должны быть удалены. Подробнее…

- **Инфраструктура сообщений**
  С точки зрения плана обмена, между узлами происходит обмен сообщениями. Каждое сообщение содержит изменения данных, изменения конфигурации (если это распределенная информационная база) и ряд служебной информации. Каждое сообщение точно ассоциировано с планом обмена, имеет уникальный номер и имеет одного отправителя и одного получателя.

Критерии отбора 1С – это объект метаданных, позволяющий осуществить отбор информации в базе 1С по заданным условиям. Это, своего рода, предопределенные правила, по которым в пользовательском режиме в списках осуществляется отбор информации по какому-либо значению. Например, установив отбор по контрагенту, можно найти любые документы (или справочники), в которых фигурирует указанный контрагент.

Отбор в журнале по полю «Контрагент»:

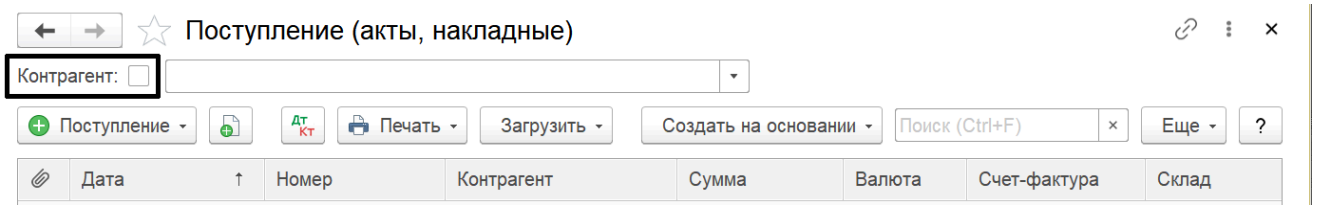


Рис.1 Отбор в журнале по полю «Контрагент»

Этот прием используется и при формировании структуры подчиненности, которую можно увидеть, воспользовавшись кнопкой на панели инструментов «Перейти к иерархическому списку связанных документов».

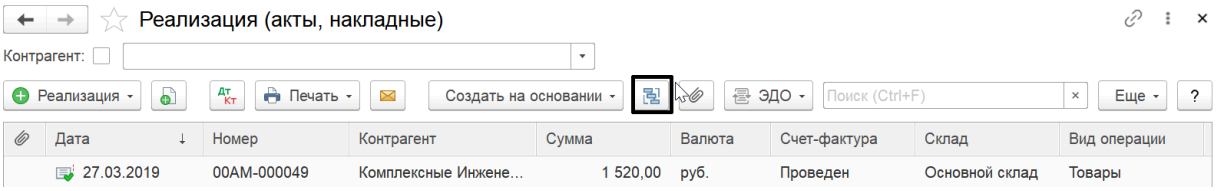


Рис.2 Иерархический список связанных документов

Разберем, что представляют собой связанные критерии отбора 1С на примере 1С:Бухгалтерия предприятия версии 3.0.

Откроем программу в режиме «Конфигуратор» и, сделав это, найдем в дереве метаданных нужный раздел критериев.

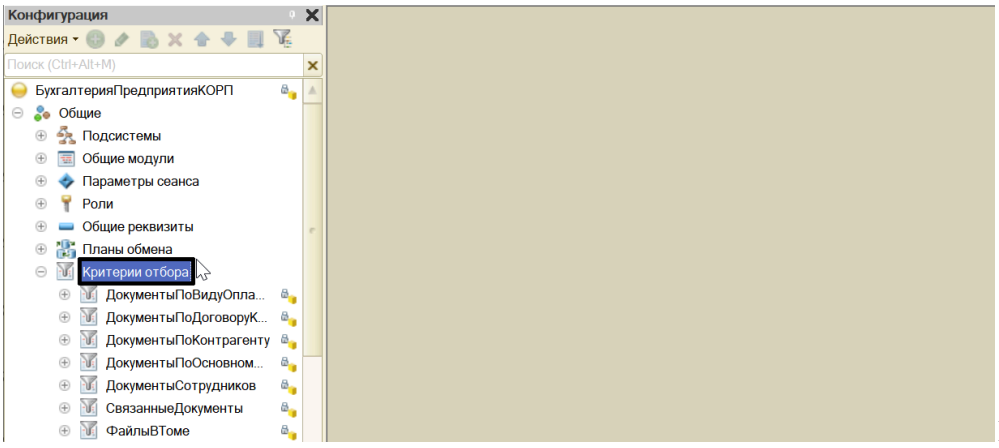


Рис.3 Нужный раздел критериев

Настройки

В «Данных» задается «Тип» – это тип значения для отбора, по которому и будет осуществляться поиск. К примеру, «ДокументыПоКонтрагенту» имеет тип СправочникСсылка.Контрагенты, а «ДокументыСотрудников», имеют тип СправочникСсылка.ФизическиеЛица.

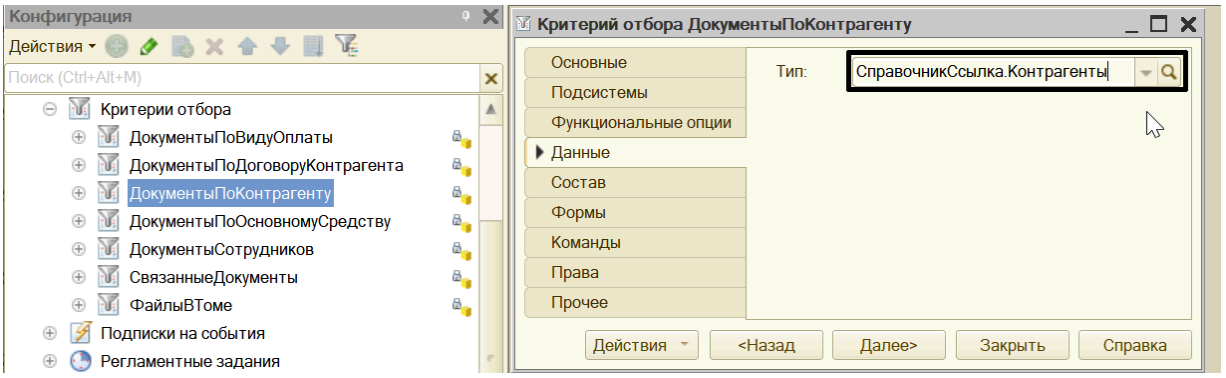


Рис.4 Настройки

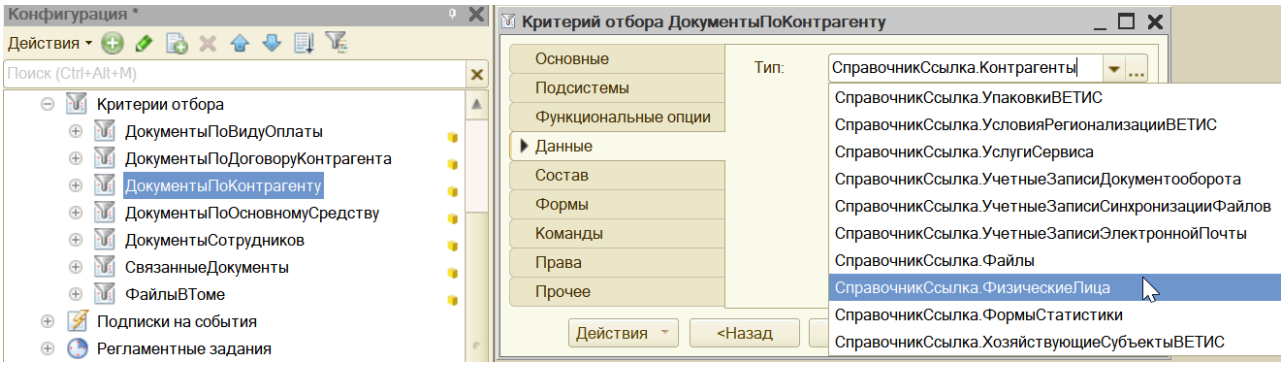


Рис.5 Продолжение настроек

При этом можно задать любой тип из стандартных или определить объект из дерева конфигурации, указав любую ссылку.

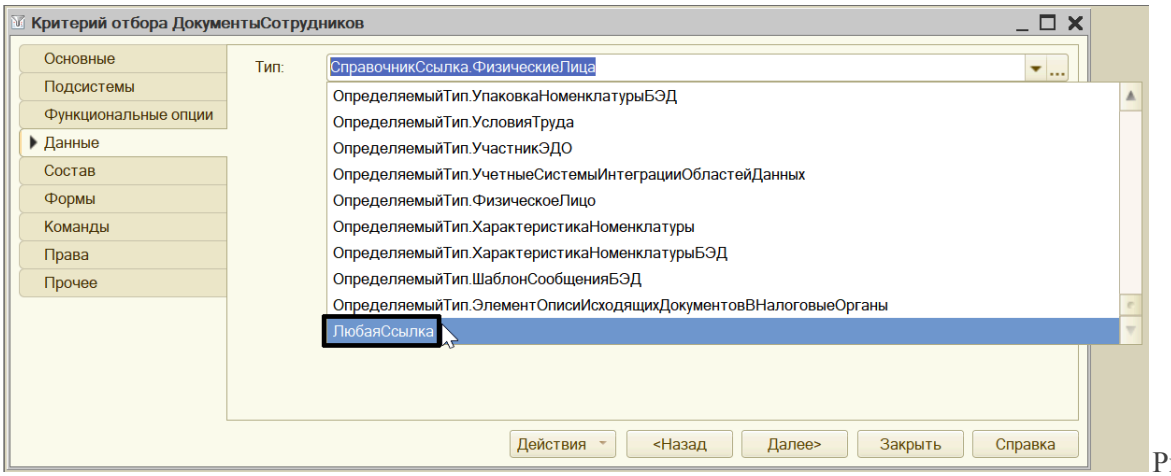


Рис.6 Выбор типа

Мы указываем тот тип, по которому будет производиться отбор. Преимущество критерия отбора заключается в том, что он может использовать также реквизиты табличных частей. Так, например, мы можем выбрать все документы с определенной номенклатурой в табличной части.

На закладке «Состав» отображаются те элементы, в которых содержатся данные указанного типа. В этом окне необходимо установить галочки для тех элементов, по которым будет осуществляться отбор. Только после определения этого списка, критерий отбора будет выполнять свою функцию.

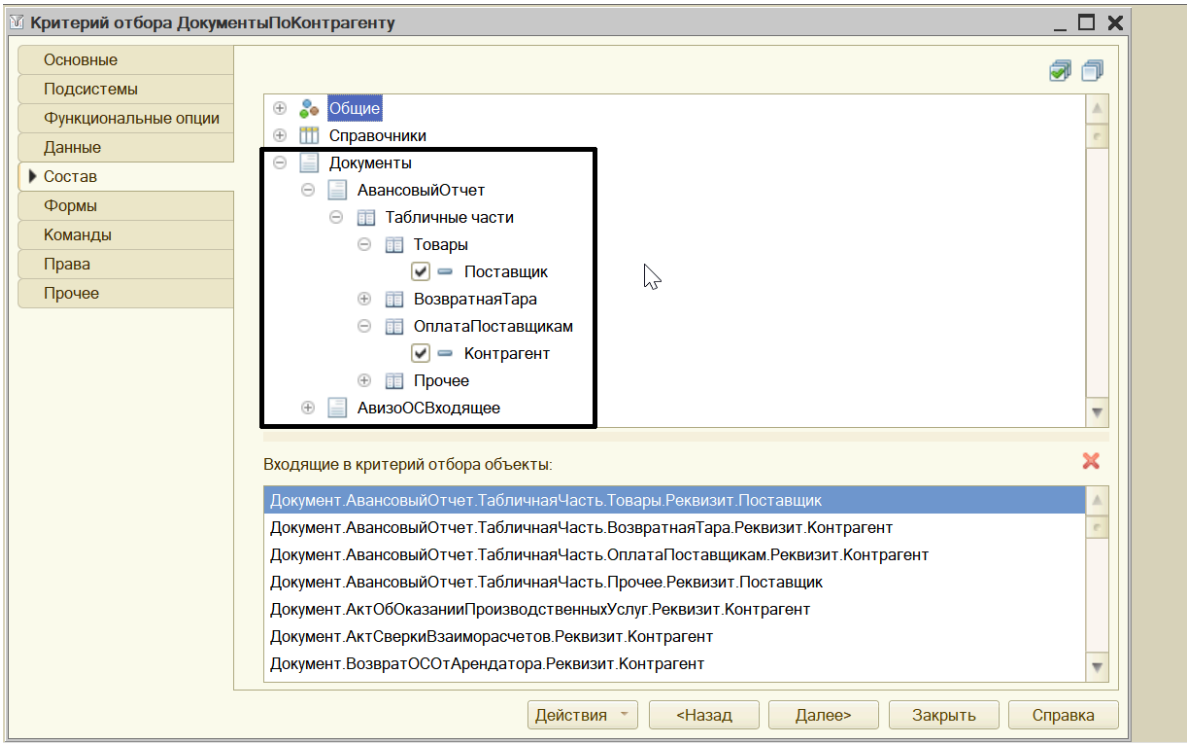
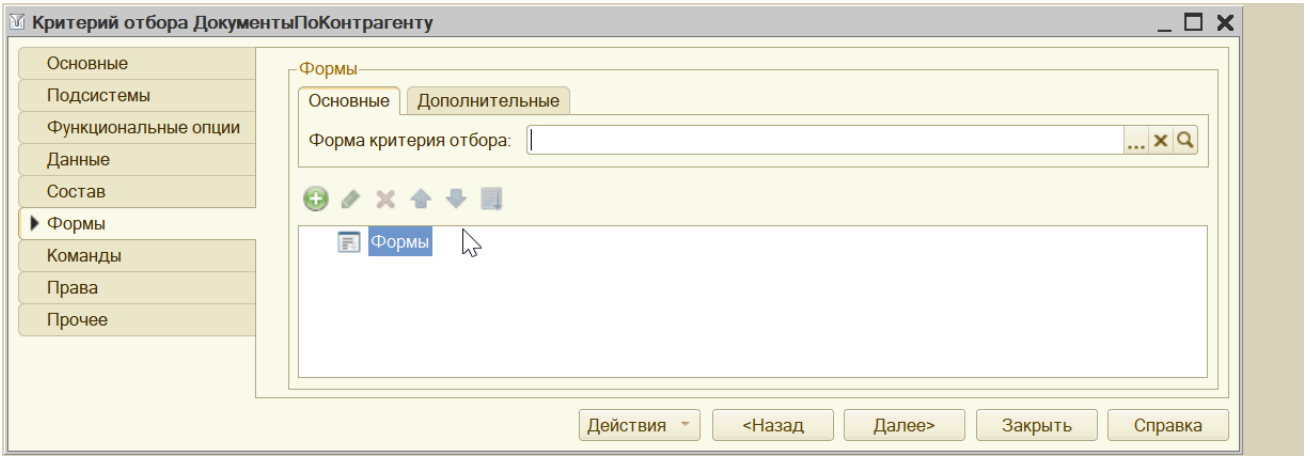


Рис.7 Закладка «Состав»

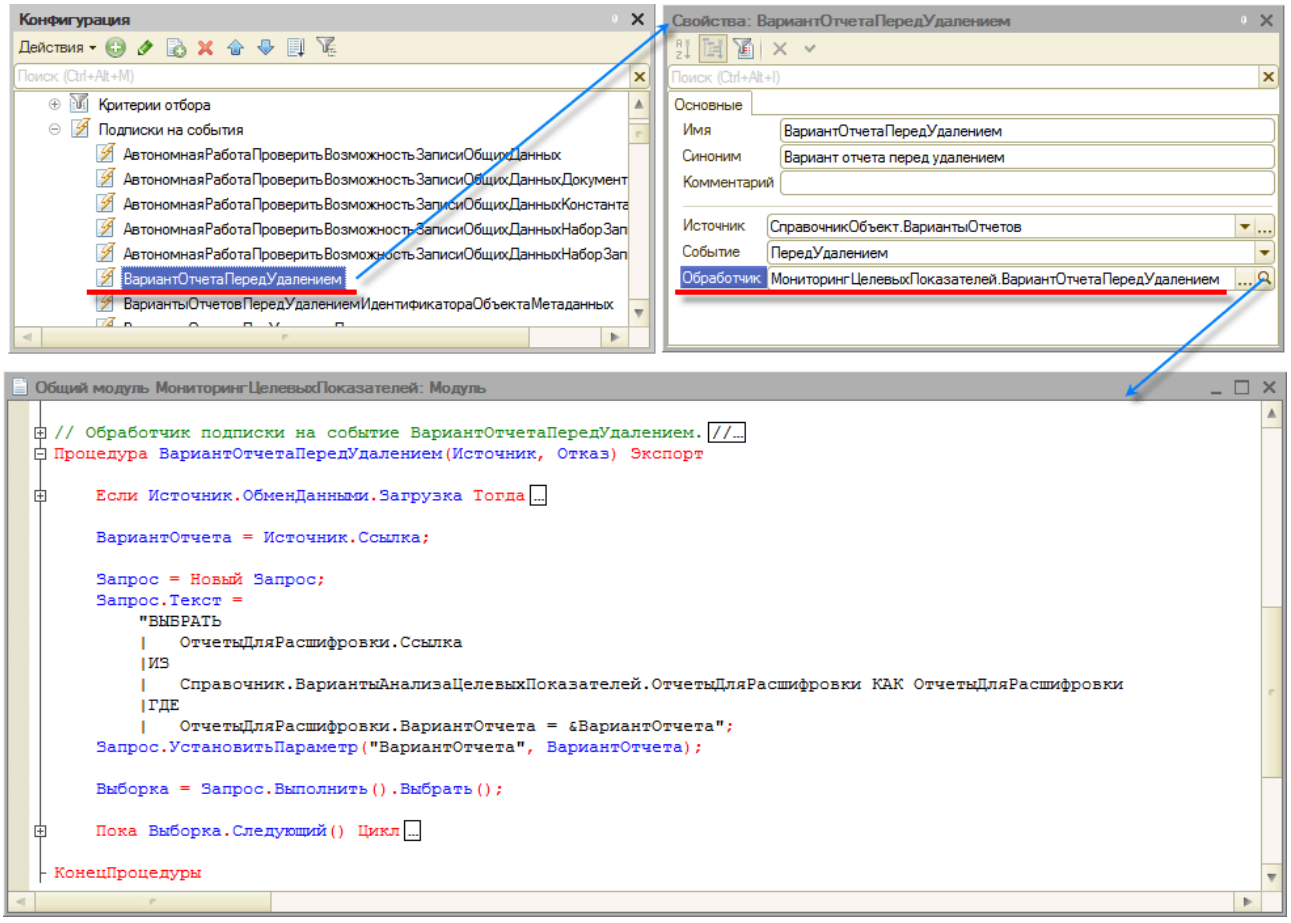
Критериев отбора можно создавать множество, но для работы с ними нужно создавать формы, которые можно размещать и на панели инструментов, и в пользовательском меню. Критерии будут доступны по кнопке «Еще» во всех отмеченных объектах.



Подписка на событие

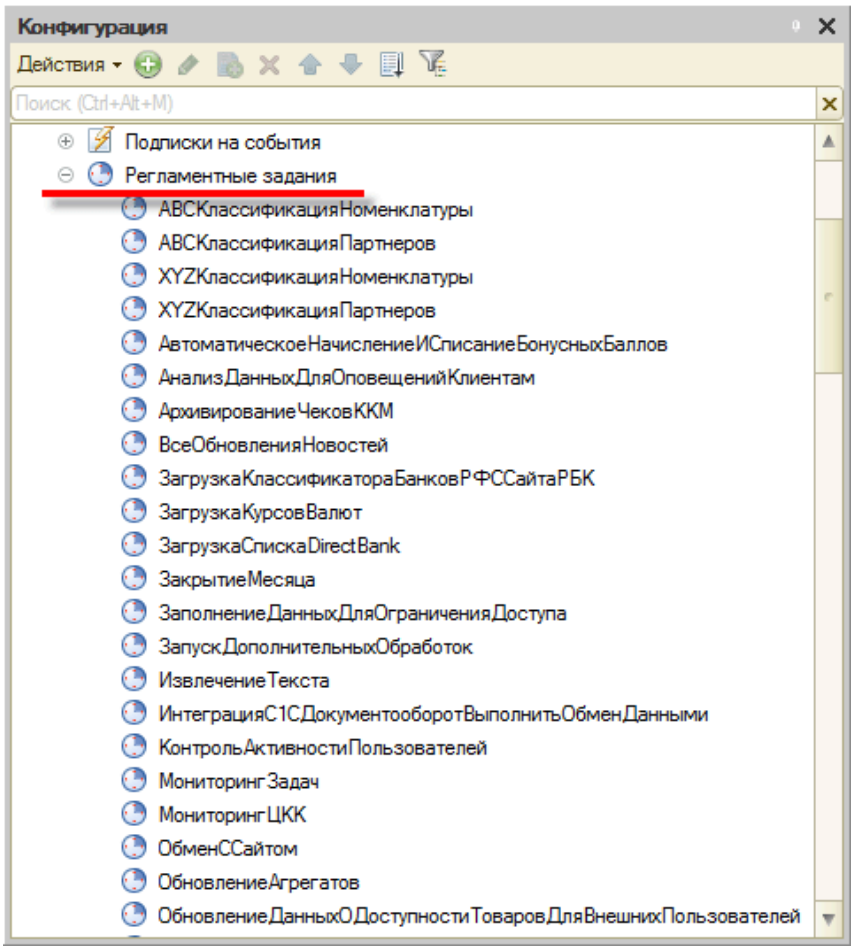
Подписки на события — это общие объекты конфигурации. Они позволяют назначать обработчики для неинтерактивных событий одного или нескольких прикладных объектов. В качестве обработчика может быть задана экспортируемая процедура общего модуля, если модуль соответствует определенному набору условий.

Ниже приведен пример подписки на событие **При записи** для всех бизнес-процессов, которые созданы в конфигурации:

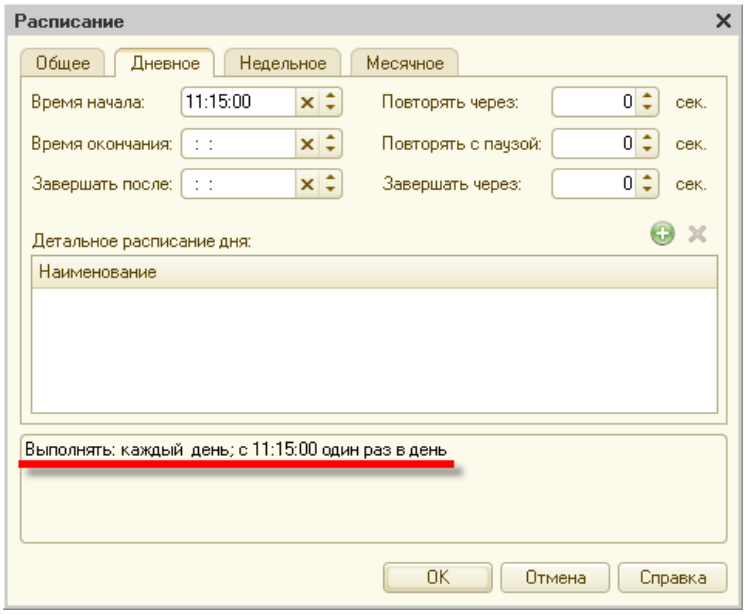


Регламентное задание

Регламентные задания — это общие объекты конфигурации. Они являются частью механизма заданий и позволяют автоматически выполнять процедуры на встроенном языке по расписанию.

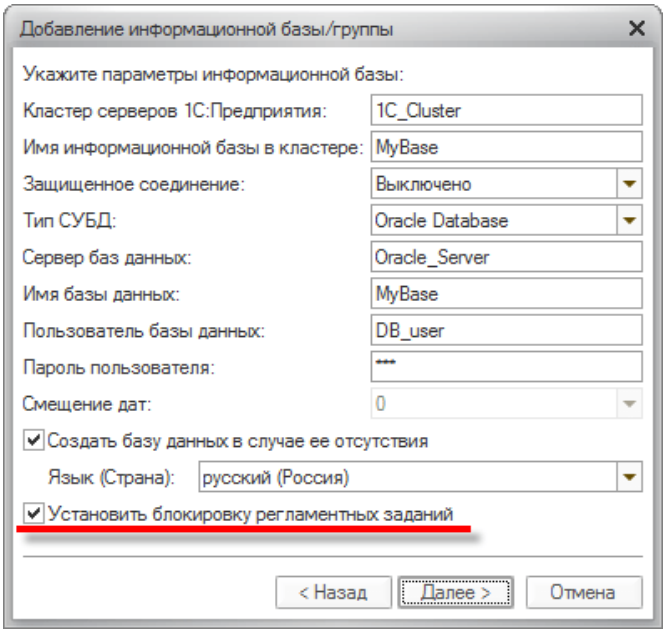


Для каждого регламентного задания может быть задано расписание, в соответствии с которым регламентное задание будет автоматически запущено на исполнение. В системе 1С:Предприятие 8 поддерживаются однократные и периодические расписания. Можно задать дату начала и окончания выполнения, дневное, недельное и месячные расписания. Расписание можно задать как на этапе конфигурирования, так и на этапе выполнения (в режиме 1С:Предприятие).



В процессе запуска регламентное задание порождает фоновое задание, которое и выполняет реальную обработку. Регламентное задание может выполняться от имени заданного пользователя и имеет возможность перезапуска (например, в случае непредвиденного завершения работы).

В утилите администрирования клиент-серверного варианта работы автоматическое выполнение регламентных заданий может быть запрещено для конкретной информационной базы. Также имеется возможность блокировать запуск регламентных заданий при создании информационной базы на сервере из диалога запуска 1С:Предприятия 8.

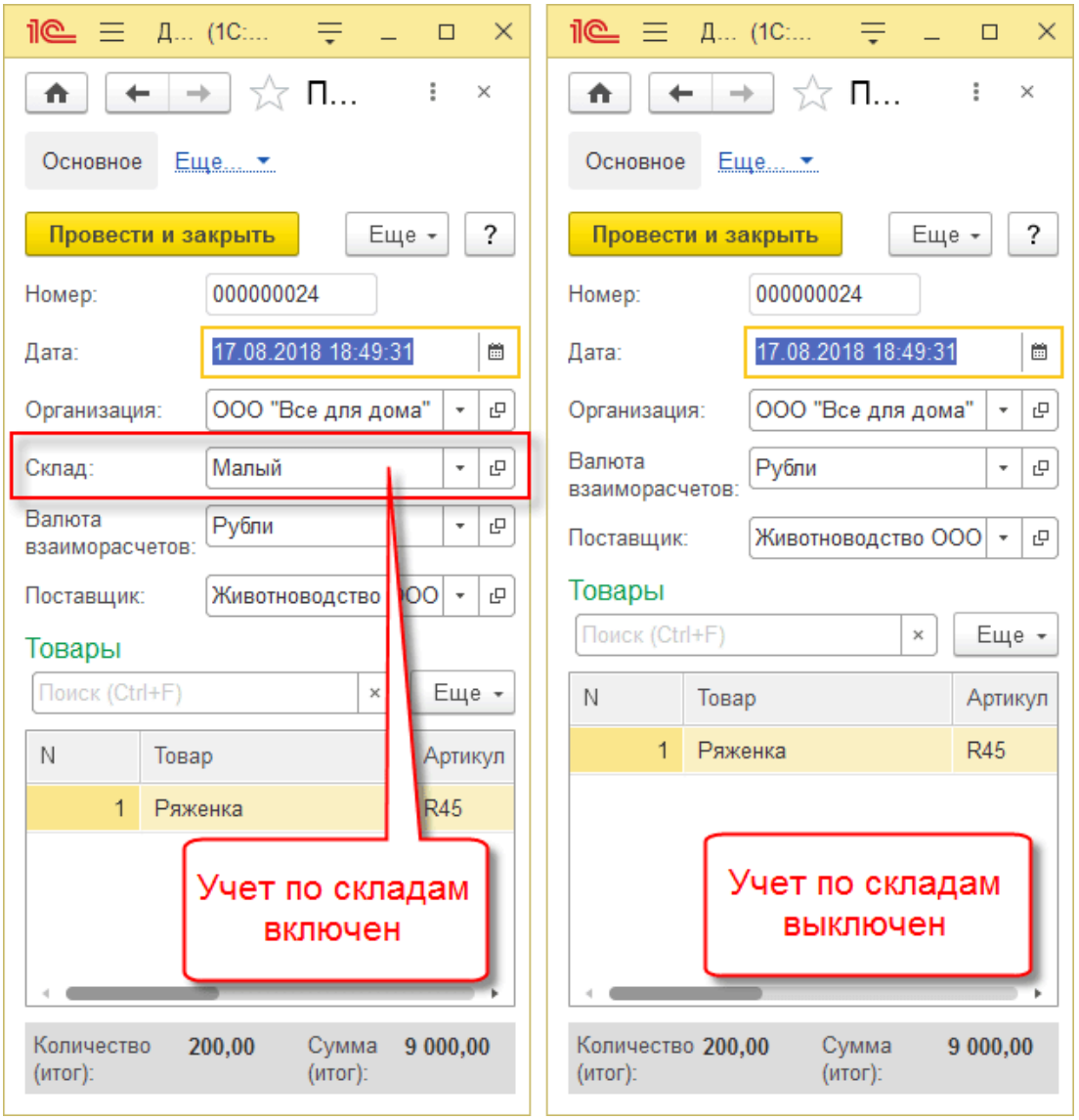


Функциональные опции

Механизм функциональных опций — это один из инструментов разработки. Он позволяет определить в конфигурации ту функциональность, которая может использоваться или не использоваться при внедрении в зависимости от потребностей конкретной организации.

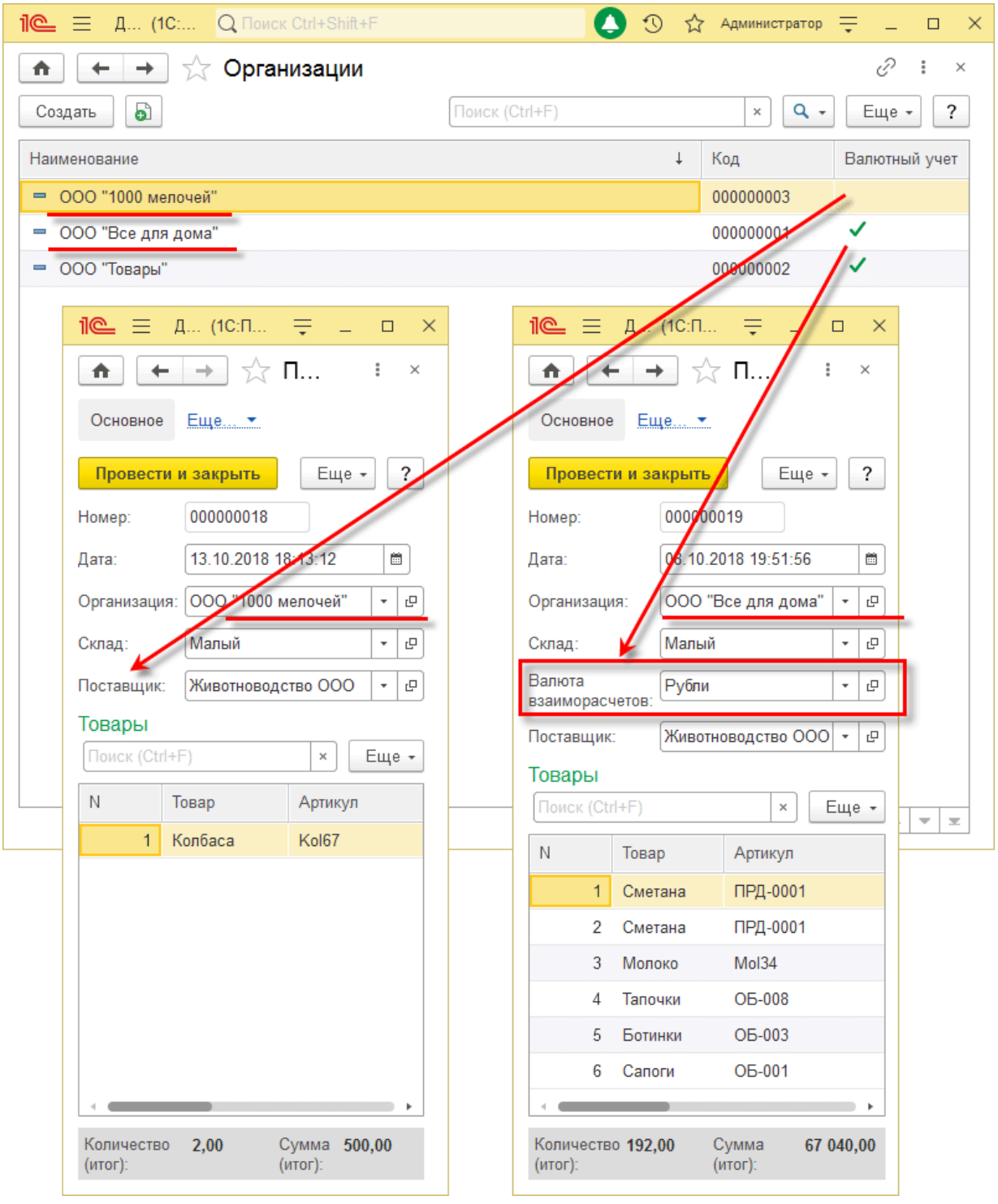
Работа механизма основана на двух объектах конфигурации:

- **Функциональная опция**
  С функциональными опциями, добавленными в прикладное решение, можно связать объекты конфигурации и их реквизиты. Например, с функциональной опцией Учет по складам можно связать реквизит Склад документа Поступление товара. Тогда, если в режиме 1С:Предприятие включить эту функциональную опцию, поле Склад будет отображаться во всех формах документа. Если выключить — поле Склад отображаться не будет. Подробнее…



- **Параметр функциональной опции**
  Функциональные опции могут использоваться с параметрами. Например, для того, чтобы вид конкретной формы мог зависеть от значения параметра, выбранного в форме. Например, параметром функциональной опции Валютный учет может

быть Организация. Тогда, в зависимости от того, какая организация выбрана в форме, поле Валюта взаиморасчетов будет скрыто или будет отображаться. Подробнее…

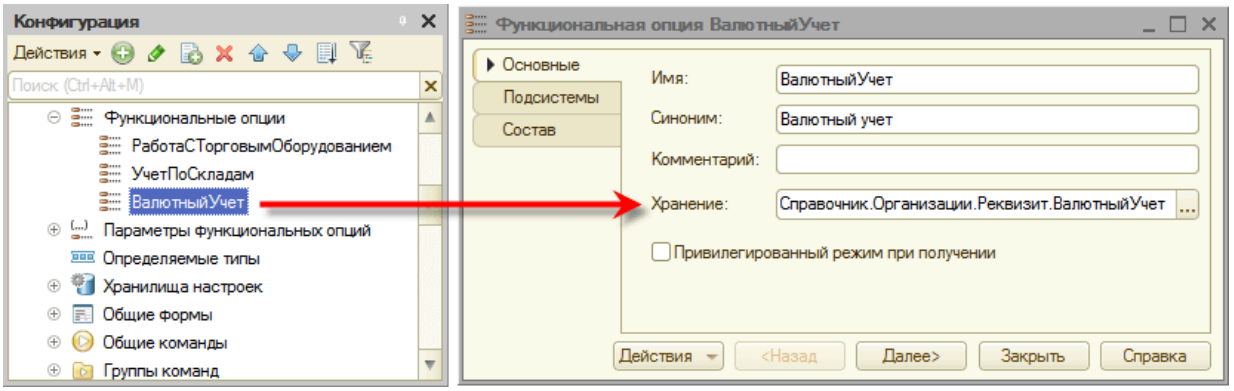


Параметр функциональной опции

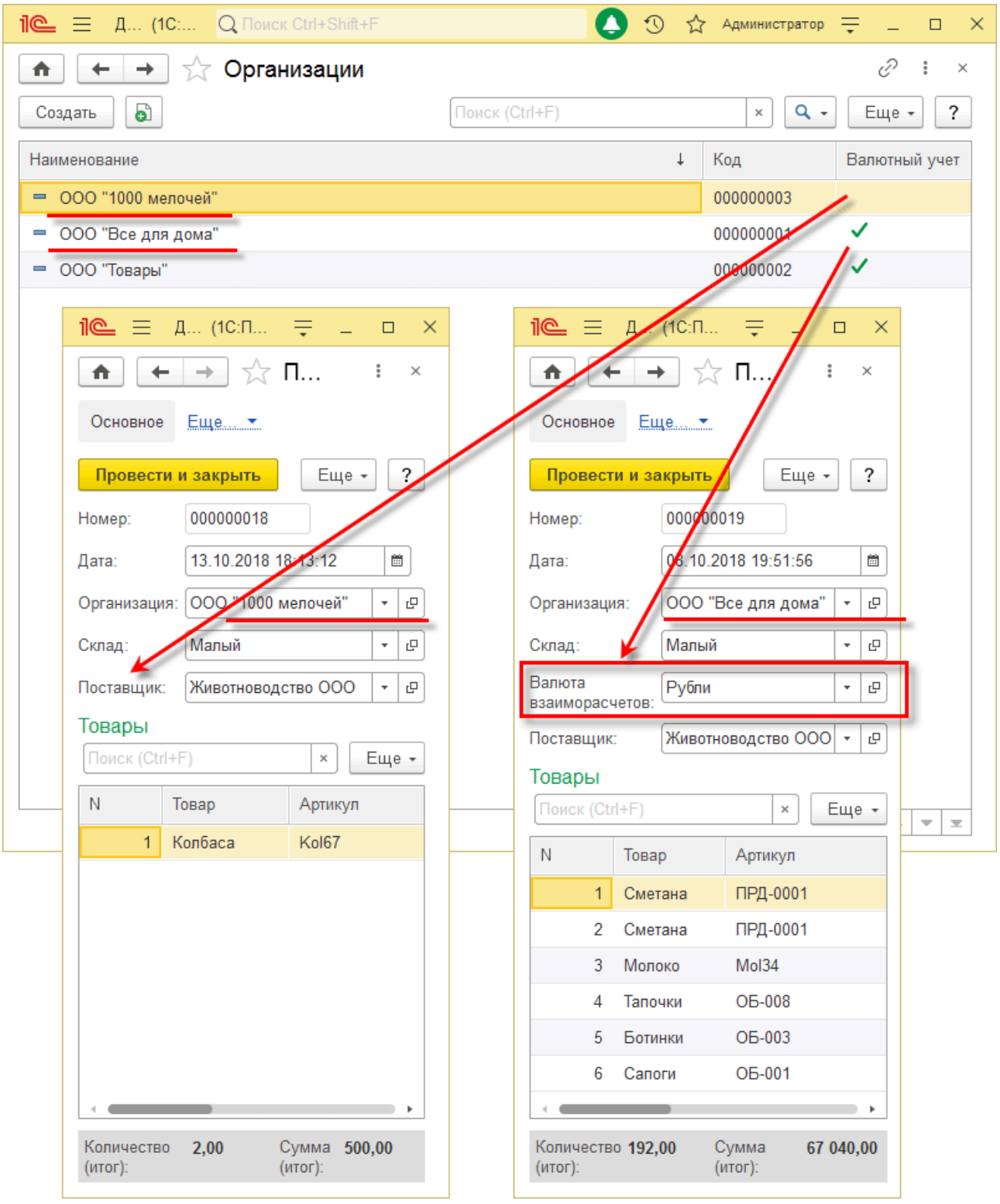
Параметры функциональных опций — это общие объекты конфигурации. Они являются частью механизма функциональных опций и позволяют создавать функциональные опции, результат работы которых задан не жестко (один раз при внедрении), а может изменяться в зависимости от данных, хранящихся в информационной базе (в зависимости от значения параметра функциональной опции).

Например, в процессе работы требуется показывать или скрывать поле **Валюта взаиморасчетов** в документах в зависимости от того, для какой организации оформляется поступление товаров. Если в этой организации ведется валютный учет — поле **Валюта взаиморасчетов** должно быть показано. Если валютный учет не ведется — это поле должно быть скрыто.

Для решения такой задачи в конфигурацию можно добавить функциональную опцию **Валютный учет** и параметр функциональной опции **Организация**. Для функциональной опции указать, что ее значение будет храниться в булевом реквизите организации (ее параметра) — **Валютный учет.**



Тогда, если организация не ведет валютный учет — поле **Валюта** взаиморасчетов будет скрыто. Если ведет — поле будет показано.


Организации
Создать
Наименование
Код
Валютный учет
ООО "1000 мелочей"
000000003
ООО "Все для дома"
000000001
ООО "Товары"
000000002
Основное
Еще...
Провести и закрыть
Еще
Номер:
000000018
Дата:
13.10.2018 18:13:12
Организация:
ООО "1000 мелочей"
Склад:
Малый
Поставщик:
Животноводство ООО
Товары
Поиск (Ctrl+F)
N
Товар
Артикул
1
Колбаса
Kol67
Количество (итог):
2,00
Сумма (итог):
500,00
Номер:
000000019
Дата:
06.10.2018 19:51:56
Организация:
ООО "Все для дома"
Склад:
Малый
Валюта взаиморасчетов:
Рубли
Поставщик:
Животноводство ООО
1 Сметана ПРД-0001
2 Сметана ПРД-0001
3 Молоко Mol34
4 Тапочки ОБ-008
5 Ботинки ОБ-003
6 Сапоги ОБ-001
Количество (итог):
192,00
Сумма (итог):
67 040,00

Определяемые типы в 1С 8.3

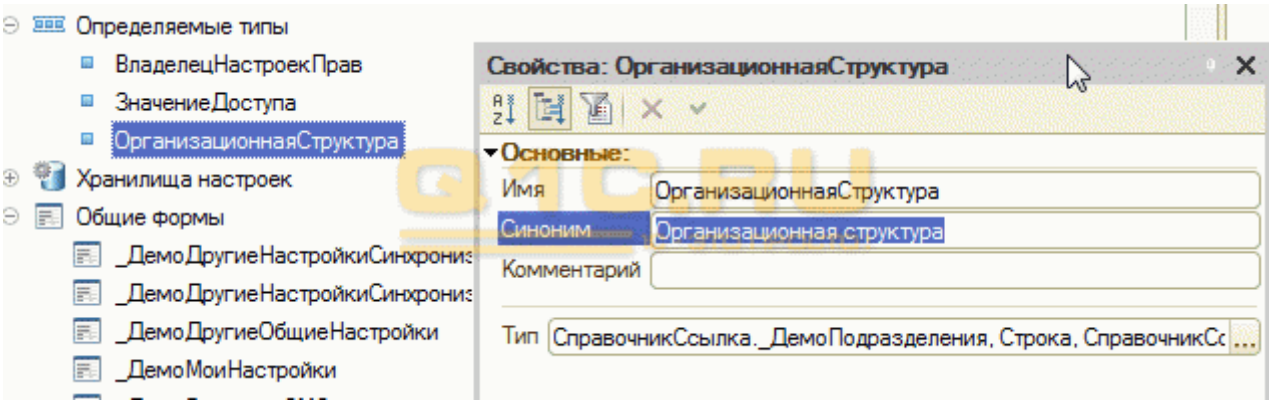
Определяемые типы —объект метаданных 1С, который появился в платформе версии 8.3. Объект имеет вспомогательный характер.

Что такое определяемые типы

Определяемые — это новые типы данных, набор которых определяет разработчик прикладного решения. Он состоит из нескольких допустимых типов. Такой тип данных может являться типом любого реквизита конфигурации, кроме самого определяемого типа и объекта «План видов характеристики«.

Настройка

Настроек у данного объекта не так уж и много:



Т.е. помимо имени и синонима в 1С определяемые типы имеют только набор типов объектов.

Примеры использования:

- можно описать в конфигурации реквизиты, ссылающиеся на данные, которые будут определены при встраивании фрагмента в конкретную конфигурацию;
- присвоить некоторому набору подписок на событие один тип определяемых данных, при изменении набора типов для всех подписок будет меняться источник динамически;
- использовать как некую альтернативу плана видов характеристик

и многое другое.

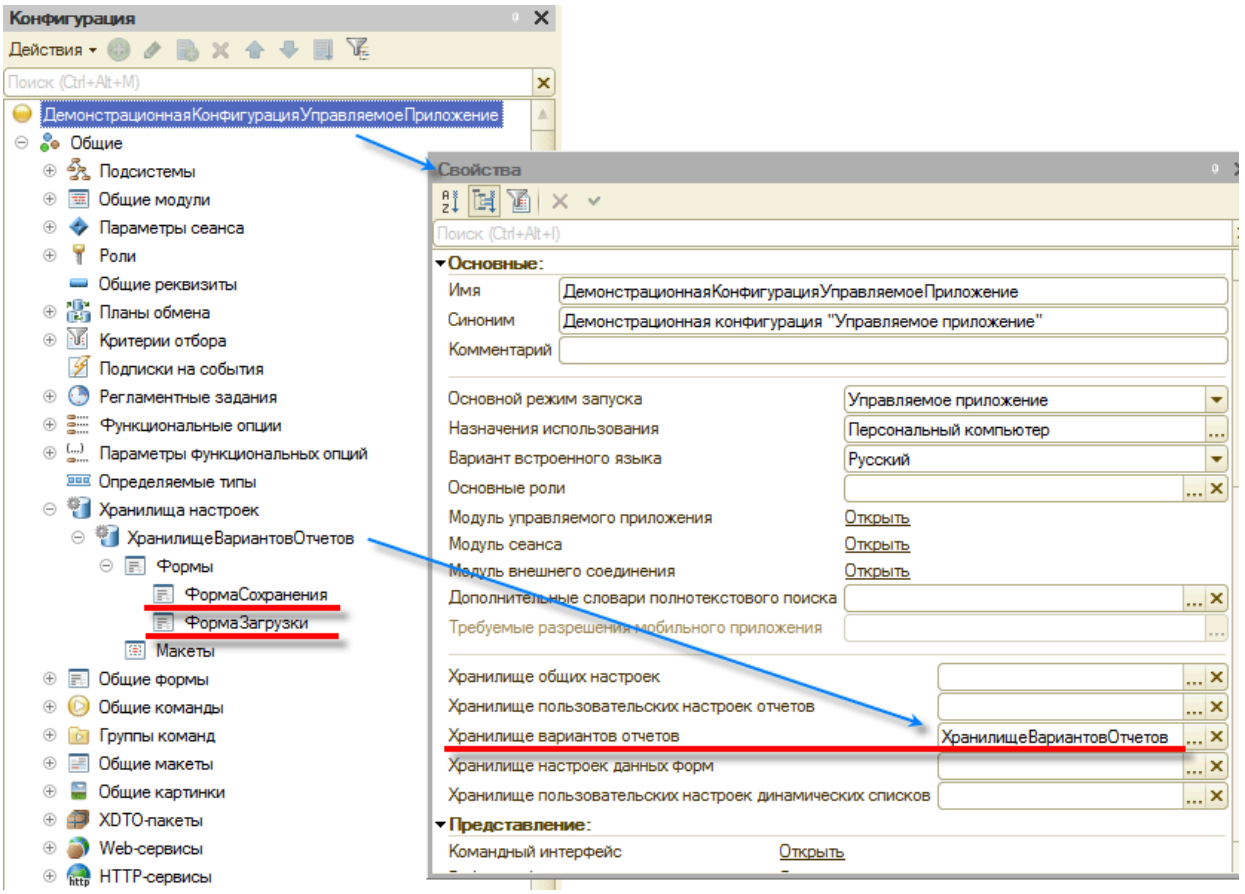
Хранилище настроек

Хранилища настроек — это общие объекты конфигурации. Они позволяют разработчику создавать собственные механизмы хранения настроек, в тех случаях, когда возможности стандартного механизма хранения настроек недостаточны.

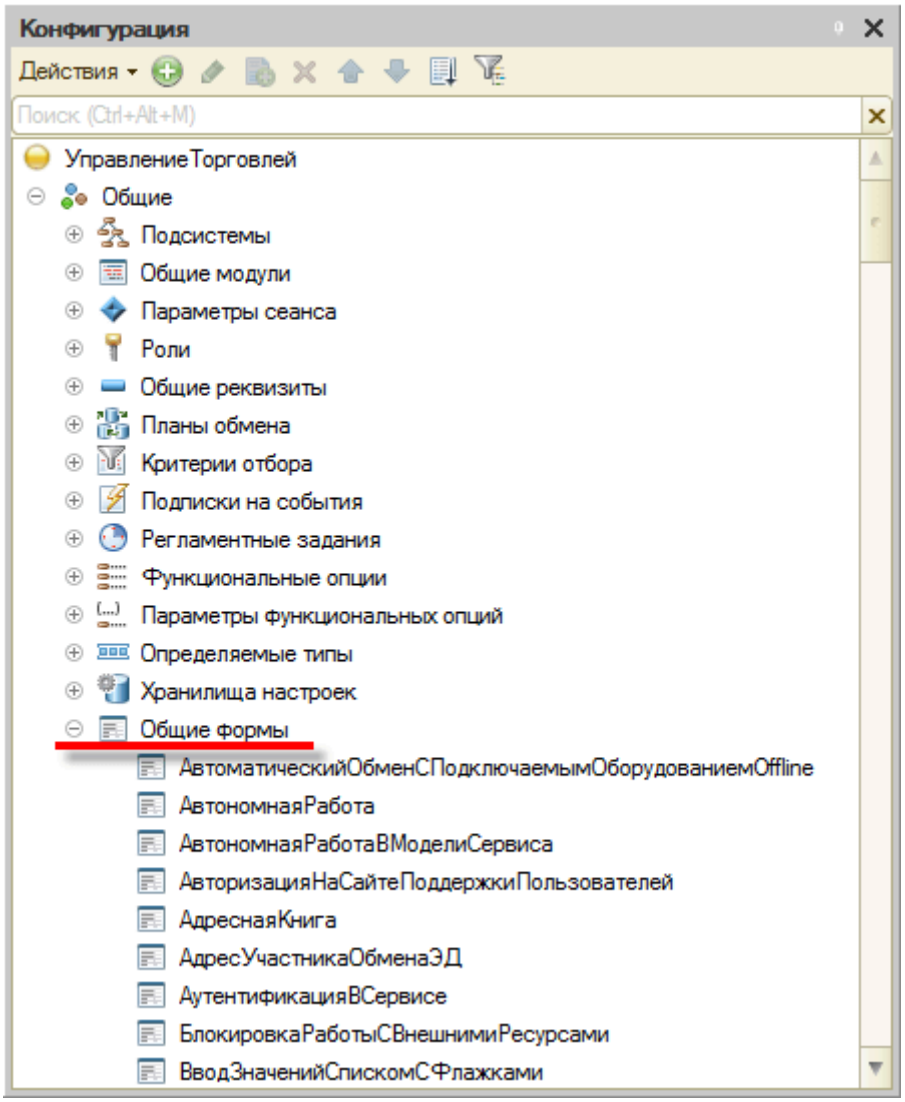
В процессе работы пользователь выполняет различные настройки в системе: создает варианты отчетов, сохраняет настройки отчетов, вводит какие-то данные в формы и т. д. Кроме этого в конкретном прикладном решении могут существовать собственные настройки, определяемые логикой работы этого решения, которые делают работу пользователя более удобной. Эти настройки сохраняются платформой в системных таблицах информационной базы.

Однако не во всех случаях системные механизмы сохранения настроек могут на 100% удовлетворить потребности разработчика. Хранилища настроек как раз позволяют разработчику хранить пользовательские настройки не в системных таблицах, а в объектах конфигурации (например, в специальных справочниках, в регистрах сведений). Благодаря этому разработчик может реализовать собственный формат хранения этих настроек и собственные алгоритмы работы с этими настройками. Например, когда требуется особенная структура хранения настроек, когда необходимо обеспечить передачу настроек в пределах распределенной информационной базы, когда существуют особенные требования к администрированию настроек и т. д.

В свойствах конфигурации, отчетов, обработок или форм можно переопределить сохранение настроек в хранилище, добавленное разработчиком. А в формах загрузки и сохранения настроек описать алгоритмы, в соответствии с которыми настройки будут сохраняться/восстанавливаться.



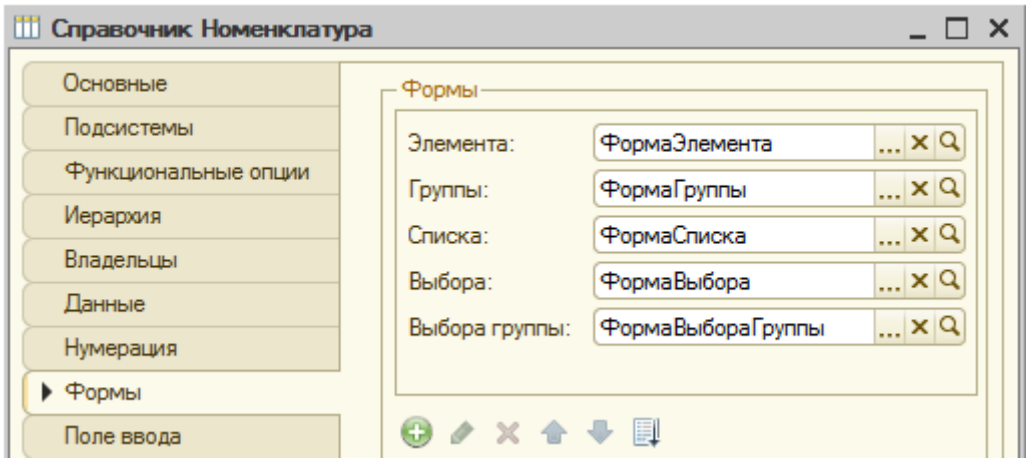
формы, не принадлежащие конкретным объектам конфигурации — **общие формы.**



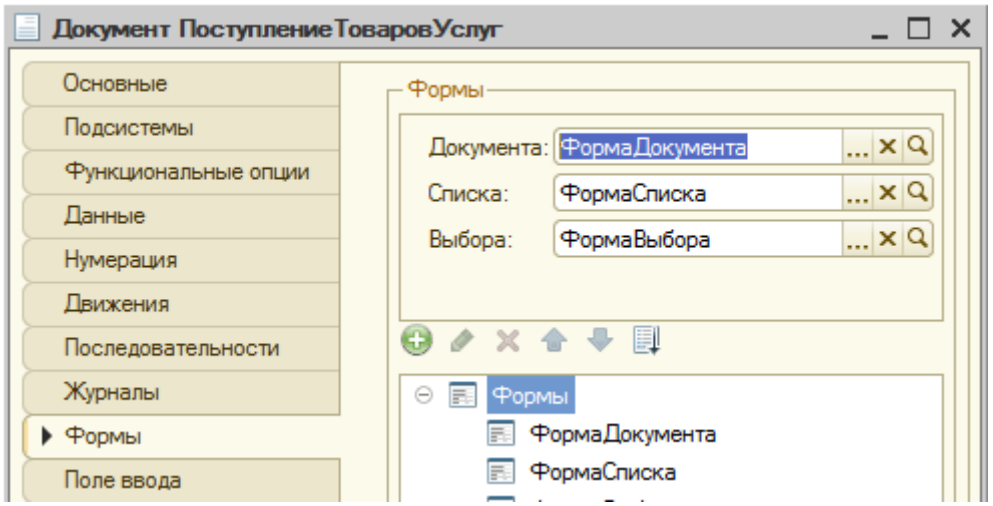
Основные формы

Каждый объект конфигурации может использоваться для выполнения некоторых стандартных действий. Например, для любого справочника может потребоваться отображать список его элементов, отображать отдельные элементы справочника, отображать группу справочника, выбирать элементы и группы элементов из справочника. Для любого документа список таких действий будет гораздо меньше: просмотр списка документов, выбор из списка документов и просмотр отдельного документа.

Чтобы обеспечить выполнение таких стандартных действий с данными объектов прикладного решения, для каждого из них существует набор основных форм, которые будут использоваться при выполнении соответствующих действий. Основной может быть назначена любая из форм, подчиненных этому объекту. Например, у справочника Номенклатура могут существовать следующие основные формы:



А у документа Поступление товаров и услуг состав основных форм будет уже другим:



Таким образом, если пользователь захочет посмотреть список справочника Номенклатура или список документов Поступление товаров и услуг, система откроет соответствующую форму, назначенную в качестве формы списка для этих объектов.

Автогенерируемые формы

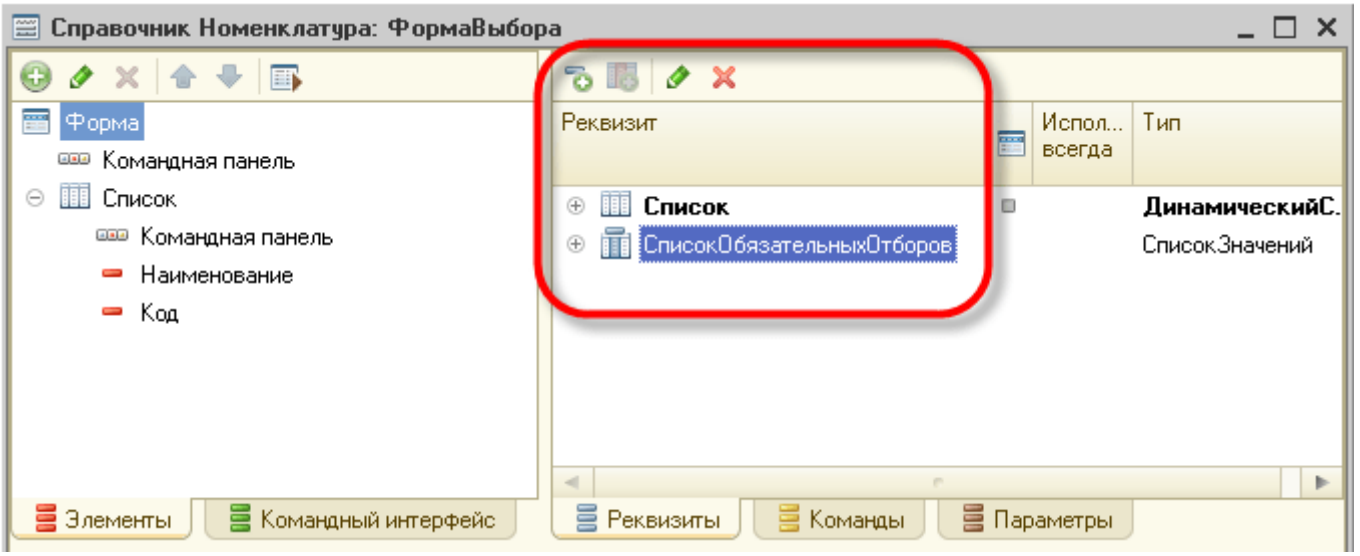
Разработчику достаточно добавить новый объект конфигурации, а система сама сгенерирует в нужные моменты работы пользователя необходимые формы для отображения информации, содержащейся в этом объекте.

Таким образом, разработчику нужно создавать собственные формы объектов прикладного решения лишь в том случае, если они должны иметь отличия (другой дизайн или специфическое поведение) от форм, автоматически генерируемых системой.

Связь формы с данными

Принадлежность формы тому или иному объекту конфигурации не определяет состав данных, которые отображаются в форме. То, что форма принадлежит, например, справочнику Номенклатура, позволяет назначить ее одной из основных форм для этого справочника, но никак не определяет, какие же именно данные будет отображать эта форма, и каково будет ее поведение.

Для того чтобы связать форму с данными, используются реквизиты формы, в которых указывается перечень данных, отображаемых формой. Все формы, сами по себе, имеют одинаковое поведение, независимо от того, какие данные они отображают. Однако один из реквизитов формы может быть назначен для нее основным (он выделяется жирным шрифтом), и в этом случае стандартное поведение формы и ее свойства будут дополнены в зависимости от того, какой тип имеет основной реквизит формы:

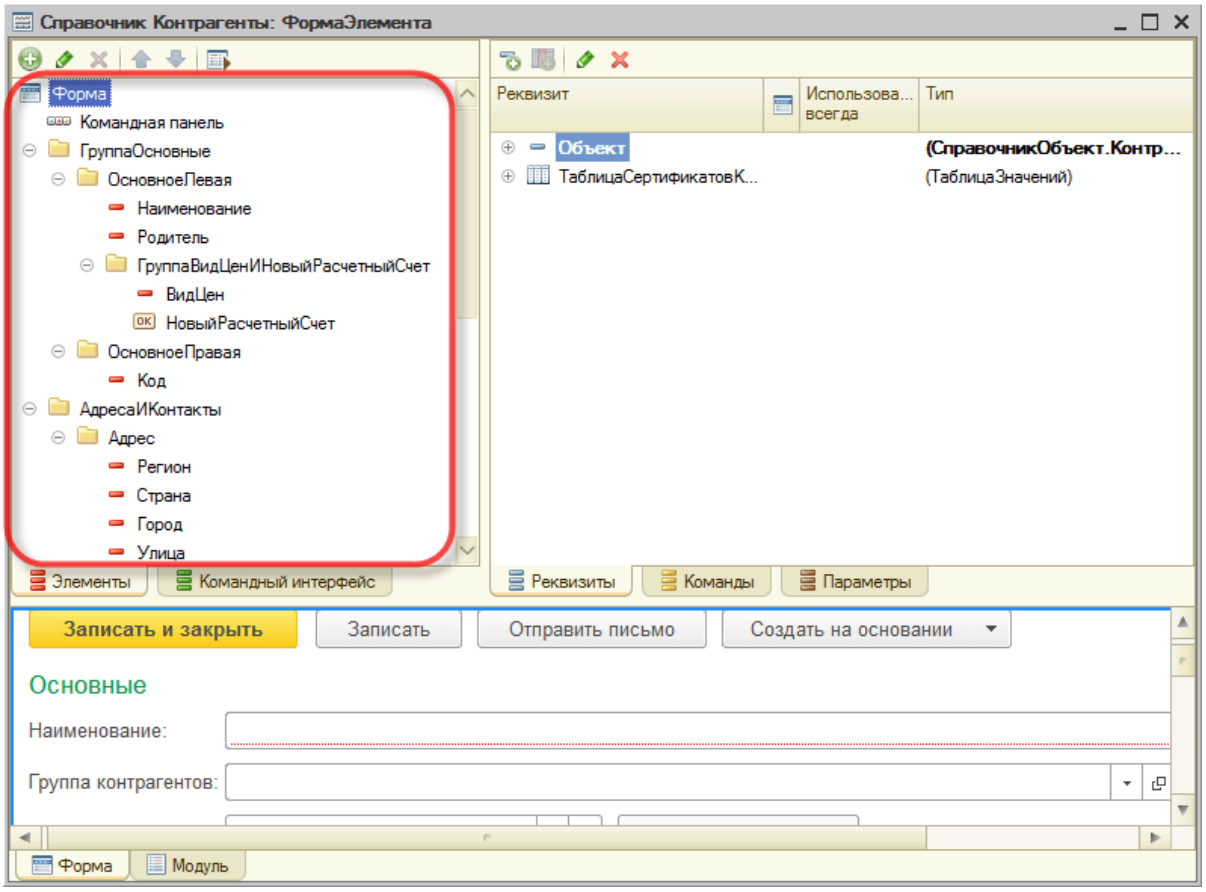


Например, если в качестве основного реквизита формы будет назначен документ Поступление товаров и услуг, то при закрытии формы система будет запрашивать подтверждение записи и проведения этого документа. Если же основным реквизитом формы назначить, скажем, справочник Номенклатура, то подобного запроса подтверждения при закрытии формы возникать не будет.

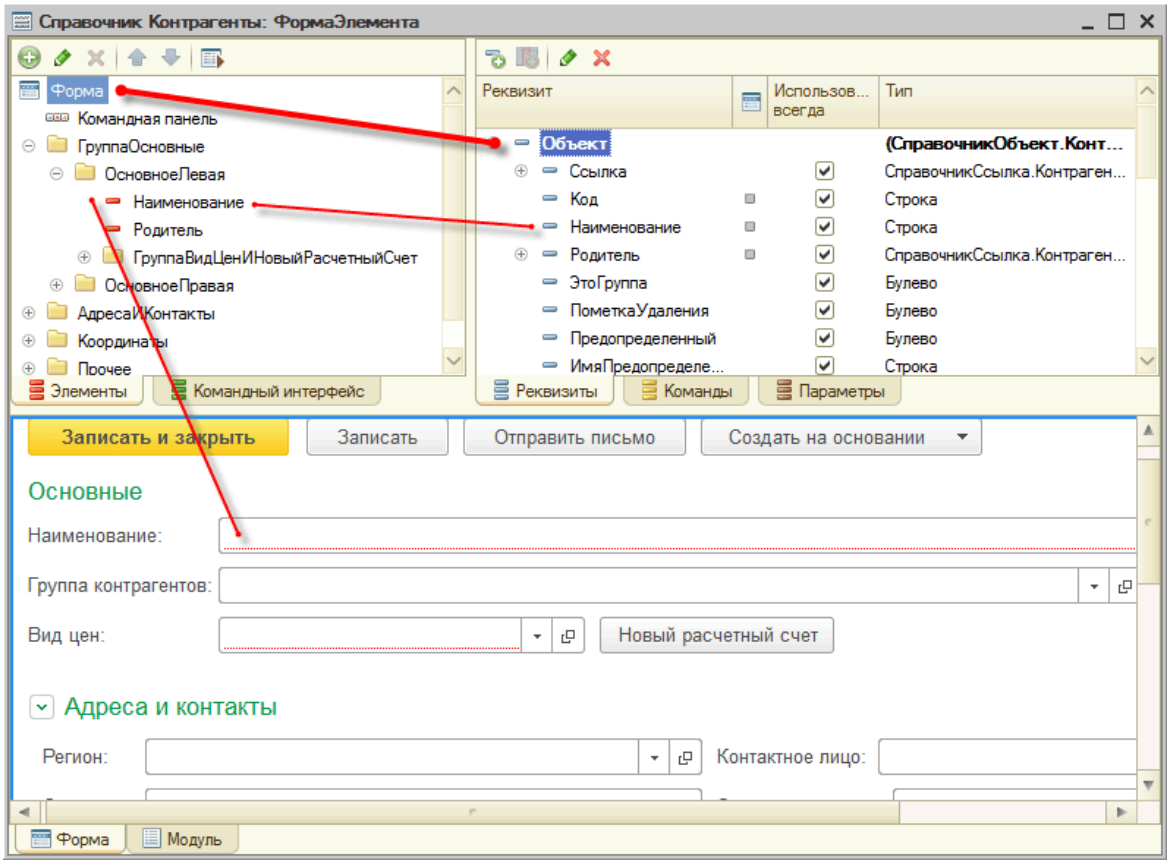
Структура формы

Основная особенность форм заключается в том, что они не нарисованы разработчиком детально, «по пикселям». Форма в конфигурации представляет собой логическое описание состава формы. А конкретное размещение элементов выполняется системой автоматически при отображении формы.

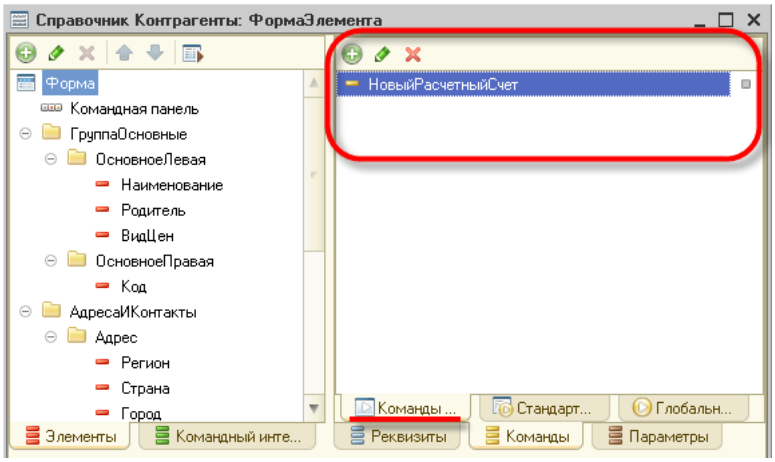
Отображаемая часть формы (видимая пользователю) описывается как дерево, включающее элементы формы.



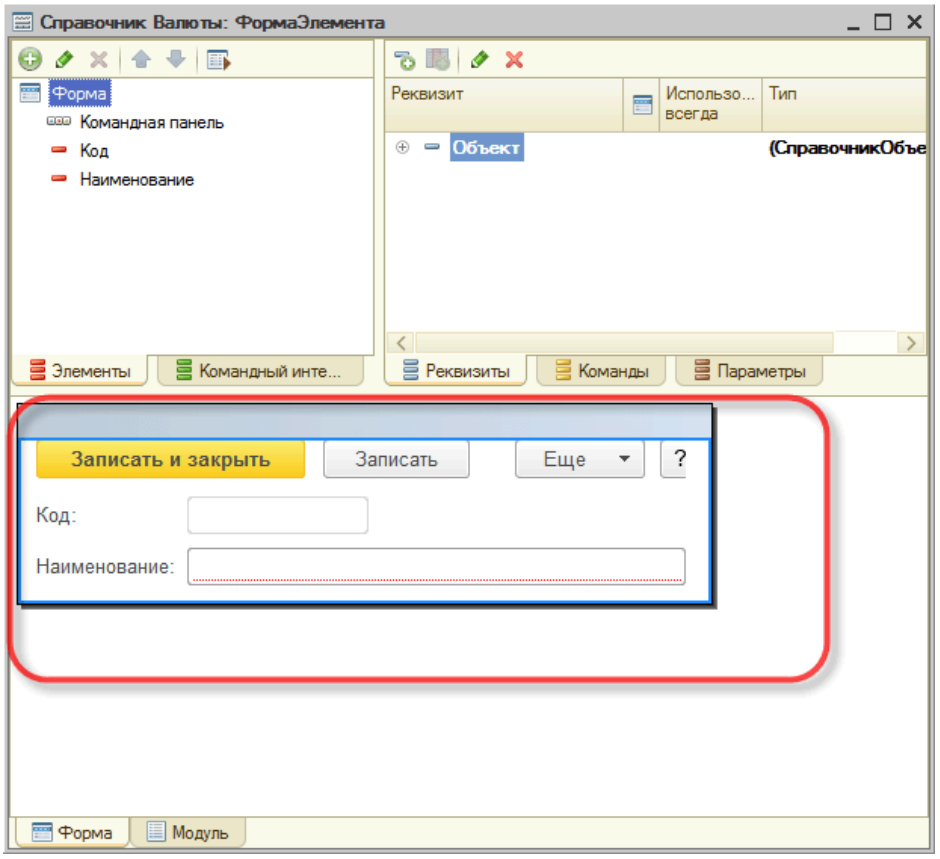
Элементы могут представлять собой поля ввода, флажки, переключатели, кнопки и т. д. Кроме того, элемент может быть группой, включающей другие элементы. Группа может представляться как панель с рамкой, панель со страницами (закладками), собственно страница, командная панель. Помимо этого элемент может представлять собой таблицу, которая тоже включает элементы (колонки). Структура элементов описывает то, как будет выглядеть форма.



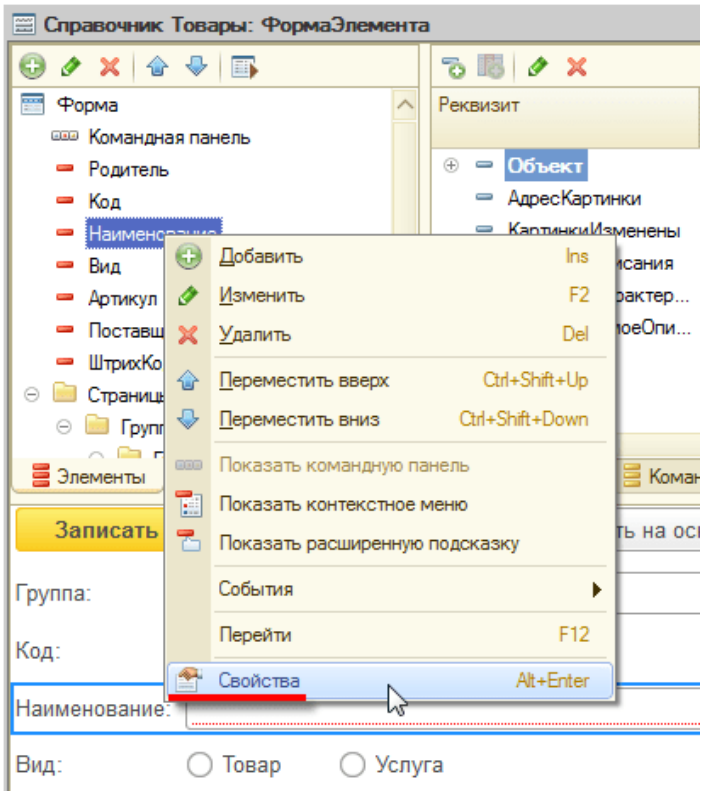
Вся функциональность формы описывается в виде реквизитов и команд. Реквизиты — это данные, с которыми работает форма, а команды — выполняемые действия. Таким образом, разработчик в редакторе формы должен включить в форму необходимые реквизиты и команды, создать отображающие их элементы формы и, если необходимо, скомпоновать элементы в группы.



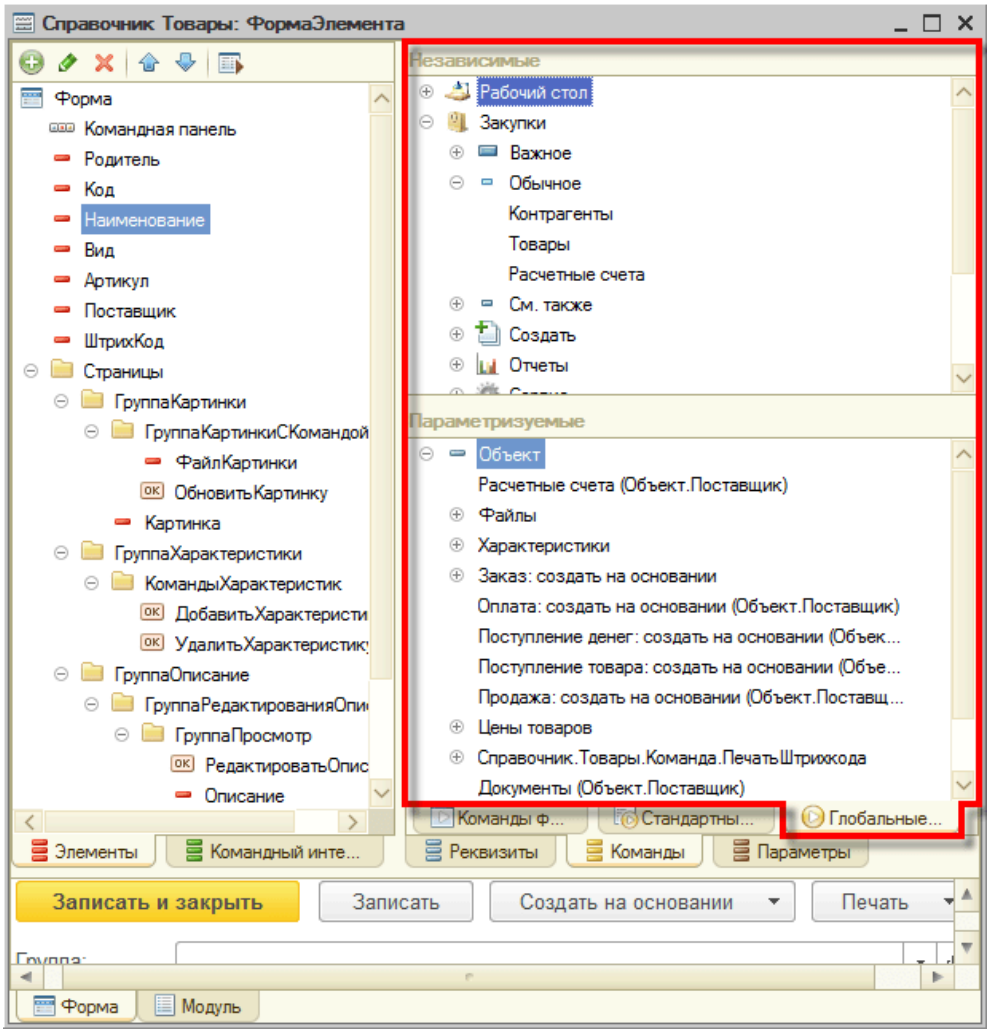
На основе этого логического описания система автоматически формирует внешний вид формы для отображения пользователю. При этом системой учитываются различные свойства отображаемых данных (например, тип), чтобы максимально удобно для пользователя расположить элементы формы.



Разработчик может влиять на расположение элементов различными установками. Он может определять порядок элементов, указывать желаемую ширину и высоту. Однако это является только некоторой дополнительной информацией, помогающей системе отобразить форму.



В формах разработчик может использовать не только команды самой формы, но и глобальные команды, используемые в командном интерфейсе всей конфигурации. Кроме того, реализована возможность создания параметризуемых команд, которые будут открывать другие формы с учетом конкретных данных текущей формы. Например, это может быть вызов отчета по остаткам на том складе, который выбран сейчас в форме расходной накладной.

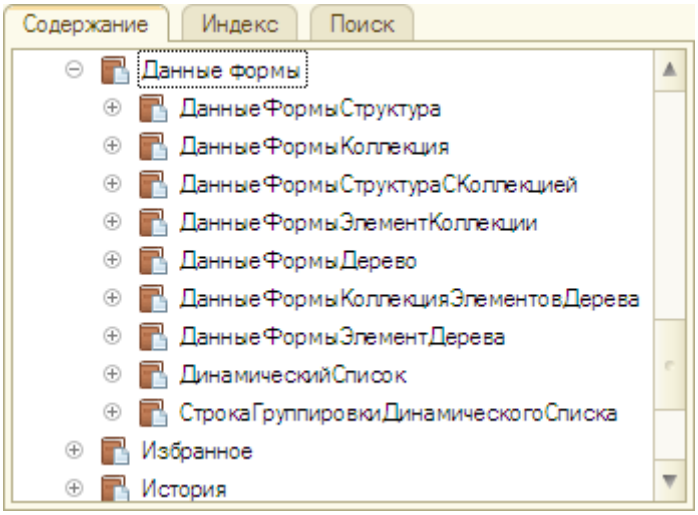


Механика работы форм

Функционирование управляемых форм имеет следующие отличительные особенности:

- Форма существует и на клиенте и на сервере
  Она осуществляет клиент-серверное взаимодействие (передачу данных и оформительских свойств элементов)
- Форма не работает с прикладными объектами
  В форме используются специальные универсальные

объекты ДанныеФормы… Прикладные объекты работают только на сервере и только во время выполнения некоторых операций

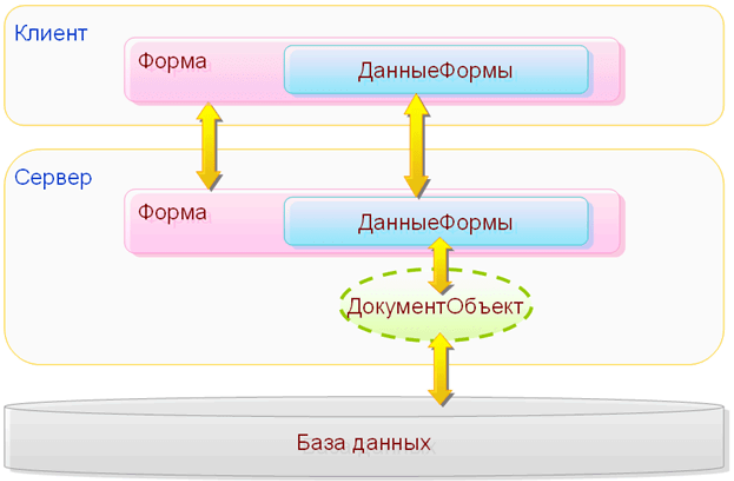


При открытии формы:

- Объект считывается из базы данных
- Объект конвертируется в данные формы
- Объект удаляется из памяти
- Данные формы передаются на клиента

При записи данных из формы:

- Данные формы получаются с клиента
- Данные формы конвертируются в объект
- Объект записывается в базу данных
- Объект удаляется из памяти

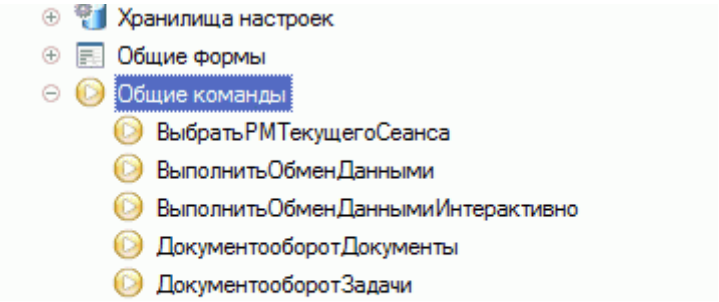


Редактор форм

Для создания и изменения форм разработчик может использовать редактор формы, обладающий удобными средствами графического конструирования форм различной сложности. Подробнее...

Общие команды и группы команд в 1С 8.3

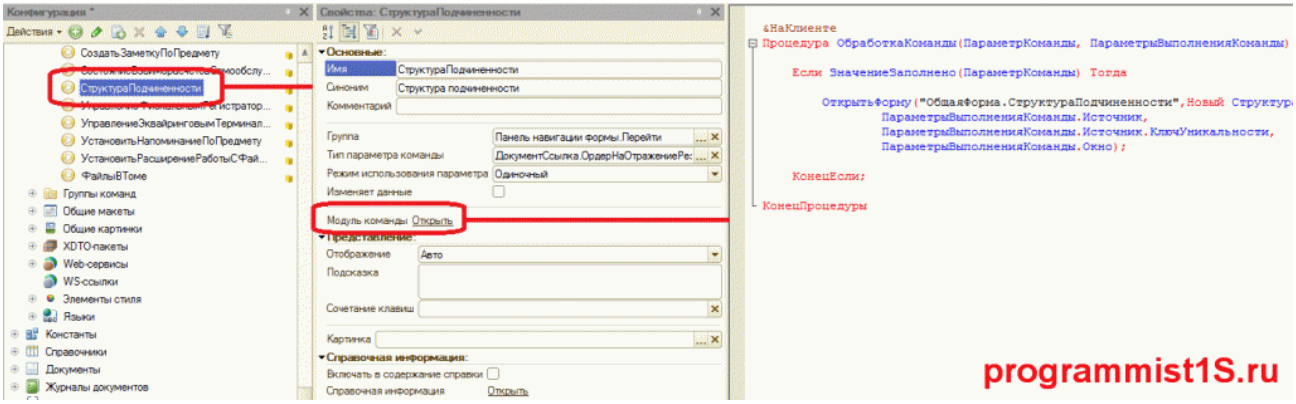
Общие команды — механизм платформы, предназначенный для описания часто используемых команд в конфигурации 1С 8.3.



Общие или глобальные команды удобно использовать, если одна команда нужна многим объектам конфигурации. Например, кнопка для вывода структуры подчиненности документов, команда отображения проводок документа, отчет об изменении объекта.
В команду можно передать параметры, например, из формы какого объекта он вызывается.

Настройка и свойства общей команды в 1С

Добавить новую команду и настроить общую форму



- **Группа** — расположение будущей команды на интерфейсе.
- **Тип параметра команды** — определяет набор объектов, в котором будет отображаться будущая команда.
- **Режим использования параметра** — задает возможность передать в качестве параметра команды одно или несколько значений.
- **Изменяет данные** — если галка установлена, то при выполнение команды форма будет пересчитана с сервера.
- **Модуль команды** — обработчик выполнения команды, выполняется на клиенте.

Пример модуля команды:

```
&НаКлиенте
Процедура ОбработкаКоманды(ПараметрКоманды, ПараметрыВыполненияКоманды)

Если ЗначениеЗаполнено(ПараметрКоманды) Тогда

ОткрытьФорму("ОбщаяФорма.СтруктураПодчиненности",Новый
Структура("ОбъектОтбора", ПараметрКоманды),
```

```
ПараметрыВыполненияКоманды.Источник,
ПараметрыВыполненияКоманды.Источник.КлючУникальности,
ПараметрыВыполненияКоманды.Окно);

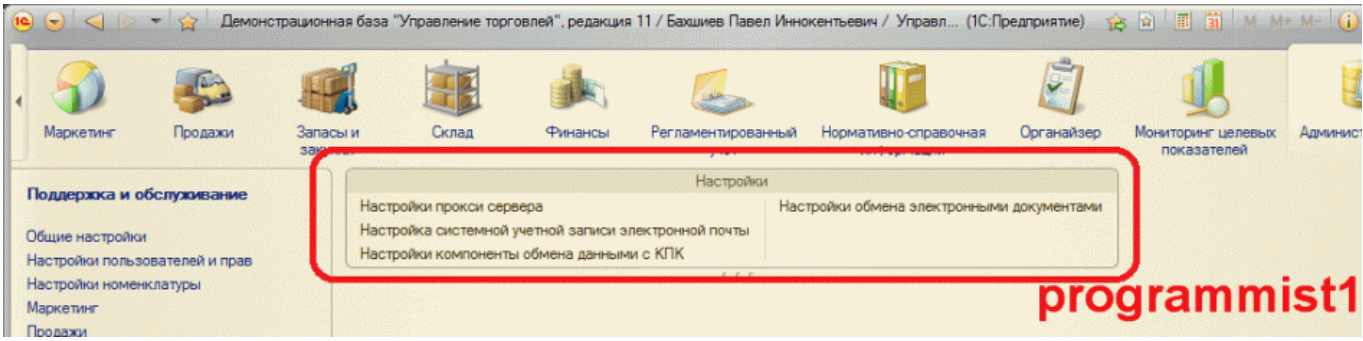
КонецЕсли;

КонецПроцедуры
```

Здесь ПараметрКоманды — объект, вызывающий команду. А в ПараметрыВыполненияКоманды структура, в которой описан Источник (вызываемая Форма), Окно (ОкноКлиентскогоПриложения), Уникальность, указывает, искать ли уже открытую форму или нет.

Группы команд 1С

Объект, позволяющий вывести команды на командный интерфейс конфигурации в отдельную группу:



Объект 1С "Группы команд" используется для логического объединения команд, выполняющих похожие действия, и позволяет:

- создать произвольную **группу команд**;
- вывести **команды группы** на командный интерфейс конфигурации (свойство Категория).

Объекты 1С "Общие макеты"

Общий макет 1С - объект конфигурации 1С, предназначенный для формирования печатных форм электронных документов, и доступный из любого места конфигурации (разных программных модулей).

Общие макеты печатных форм располагаются в ветви "Общие" дерева конфигурации, в отличие от печатных форм объектов конфигурации (справочников, документов, журналов документов, планов счетов, планов видов характеристик, планов видов расчетов, регистров, отчетов и обработок и др.), которые располагаются в подчиненных объектах "Макеты", а также во внешних файлах (в этом случае должно быть установлено свойство табличного документа Макет).

Рисунок "Общие макеты 1С"

Таким образом, "Общие макеты" (шаблоны печатных форм) не принадлежат конкретному объекту конфигурации, и могут используются из разных программных модулей.

Типы макетов 1С:

- Табличный документ – использование стандартной технологии создания и использования макетов: подготовка макета производится с помощью табличного редактора.
- Текстовый документ – использование текстового документа в качестве макета: подготовка текстового макета производится с помощью редактора текстовых макетов.
- Двоичные данные – используются двоичные данные.
- ActiveDocument – использование технологии OLE Active document (документы Microsoft Office OpenXML не могут быть использованы в качестве макета ActiveDocument).
- HTML-документ – использование редактора HTML-документа.
- Географическая схема – использование в качестве макета географической схемы, подготовленной в редакторе географических схем.
- Графическая схема – использование подготовленной в редакторе графической схемы.
- Схема компоновки данных – использование схемы компоновки данных, подготовленной в конструкторе.
- Макет оформления компоновки данных – предполагает использование макета оформления системы компоновки данных.
- Внешняя компонента – хранение внешних компонент, используемых в мобильном или обычном прикладном решении.

Замечание по Внешней компоненте

Внешняя компонента - это механизм интеграции, сторонняя программа, которая используется для решения задач, недоступных встроенному языку 1С:Предприятия (например, для настройки низкоуровневого взаимодействия с операционной системой или подключаемым оборудованием).

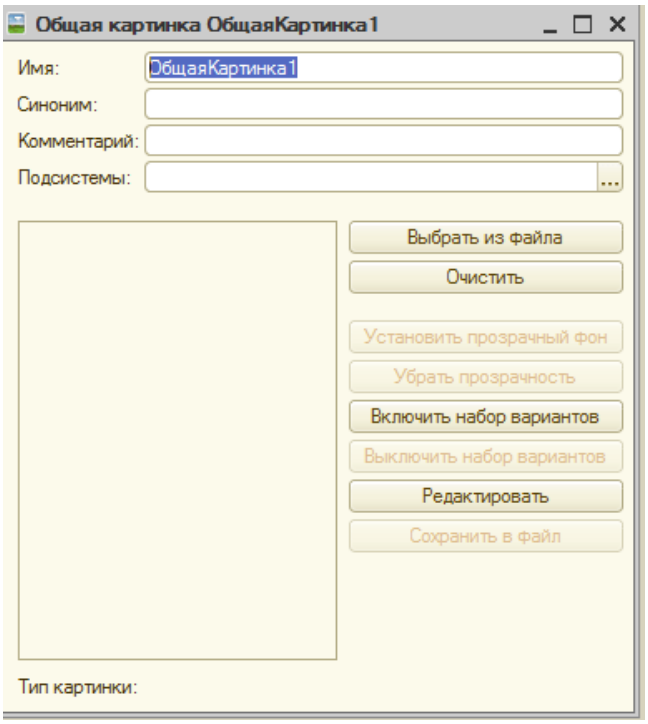
Внешняя компонента поставляется в виде dll-библиотеки, и расширяет встроенный язык новыми методами.

При работе с внешней компонентой после программной загрузки объекты внешней компоненты подключаются к "1С:Предприятию 8".

Общие картинки

Конфигуратор позволяет включать в конфигурацию графические изображения – картинки. Картинки можно размещать в некоторых элементах управления, в формах, в макетах, а также обращаться к ним при помощи встроенного языка системы «1С:Предприятие».
Если картинку планируется использовать в качестве пиктограммы в меню, панели инструментов, табличном документе и т. п., важно задать ей правильный размер, чтобы она отображалась без искажений.
Рекомендуемый размер картинок:
● для пиктограмм – не более 16×16 точек;
● для использования в табличном поле – не более 14×14 точек;
● в качестве картинки кнопки выбора поля редактирования – не более 9×9 точек;
● для картинки, которая будет использована в качестве представления подсистемы – не более 48×48 точек.
СОВЕТ. Если картинку предполагается использовать в нескольких местах, ее размер разумно ограничить меньшим значением.
Для работы с картинками предназначено окно Библиотека картинок. Для его вызова в окне Конфигурация нужно указать ветвь Общие картинки и в контекстном меню выбрать пункт Все картинки. На экран выводится окно ведения списка картинок

Для добавления новой картинки следует нажать кнопку Добавить. На экран выводится окно, в котором можно выбрать картинку из файла или открыть редактор картинок и создать новую картинку, а также выбрать или изменить для картинки прозрачный фон. Для выбора готовой картинки нужно нажать кнопку Выбрать из файла и выбрать файл, в котором расположена подготовленная заранее картинка или их набор .



Система «1С:Предприятие» позволяет использовать картинки форматов BMP, GIF, JPG, PNG, TIFF, ICO, SVG, метафайлы (WMF, EMF) или наборы картинок в zip- архиве .
Для картинок формата gif поддерживаются расширения NETSCAPE2.0 и ANIMEXTS1.0. Другие расширения не поддерживаются и поведение системы в этом случае не определено.
Для картинок формата SVG игнорируются следующие элементы:
- Анимация, java-script и любая интерактивность, в том числе css-свойства связанные с интерактивностью, например псевдокласс :hover.
- Свойства стилей:
- color-interpolation;
  - color-rendering.
- Маркеры для пути с нулевой длиной.
- Фильтр-примитив feTile.
- Элементы встроенных шрифтов:
  - font,
    - altGlyph,
    - altGlyphDef,
    - altGlyphItem,
    - glyphRef.
- Свойство enable-background.
- Ссылки на внешние ресурсы.
- Для элемента image не поддерживается относительный путь к файлу изображения.

СОВЕТ. Для картинок, которые будут использоваться в качестве пиктограмм в интерфейсе , рекомендуется использовать форматы,

поддерживающие сжатие без потерь (PNG и GIF), для того чтобы избежать искажения и минимизировать трафик между клиентом и сервером. Нужно задать имя, по которому данная картинка будет выбираться средствами встроенного языка.

Чтобы картинка при вставке хорошо вписывалась в элемент управления или форму, желательно установить ей прозрачный фон. Для этого в качестве цвета фона при редактировании картинки достаточно выбрать любой не используемый в картинке цвет, сформировать изображение и сохранить картинку. Для существующей картинки можно указать любой цвет. Благодаря установленной прозрачности данного цвета сквозь него становятся видны детали той части формы, которую закрывает область картинки.
Для установки прозрачного фона нужно нажать кнопку Установить прозрачный фон. Указатель мыши изменит вид. Подвести курсор к той части картинки, цвет которой нужно сделать прозрачным, и щелкнуть левой кнопкой мыши. Выбранный цвет становится прозрачным.
Для снятия прозрачности следует нажать кнопку Убрать прозрачность.
Кнопки Установить прозрачный фон и Убрать прозрачность доступны только для картинок форматов BMP, JPG и TIFF. Для картинок в остальных форматах данные кнопки недоступны.
Подобные действия можно выполнить с помощью ссылки Открыть в свойстве Картинка. На экран выводится окно выбора картинки. Для редактирования картинки нужно нажать кнопку Редактировать. Запускается редактор картинок.
Кнопки Включить набор вариантов и Выключить набор вариантов позволяют работать с картинками, предназначенными (в основном) для мобильных конфигураций и состоящие из набора картинок. Подробнее про такие картинки см. здесь. Нажатие кнопки Выключить набор вариантов приводит к тому, что из набора картинок удаляются все картинки, кроме базовой картинки (картинки для разрешения ldpi). Если в наборе нет картинки для разрешения ldpi, то набор становится пустым. Если в наборе есть картинка для разрешения ldpi, и она установлена как контурная, то признак шаблонности сбрасывается. Нажатие кнопки Включить набор вариантов приводит к тому, что текущая картинка становится базовой картинкой (для разрешения ldpi).
При необходимости отредактировать одну из картинок в наборе, следует нажать кнопку Редактировать. В этом случае будет открыт диалог, в котором надо выбрать одну из картинок набора, которую необходимо редактировать.

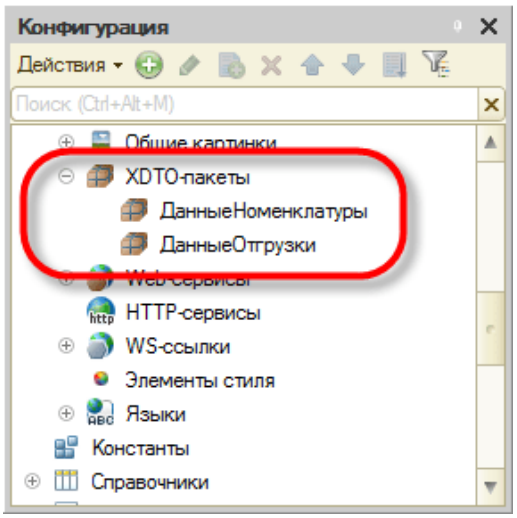
Если картинка для какого-либо разрешения не задана, поле ввода, соответствующее картинке, будет отображать пустым . Нажатие кнопки …,

в соответствующем поле диалога редактирования набора картинок, приведет к открытию диалога выбора картинки . Нажатие кнопки X приведет к очистке соответствующей картинки диалога. Редактирование отдельной картинки набора не отличается от редактирования отдельной картинки.

XDTO пакет

XDTO-пакеты — это общие объекты конфигурации. Они являются частью механизма XDTO. XDTO-пакеты позволяют описать в конфигурации систему типов и значений для взаимодействия с различными внешними источниками данных, например с Web-сервисами.

В конфигурацию может быть добавлено произвольное количество XDTO-пакетов.

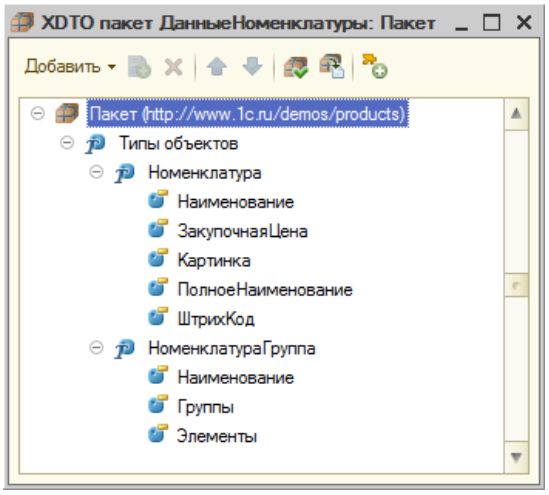


Возможно выполнение следующих действий:

- создание XDTO пакета с помощью импорта схемы XML;
- добавление нового XDTO пакета в дерево конфигурации и последующее его редактирование;
- экспорт существующего XDTO пакета в схему XML.

Также система 1С:Предприятие 8 поддерживает возможность экспорта набора XML схем данных конфигурации.

Редактирование XDTO пакета осуществляется в специальном окне, отображающем иерархическую структуру XDTO-пакета:


XDTO пакет ДанныеНоменклатуры: Пакет
Добавить
Пакет (http://www.1c.ru/demos/products)
Типы объектов
Номенклатура
Наименование
ЗакупочнаяЦена
Картинка
ПолноеНаименование
ШтрихКод
НоменклатураГруппа
Наименование
Группы
Элементы

Web-сервисы

Web-сервисы — это один из механизмов платформы, используемых для интеграции с другими информационными системами. Он является средством поддержки SOA (Service-Oriented Architecture) — сервис-ориентированной архитектуры, которая является современным стандартом интеграции приложений и информационных систем.

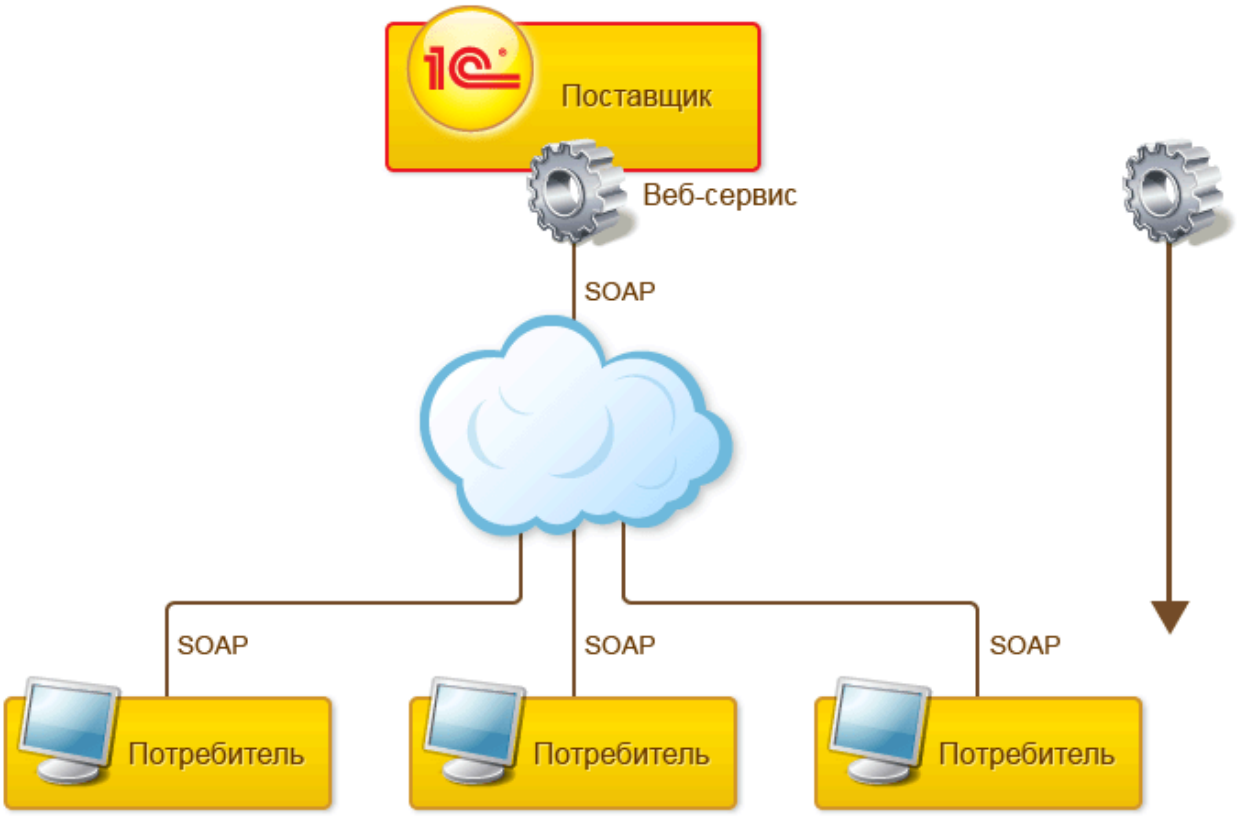
Значительным преимуществом сервис — ориентированной архитектуры является то, что она позволяет развивать инфраструктуру предприятия однородным образом, без разрушения уже существующих решений. Ее использование позволяет минимизировать издержки за счет интеграции разнородных и унаследованных систем в современный ландшафт предприятия. Она позволяет реализовывать слабо связные программные компоненты с тем, чтобы максимально повысить их повторную используемость.

Сервис-ориентированная архитектура интенсивно развивается и поддерживается крупными вендорами. Она строится на базе сервисов, автономных или управляемых извне. Предпочтительным способом их реализации являются веб-сервисы. Они независимы от платформы, автономны и поддерживаются повсеместно.

Прикладное решение 1С:Предприятия 8 может являться как поставщиком веб-сервисов, так и потребителем веб-сервисов, опубликованных другими поставщиками.

- 1С:Предприятие — поставщик веб-сервисов
  В конфигурацию можно добавить специальный объект, — Web-сервис, — с помощью которого описать некоторую функциональность прикладного решения, например, получение списка имеющихся на некотором складе товаров, их количества и цен. После публикации на веб-сервере такой сервис будет

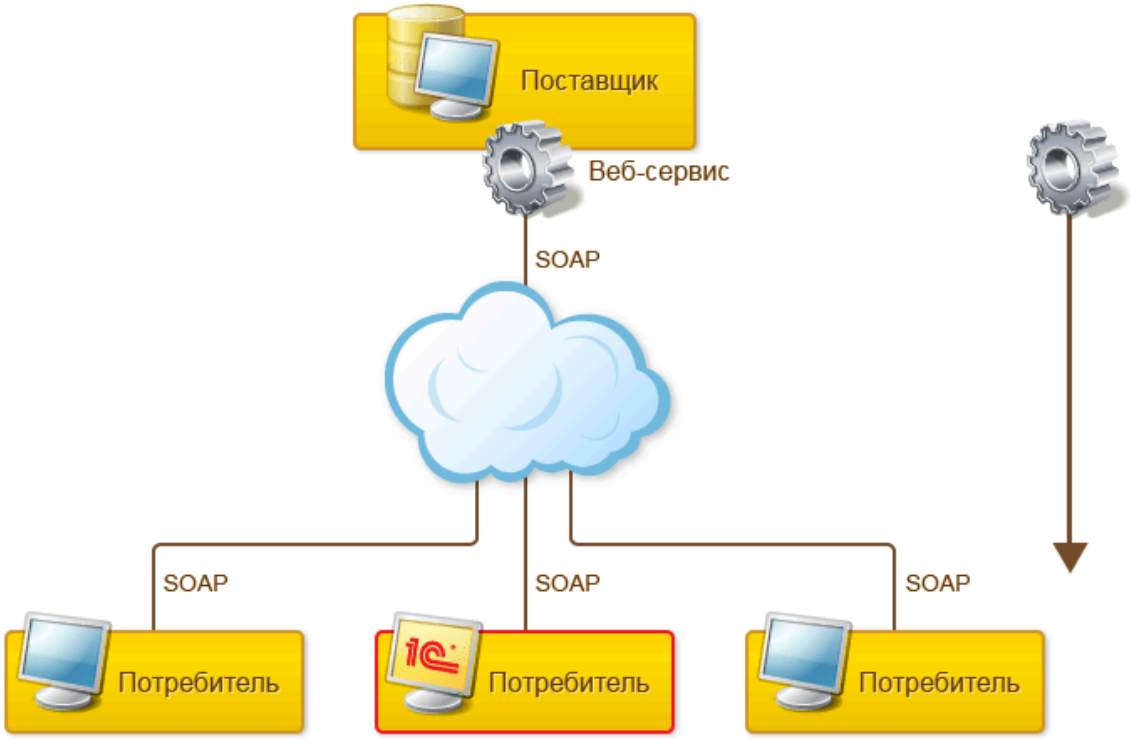
доступен сторонним потребителям.



В качестве потребителей могут выступать системы, использующие произвольные аппаратные и программные платформы. Технология веб-сервисов является платформенно независимой.

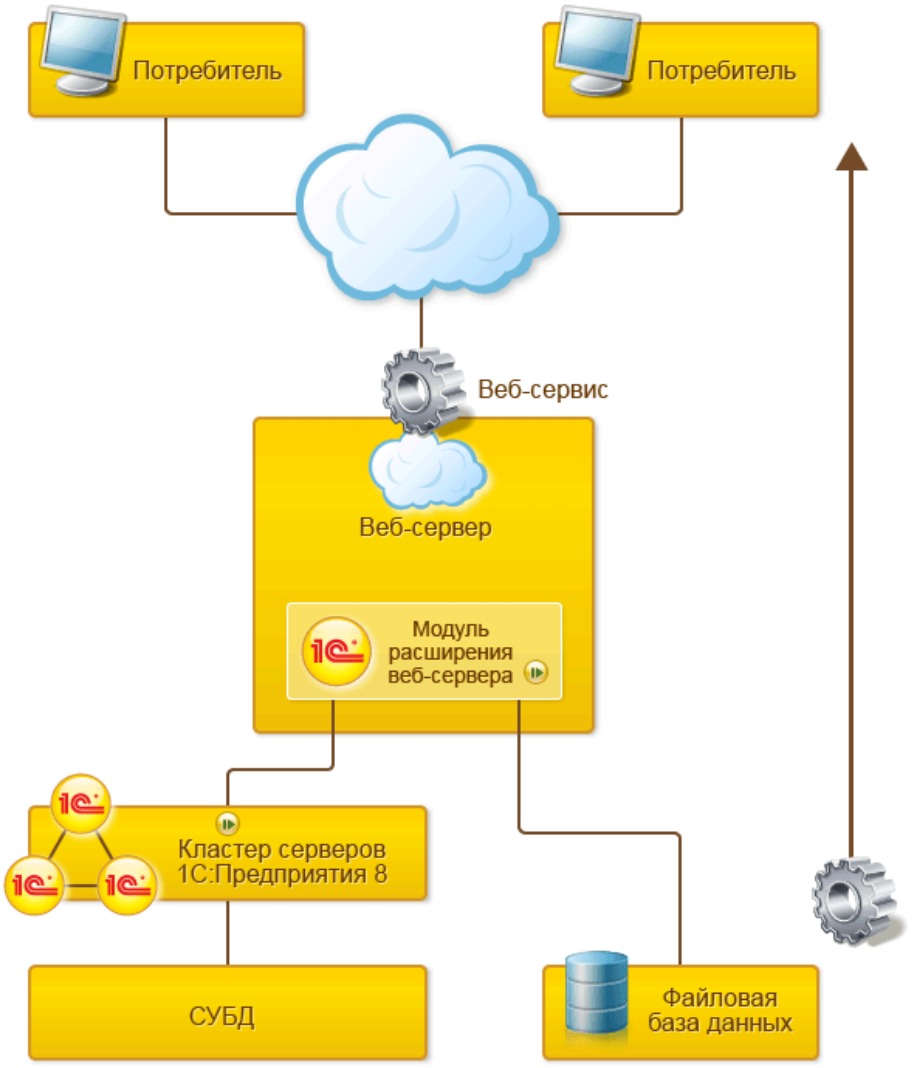
- 1С:Предприятие — потребитель веб-сервисов
  В прикладном решении можно описать ссылку на веб-сервис, опубликованный сторонним поставщиком. После этого прикладное решение сможет использовать данные, получаемые с помощью такого веб-сервиса, в своих внутренних

прикладных алгоритмах. Подробнее...



Техническая реализация web-сервисов

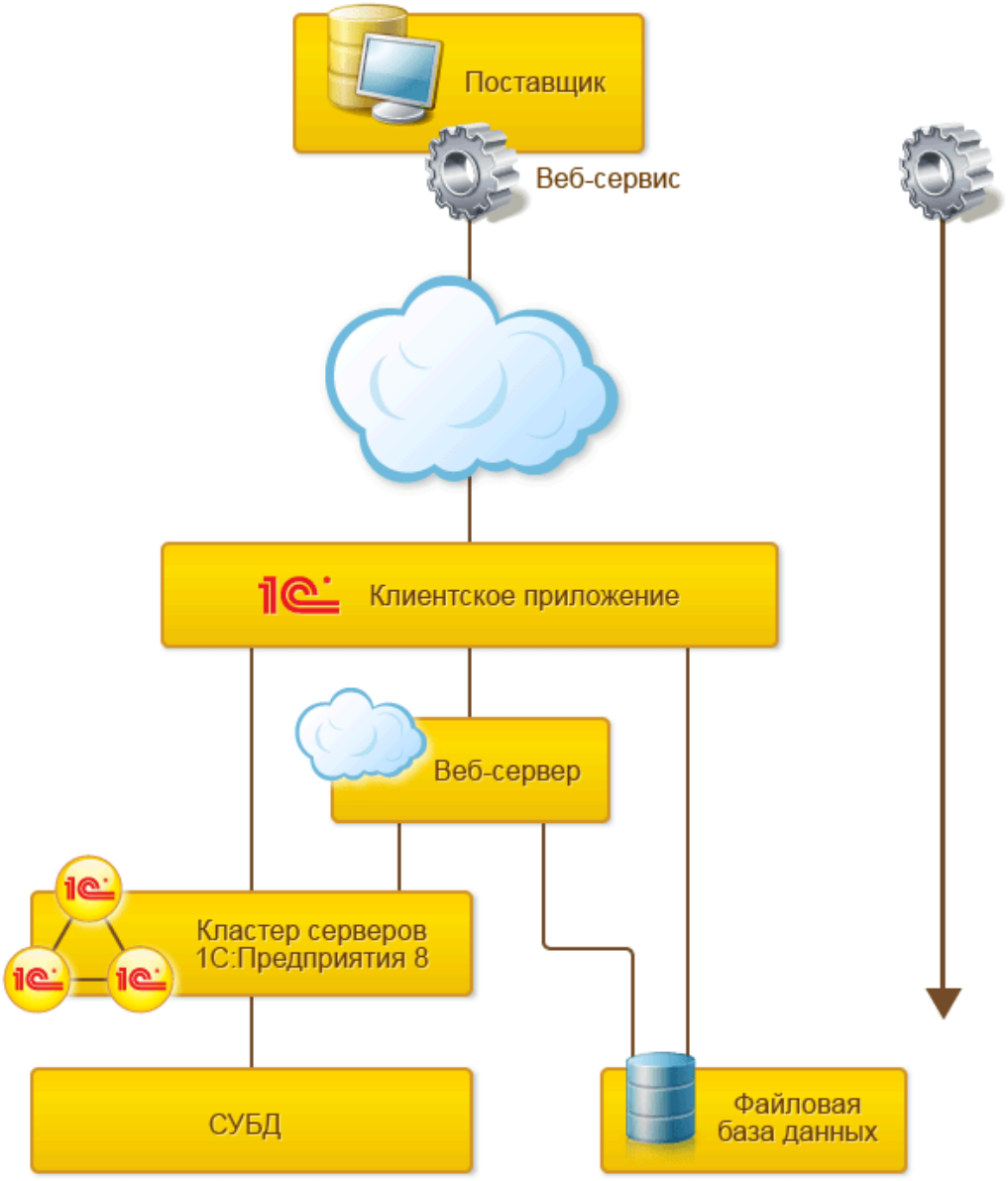
Если прикладное решение является поставщиком веб-сервиса то и в файловом, и в клиент-серверном варианте работы взаимодействие между прикладным решением и потребителями веб-сервиса осуществляется через веб-сервер, с помощью модуля расширения веб-сервера.



При этом, когда потребитель обращается к web-сервису прикладного решения, выполняется модуль web-сервиса. Этот модуль содержится в конфигурации и в нем располагаются процедуры, выполняемые при вызове тех или иных операций web-сервиса.

В случае клиент-серверного варианта работы этот модуль будет исполняться в кластере. В случае файлового варианта работы — в модуле расширения веб-сервера.

Если прикладное решение является потребителем веб-сервиса стороннего поставщика, то в этом случае взаимодействие между прикладным решением и поставщиком веб-сервиса осуществляет клиентское приложение. Оно вызывает те или иные операции веб-сервиса и обрабатывает полученные данные.



Платформа 1С:Предприятия, выступающая в роли потребителя веб-сервиса, поддерживает оптимизированную передачу двоичных данных по протоколу MTOM. Включение режима MTOM в 1С:Предприятии происходит автоматически.

HTTP-сервисы

В дополнение к автоматическому REST интерфейсу прикладного решения в платформе существует возможность создания собственных произвольных HTTP-сервисов в прикладном решении.

Разработчик самостоятельно, с помощью встроенного языка, формирует ответ на запрос. При этом есть удобный доступ к телу, заголовкам и строке исходного запроса, а также есть возможность формировать код, тело и заголовки ответа по своему усмотрению.

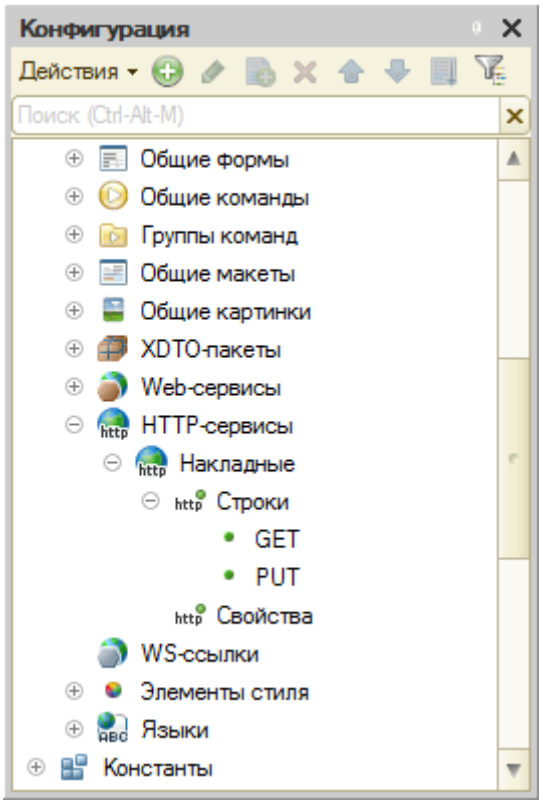
По сравнению с имеющимися в платформе SOAP web-сервисами, HTTP-сервисы имеют ряд преимуществ:

- Простота программирования клиента таких сервисов;
- Потенциально меньший объем передаваемых данных;
- Потенциально меньшая вычислительная нагрузка;
- HTTP-сервисы ориентированы на «ресурсы», в то время как SOAP сервисы ориентированы на «действия».

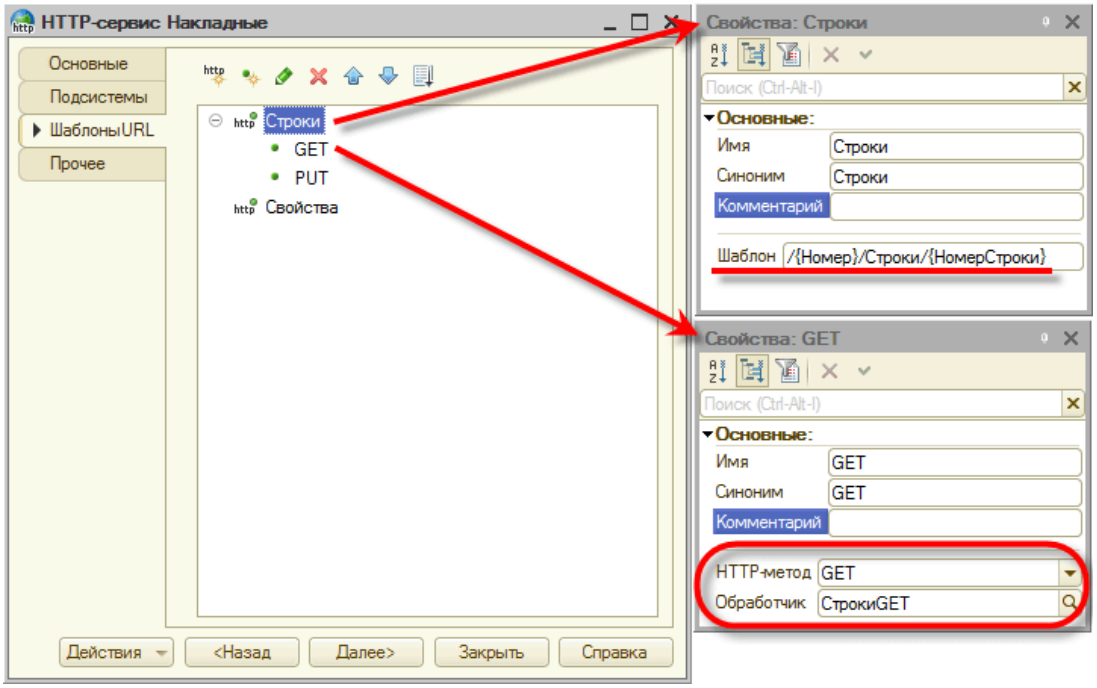
Первые три фактора особенно важны для приложений, работающих на мобильных устройствах.

Можно использовать HTTP-сервисы как «легкие» RPC-сервисы, не требующие сложной подготовки XML-пакетов. Методы могут идентифицироваться в URL, а параметры могут передаваться в опциях запроса, или в его теле.
В последнем случае такие сервисы уже вплотную приближаются как SOAP, проигрывая ему в четкости спецификации, но выигрывая в гибкости.

По своему «конструктивному исполнению» HTTP-сервисы очень напоминают web-сервисы, имеющиеся в платформе. Точно так же есть специальный объект конфигурации HTTP сервис. Такие объекты добавляются в ветку Общие — HTTP-сервисы:



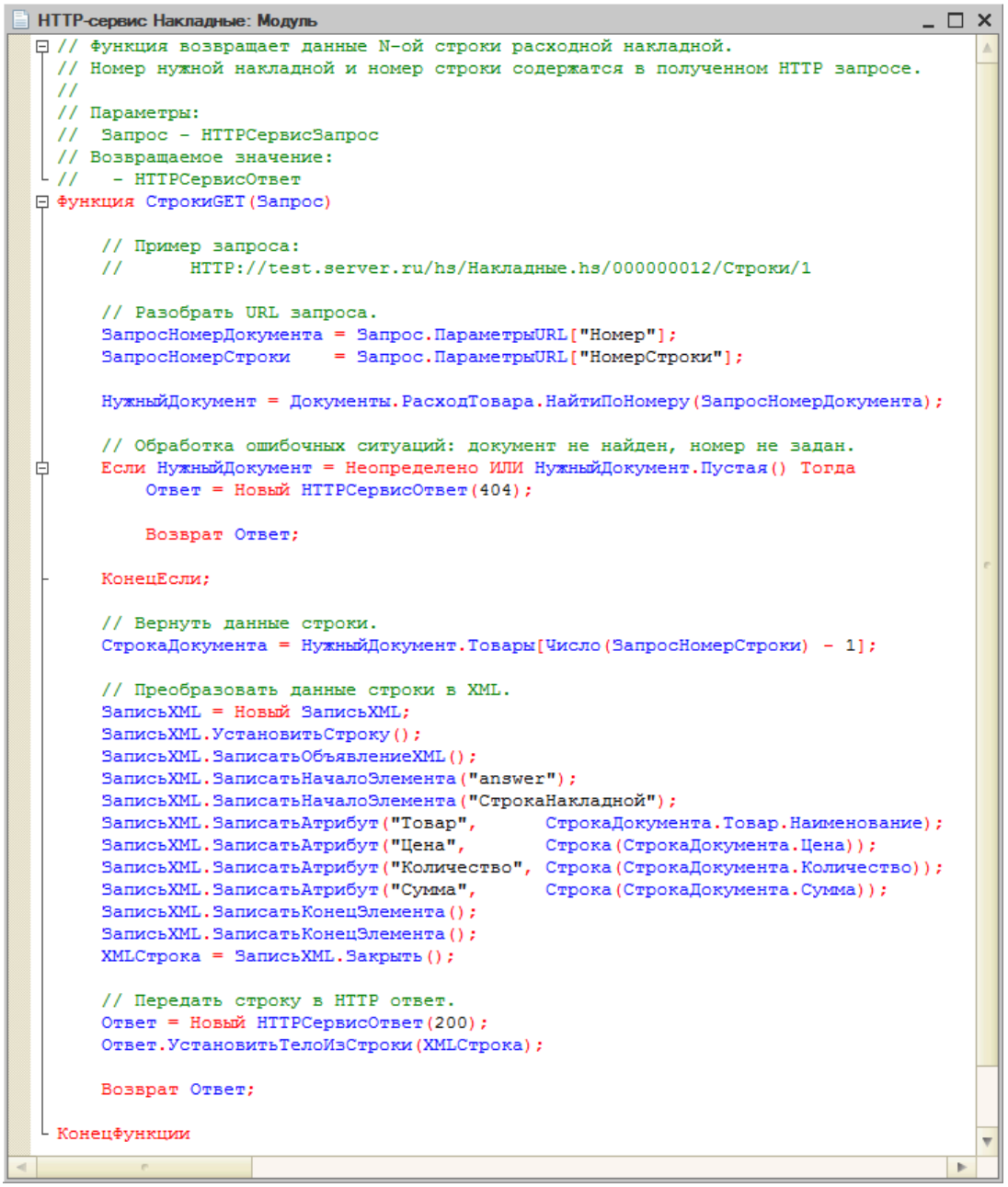
Каждый HTTP-сервис может содержать в себе один или несколько **шаблонов**. Для каждого шаблона можно создать один или несколько **методов**, выполняющих обработку данных:



Шаблон задаёт путь, по которому может происходить обращение к HTTP-сервису. В шаблоне можно использовать определённый набор символов, в том числе параметризованные сегменты вида **{какой-то текст}**.

Для каждого метода указывается, во-первых, обрабатываемый HTTP метод, а также создаётся процедура на встроенном языке, которая и будет выполнять обработку данных. Также можно указать, что будет обрабатываться не какой-то конкретный, а любой HTTP-метод из доступных.

При обращении к такому HTTP-сервису платформа сначала попытается сопоставить URL, по которому произошло обращение, с одним из имеющихся шаблонов и методов. Если сопоставить не удалось, то платформа выдаст код ответа **404 Not Found**. Если подходящий метод будет найден, то платформа начнёт выполнение его обработчика, передав в него все имеющиеся в запросе данные в виде объекта встроенного языка **HTTPСервисЗапрос**:



Из этого объекта можно легко получить, например, параметры, содержащиеся в исходном URL, и использовать их для извлечения из базы нужных данных.

Полученные данные можно вернуть в разных форматах. Например, их можно преобразовать в XML, как на картинке выше, или даже просто в текстовую строку с разделителями.

Ответ сервиса формируется специальным объектом встроенного языка **HTTPСервисОтвет**, в тело которого можно поместить подготовленные данные.

Публикация HTTP-сервисов выполняется аналогично тому, как публикуются web-сервисы. Также аналогичным образом для них работает аутентификация, использование разделения данных и отладка.

WS-ссылки 1С

WS-ссылка 1С – это описание стороннего веб-сервиса по его WSDL описанию.

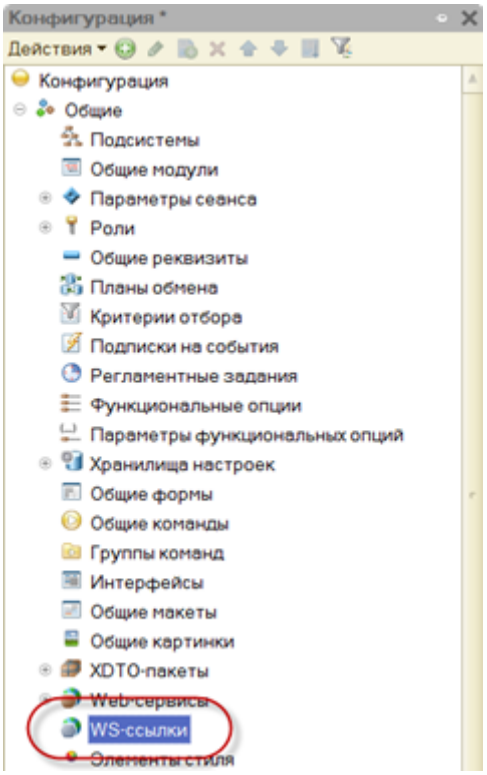
Это может быть веб-сервис другой базы 1С. Это может быть веб-сервис другой программы или веб-сайта.

Для нас это совершенно не важно – важно, что есть набор функций, которые мы можем вызвать через интернет.

Соответственно нам интересно – как добавить ссылку на веб-сервис и как вызывать его методы.

Добавление WS-ссылки 1С

WS-ссылки 1С это объект 1С, который расположен в конфигурации в ветке Общие/WS-ссылки 1С.



Добавление ссылки на веб сервис производится автоматически. Нажмите правой кнопкой на ветку, выберите Добавить, укажите путь к WSDL описанию веб сервиса.

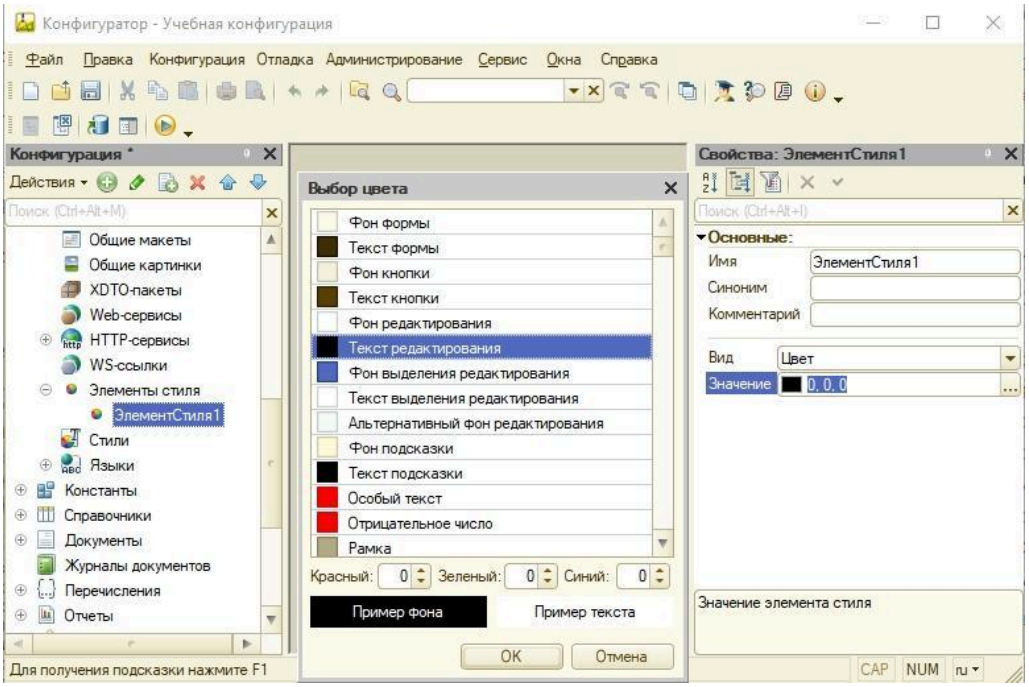
В результате добавления WS-ссылки 1С будет создана автоматически такая же структура, как создается вручную при добавлении Web сервиса.
Добавление производится в конфигураторе и если у Вас используется прокси, то в вместо корректного добавления Web-ссылки Вы увидите ошибку с текстом вроде «Extra content at the end of the document».

Эта замечательная надпись может конечно обозначать и то, что она пишет – что формат XML в описании веб-сервиса не верный, но чаще всего значит просто – 1С не смогла произвести соединение с веб-сервисом, и чаще всего по причине прокси.

Ссылку на веб сервис в конфигурацию добавлять не обязательно. Возможно создавать ее динамически в момент выполнения кода на языке 1С (указав URL WSDL описания). В этом случае на импорт WSDL в момент создания объекта будет потрачено определенное время.

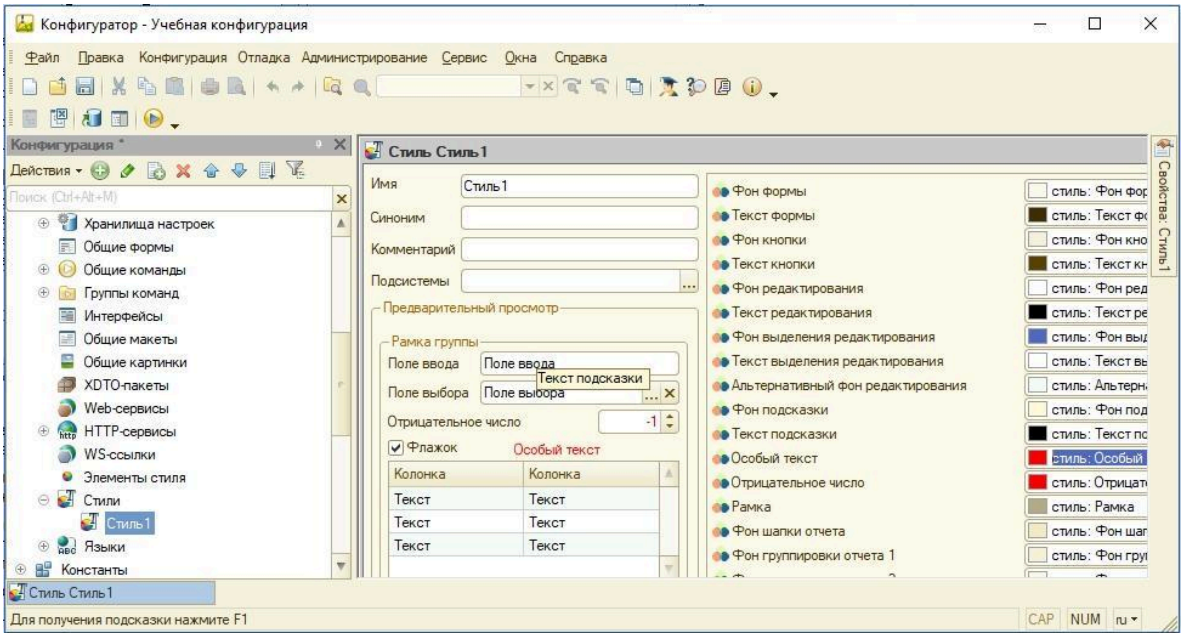
-**"Элементы стиля"** — набор определенных заранее параметров оформления элементов формы (цвет, шрифт, рамка), ссылка на который используется в оформлении прикладного решения. При смене стиля в этом объекте конфигурации, стиль автоматически меняется во всей конфигурации.

Элементы стилей, созданные для данной конфигурации, располагаются в списке стилей под стандартными стилями.



Объект 1С "Стиль" - это совокупность различных настроек (элементов стиля) для оформления форм, текстов, рамок и других составляющих элементов управления.

Стиль (в конфигураторе) представляет собой стандартный набор элементов (см. рисунок ниже), состав которого не подлежит изменению.



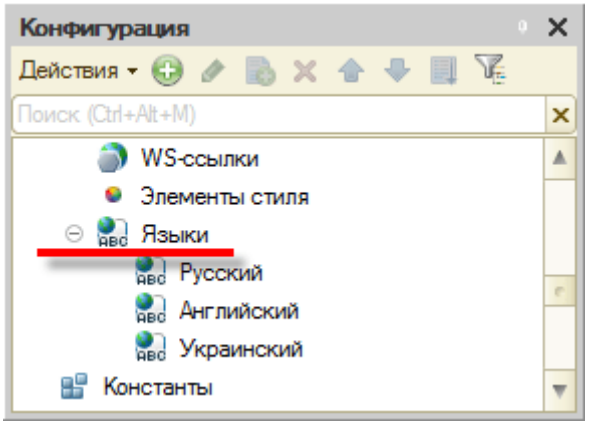
Использование стиля позволяет быстро выбрать нужное оформление. При редактировании стиля внешний вид формы отображается в окне просмотра, что помогает сразу же оценивать вносимые изменения.

Конфигуратор позволяет создать произвольное число стилей, а также дополнительных элементов стилей. Приемы создания, редактирования свойств и удаления объектов конфигурации типа "Элементы стиля" и "Стили" совпадают с общими приемами управления объектами конфигурации.

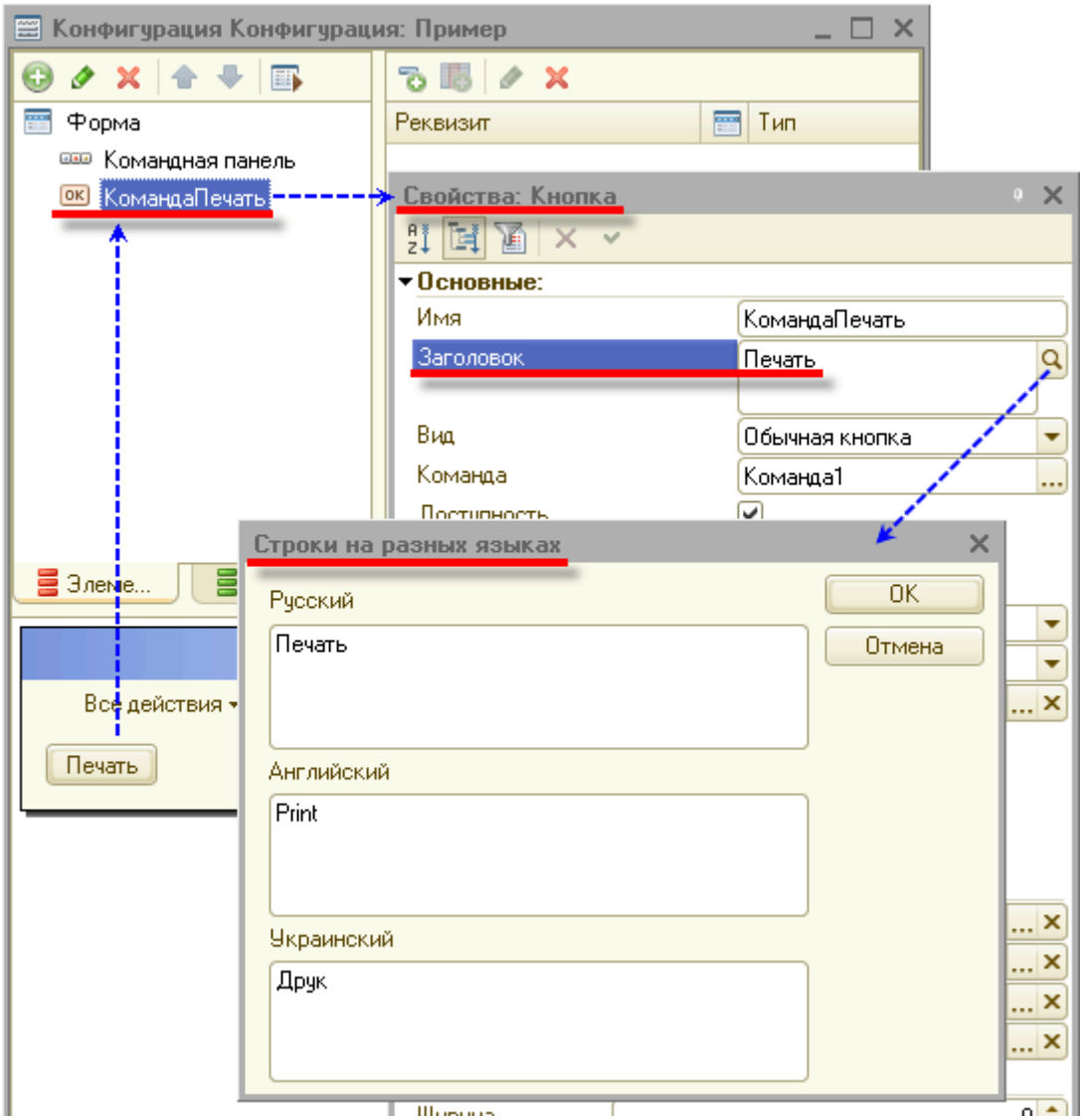
Язык

Язык — это один из общих объектов конфигурации. Языки предназначены для создания интерфейса прикладного решения на разных языках.

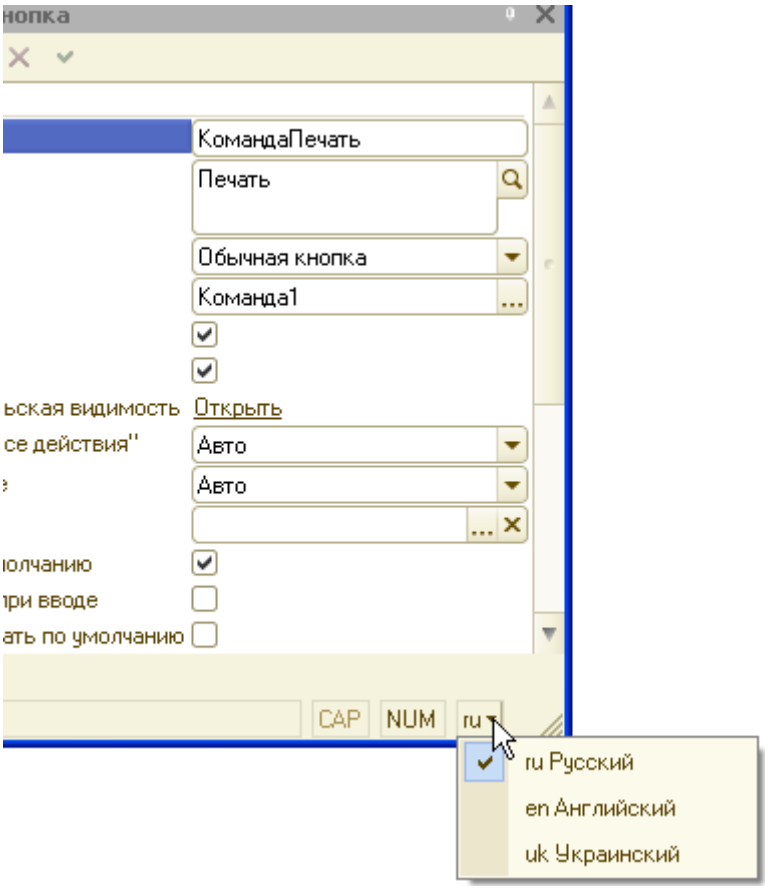
В конфигурацию можно добавить несколько различных языков:



Поскольку все тексты конфигурации и базы данных хранятся в формате UNICODE, разработчик может указывать для одной и той же надписи различные варианты ее отображения на каждом из этих языков.



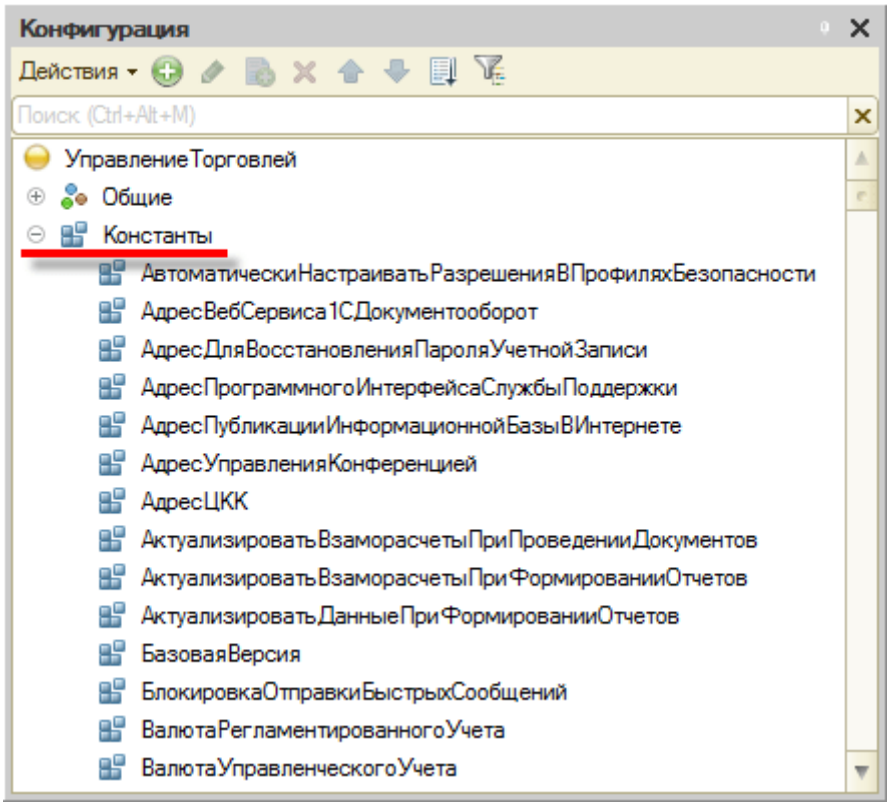
Если разработчик создает прикладное решение на нескольких языках, он может при помощи мыши переключаться с одного языка на другой. Эта возможность очень удобна при разработке форм, т. к. позволяет быстро увидеть внешний вид формы, скажем, сначала на английском языке, а затем на русском.


нопка
КомандаПечать
Печать
Обычная кнопка
Команда1
ьская видимость
Открыть
се действия"
Авто
Авто
юлчанию
ри вводе
ать по умолчанию
CAP
NUM
ru
ru Русский
en Английский
uk Украинский

Константа

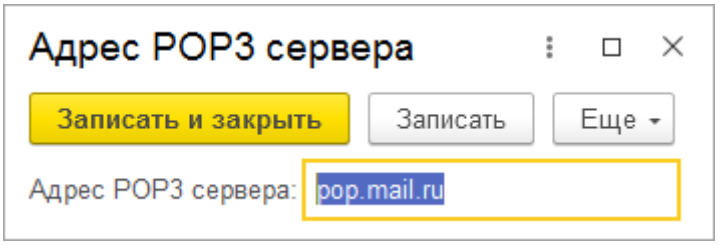
Константы — это прикладные объекты конфигурации. Они позволяют хранить в информационной базе данные, которые не изменяются во времени, или изменяются очень редко. Каждая константа позволяет хранить одно значение.

Например, в константе может храниться наименование предприятия, его ИНН и другая информация. В прикладном решении может быть создано произвольное количество констант:

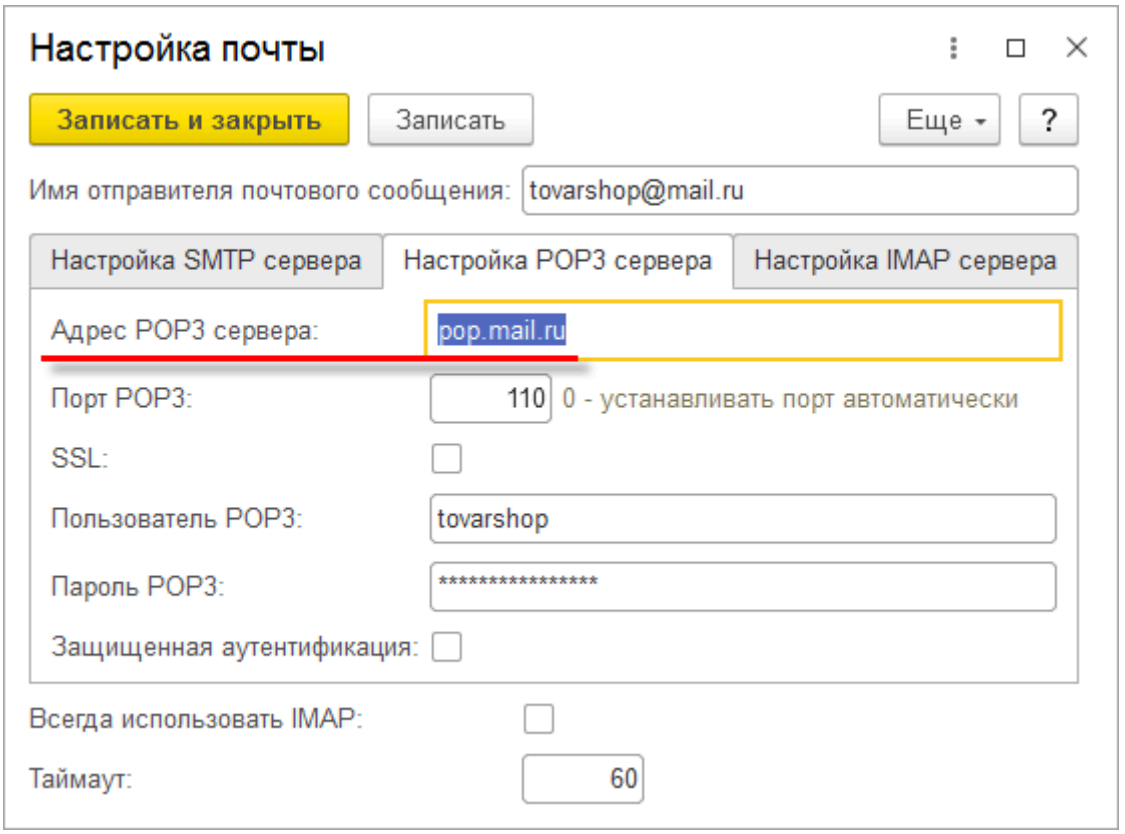


Форма констант

Для того, чтобы пользователь мог просматривать и изменять значения констант, система может автоматически сгенерировать форму константы:



Наряду с этим разработчик имеет возможность создать собственную форму, которую система будет использовать вместо формы по умолчанию:

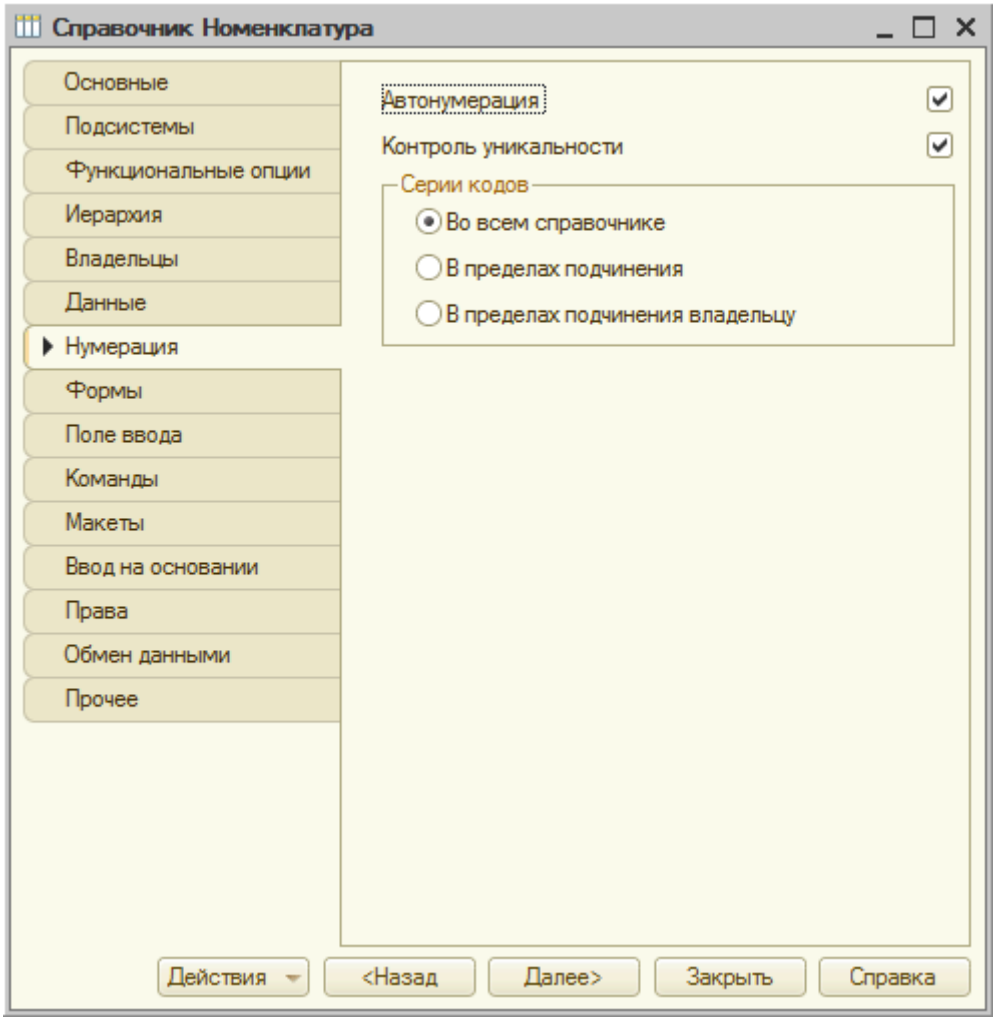

Настройка почты
Записать и закрыть
Записать
Еще
?
Имя отправителя почтового сообщения:
tovarshop@mail.ru
Настройка SMTP сервера
Настройка POP3 сервера
Настройка IMAP сервера
Адрес POP3 сервера:
pop.mail.ru
Порт POP3:
110
0 - устанавливать порт автоматически
SSL:
Пользователь POP3:
tovarshop
Пароль POP3:
****
Защищенная аутентификация:
Всегда использовать IMAP:
Таймаут:
60

Справочники

Справочники — это прикладные объекты конфигурации. Они позволяют хранить в информационной базе данные, имеющие одинаковую структуру и списочный характер. Это может быть, например, список сотрудников, перечень товаров, список поставщиков или покупателей.

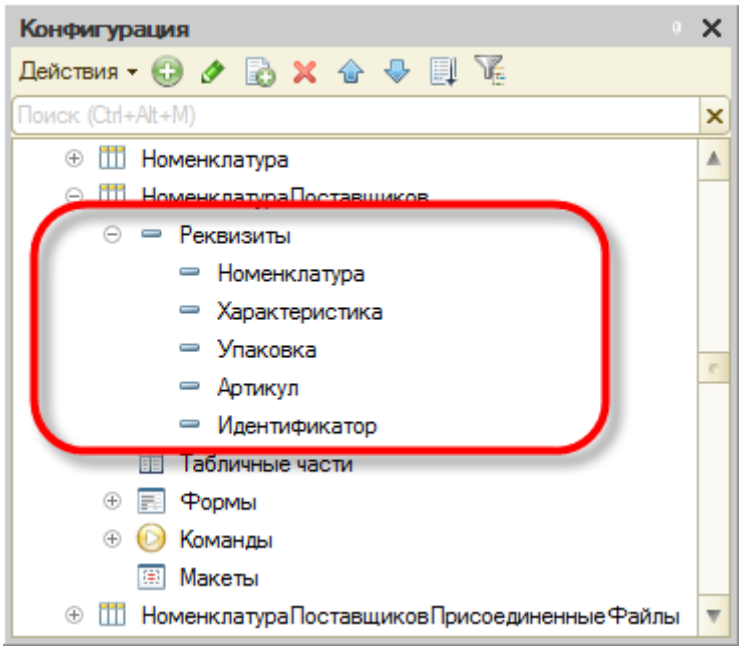
Структура справочника

Каждый элемент справочника характеризуется кодом и наименованием. Система поддерживает режим автоматической нумерации элементов, при котором она самостоятельно может генерировать код для нового элемента справочника. Кроме этого система позволяет осуществлять контроль уникальности кодов справочника, не разрешая создавать элементы с одинаковыми кодами:

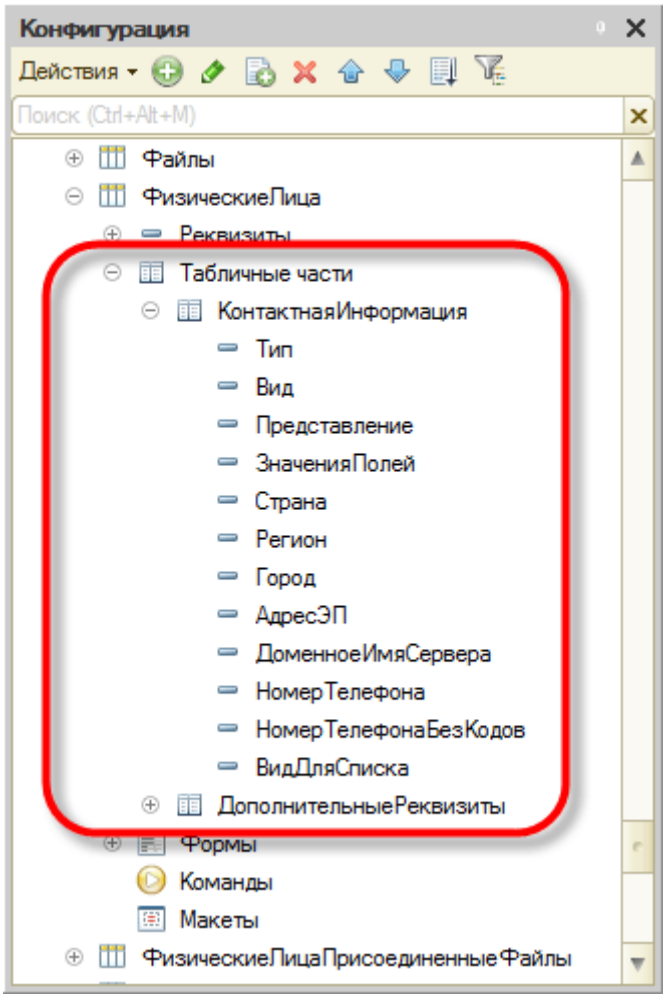


Помимо кода и наименования, каждый элемент справочника, как правило, содержит некоторую дополнительную информацию, которая подробно описывает этот элемент. Например, для товара это может быть информация об артикуле, упаковке и т. п. Набор такой информации является одинаковым

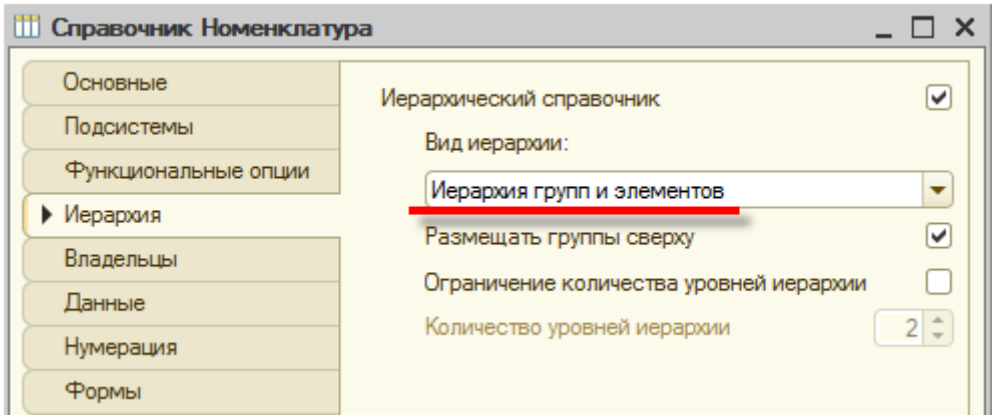
для всех элементов конкретного справочника, и для ее хранения служат реквизиты справочника:



Кроме этого, каждый элемент справочника может содержать некоторый набор информации, которая одинакова по своей структуре, но различна по количеству, для разных элементов справочника. Например, для каждого сотрудника в справочнике **Физические лица** это может быть контактная информация или информация о составе семьи, образовании. Для хранения подобных данных служат табличные части справочника:

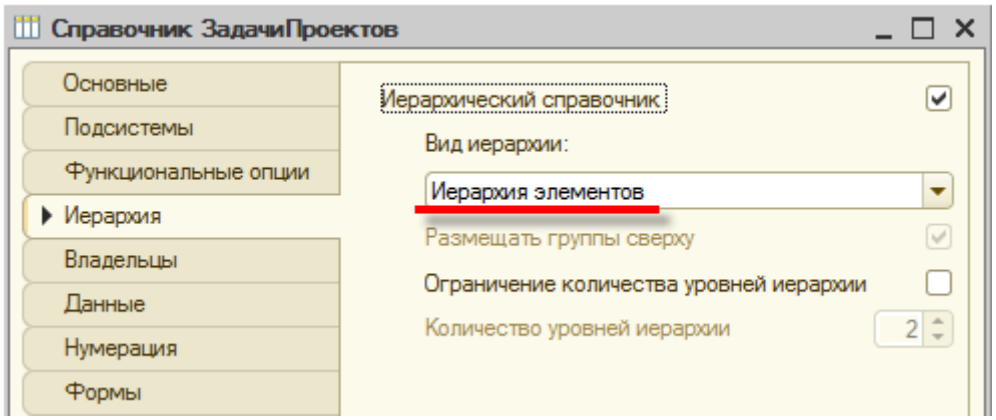


Справочники могут поддерживать иерархическое расположение элементов. Например, в справочнике **Номенклатура** могут быть созданы группы: **Бытовая техника**, **Обувь**, **Продукты** и т. д., в которых располагаются элементы, относящиеся к этим группам. Кроме того группы справочника могут включать в себя и другие группы, создавая тем самым многоуровневую иерархическую структуру.

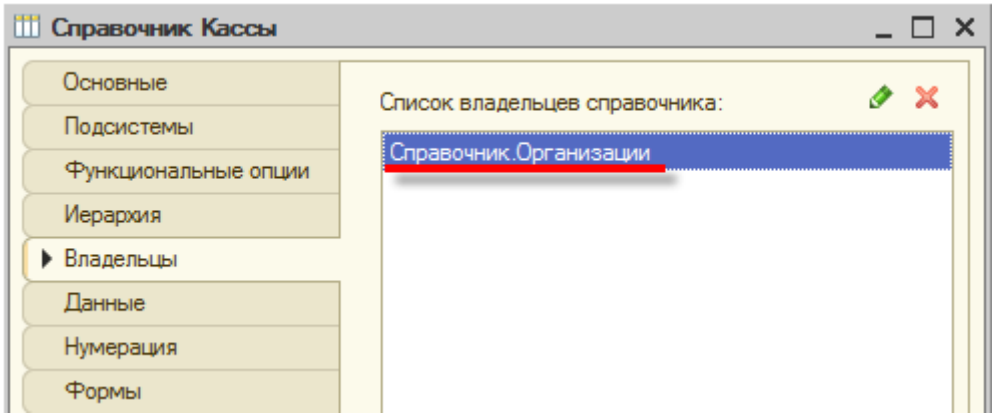


В справочниках возможен и другой вид иерархии, при котором элементы справочника будут относиться не к группам, а к другим элементам этого же справочника. Такой вид иерархии может использоваться, например, при

создании справочника **Задачи**, где одна задача может включать в себя несколько других задач.

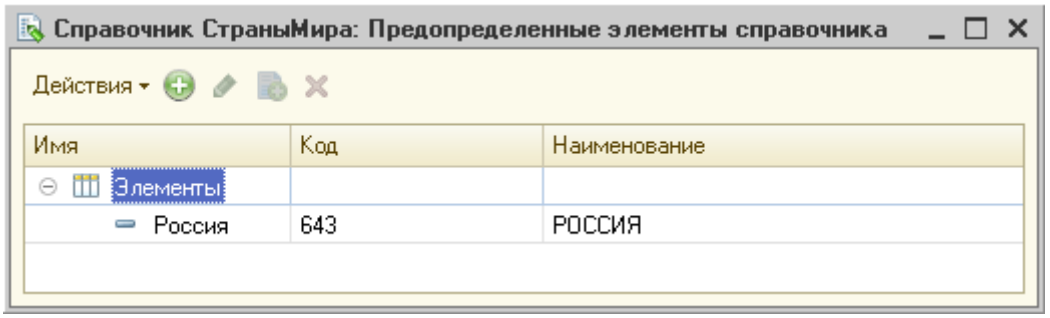


Разные справочники могут находиться в состоянии подчинения, т. е. элементы одного справочника могут быть подчинены элементам или группам другого справочника. Например, справочник **Кассы** может быть подчинен справочнику **Организации**. Тогда при оформлении кассовых документов для некоторой организации можно будет выбрать кассу не среди всех имеющихся в программе касс, а среди касс, существующих только в этой организации:



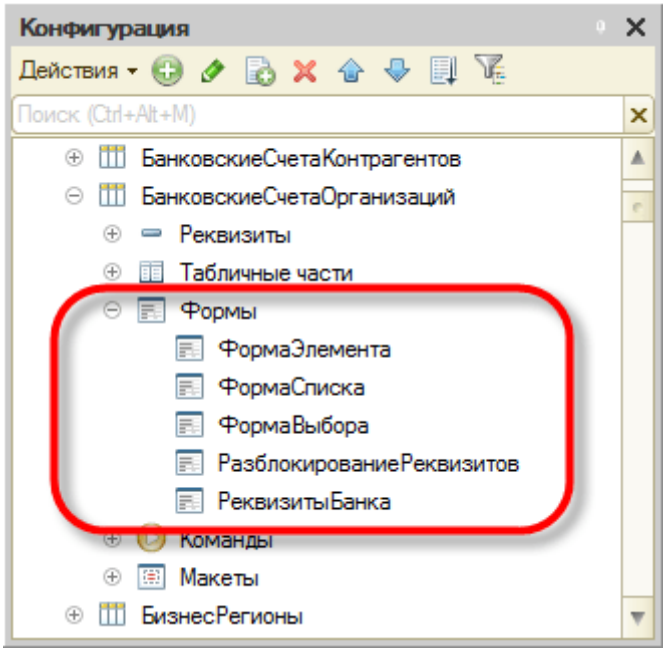
Справочники допускают также создание предопределенных элементов, которые существуют в справочнике всегда, вне зависимости от действий пользователя. Такие элементы справочника создаются разработчиком при разработке прикладного решения и не могут быть удалены или перемещены пользователем.

Например, если в основном прикладное решение будет внедряться на предприятиях одного государства, России, в справочник **Страны мира** сразу же можно добавить предопределенный элемент **Россия**, поскольку он понадобится подавляющему большинству пользователей.



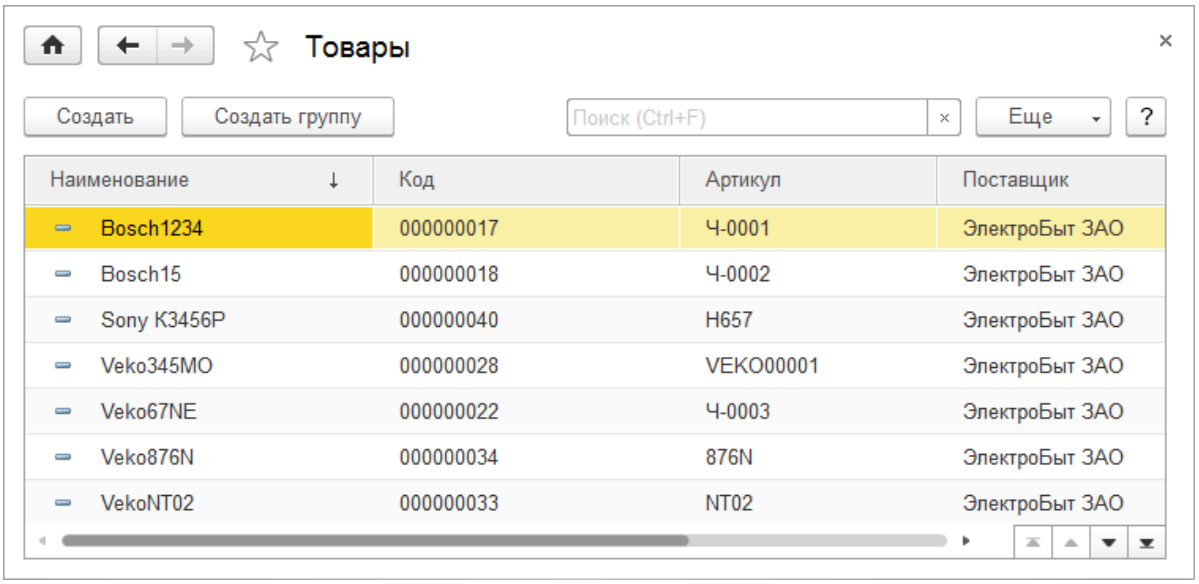
Формы справочника

Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в справочнике, система поддерживает несколько форм представления справочника. Система может автоматически генерировать все нужные формы справочника. Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию:



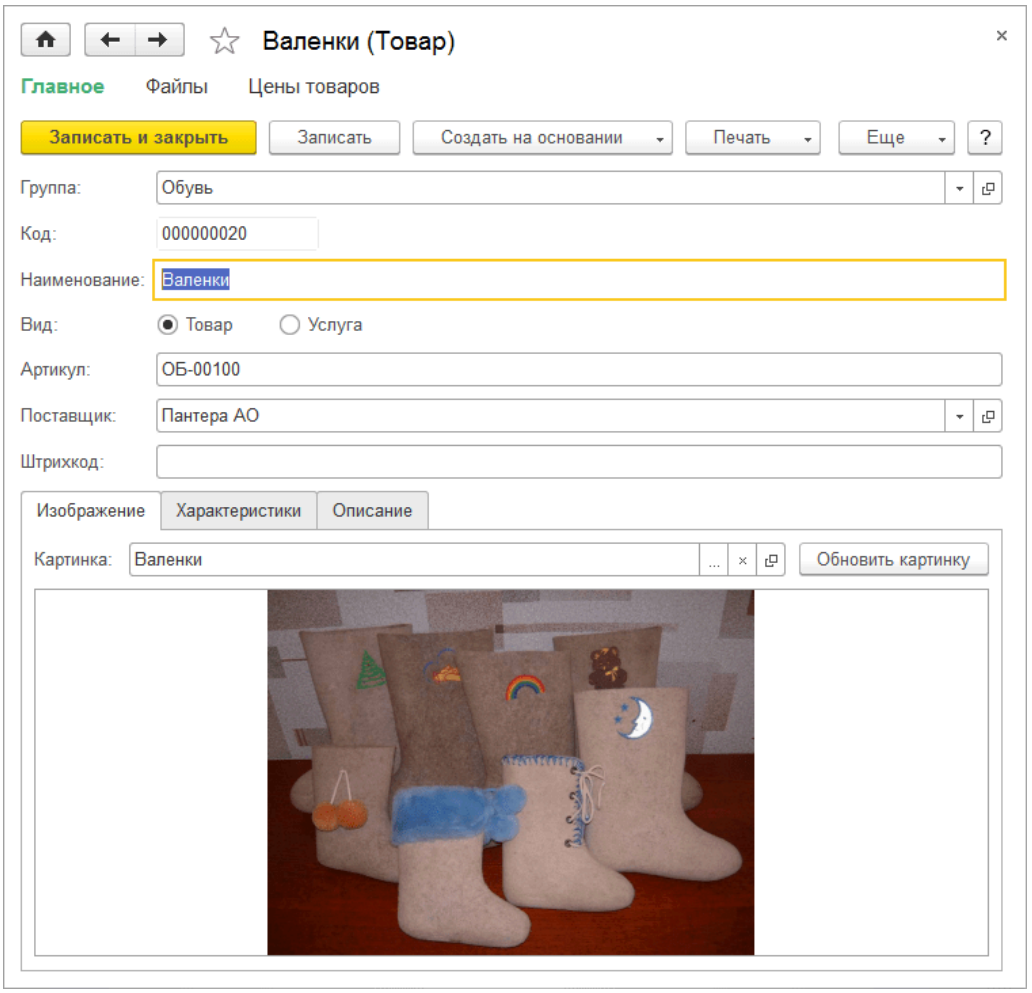
Форма списка

Для просмотра данных, содержащихся в справочнике, используется форма списка. Она позволяет выполнять навигацию по справочнику, добавлять, помечать на удаление и удалять элементы и группы справочника, перемещать элементы и группы. Форма списка может представлять данные в иерархическом и не иерархическом виде и позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



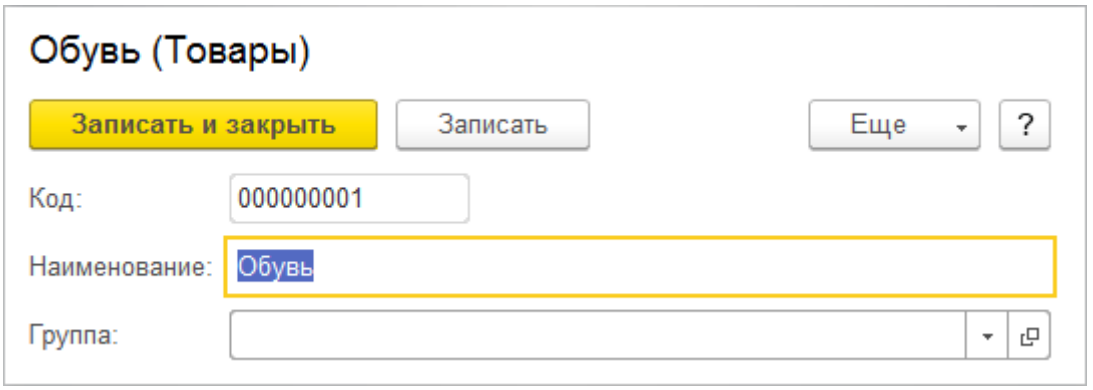
Форма элемента

Для просмотра и изменения данных отдельных элементов справочника используется форма элемента. Как правило, она представляет данные в удобном для восприятия и редактирования виде:



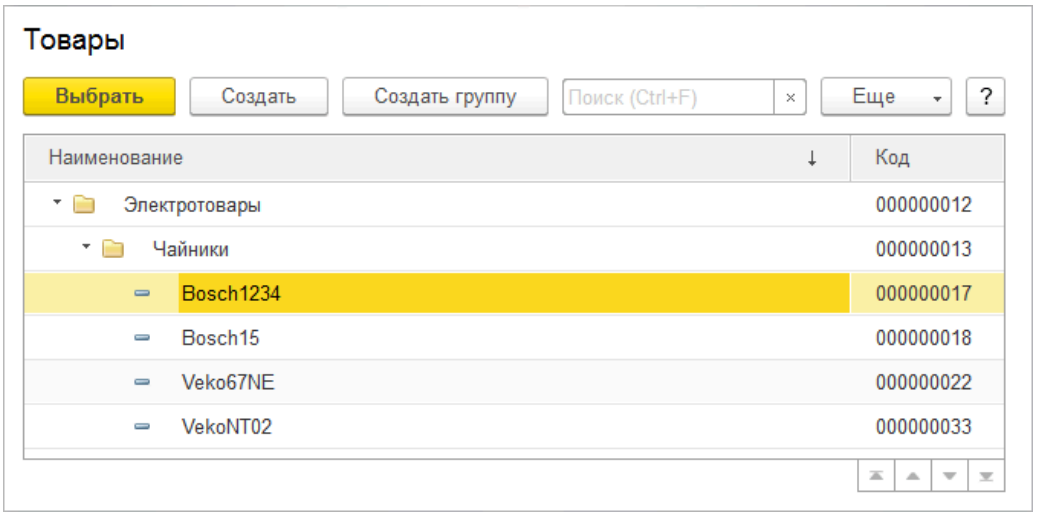
Форма группы

Система поддерживает отдельный вид формы для групп справочника. Это связано с тем, что, как правило, состав информации, относящейся к группе справочника, намного меньше, чем состав данных, имеющих отношение к отдельным элементам справочника:



Форма выбора, форма выбора группы

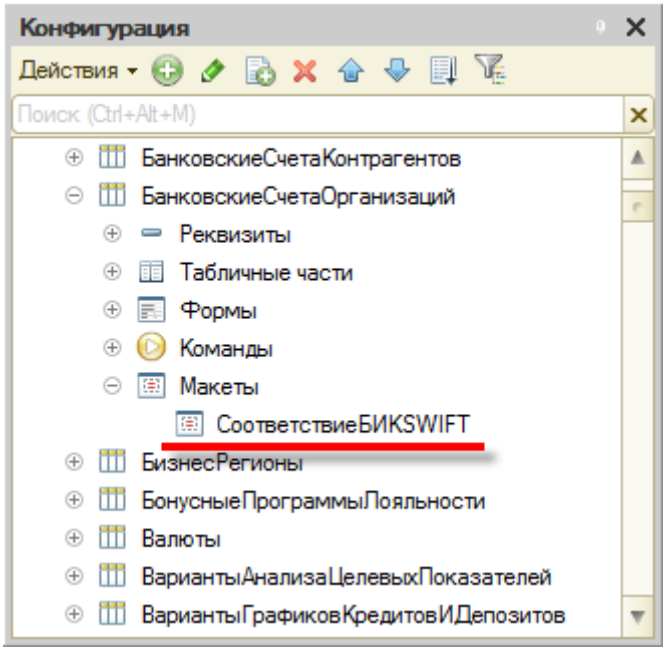
При различных действиях пользователя в прикладном решении возникает необходимость выбора конкретных элементов справочников. Это может потребоваться, например, при создании приходной накладной, в которой должны быть перечислены определенные товары. В этом случае система использует форму выбора справочника, которая, обычно, содержит минимальный набор информации, необходимой для выбора того, или иного элемента:



Кроме всех перечисленных форм, система поддерживает еще форму выбора группы справочника, предназначенную для выбора только среди групп, содержащихся в справочнике.

Макеты

Справочнику могут быть сопоставлены несколько макетов, содержащих данные, необходимые для обеспечения работы справочника:



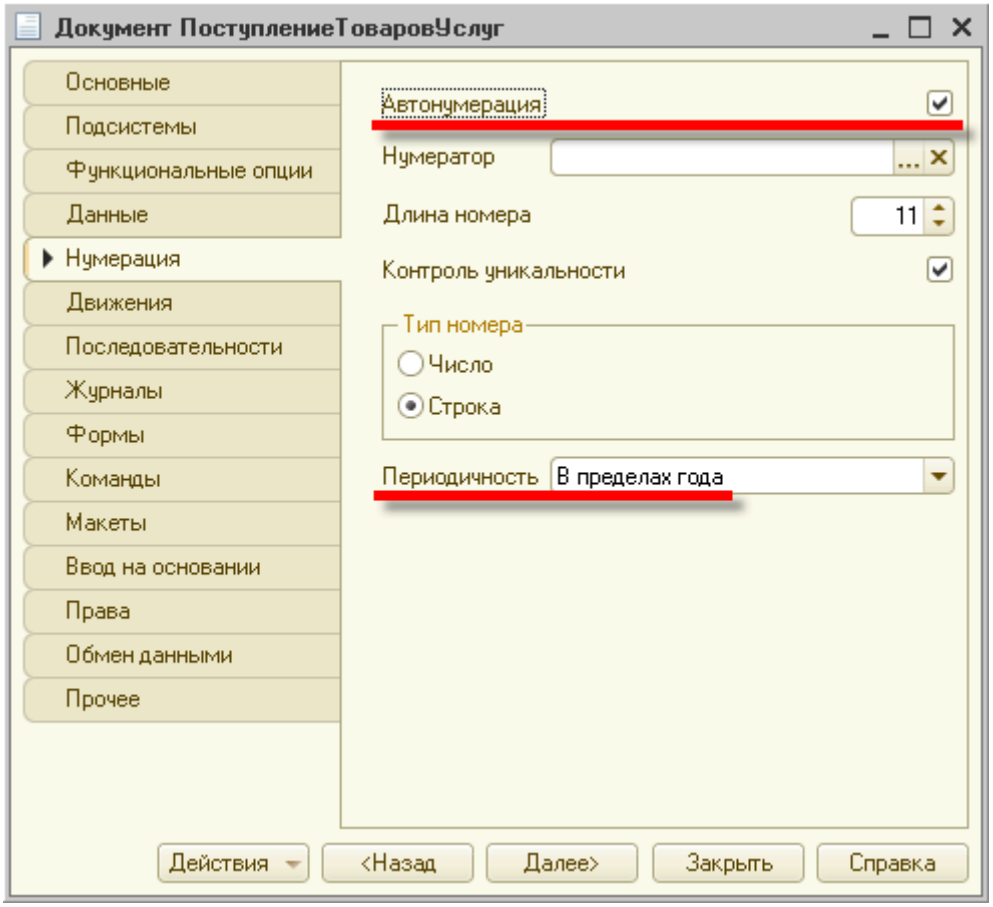
Макеты могут использоваться для формирования печатных форм справочника или для отображения дополнительной информации, имеющей отношение к справочнику.

Документы

Документы — это прикладные объекты конфигурации. Они позволяют хранить в прикладном решении информацию о совершенных хозяйственных операциях или о событиях, произошедших в «жизни» предприятия вообще. Это могут быть, например, приходные накладные, приказы о приеме на работу, счета, платежные поручения и т. д.

Структура документа

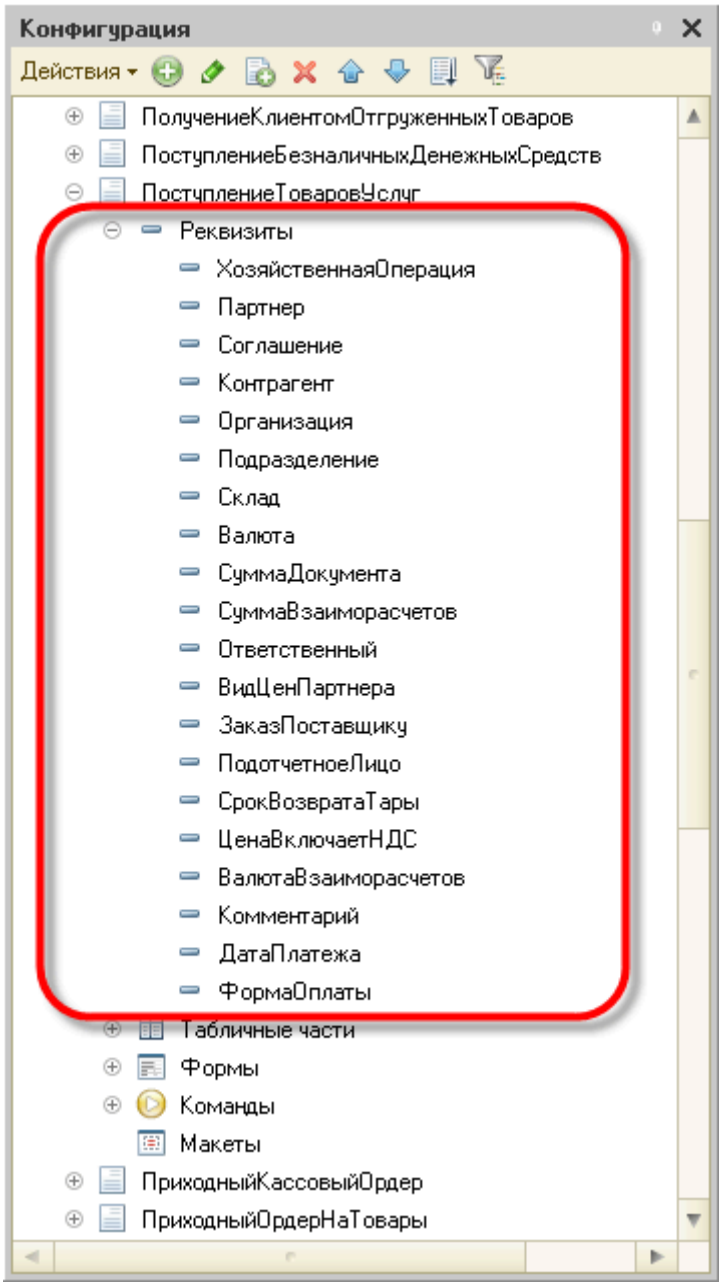
Каждый документ характеризуется номером, датой и временем. Система поддерживает режим автоматической нумерации документов, при котором она самостоятельно может генерировать номер для нового документа. Кроме этого система позволяет осуществлять контроль уникальности номеров документов, не разрешая создавать документы с одинаковыми номерами:



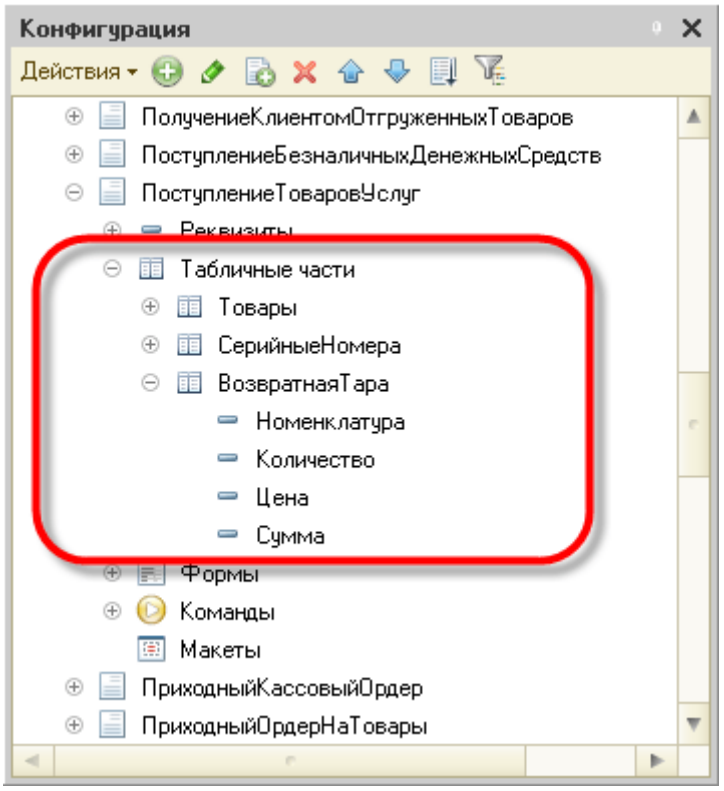
Система автоматически поддерживает режим, при котором уникальность номеров и автоматическая нумерация могут выполняться в пределах определенного периода (день, месяц, квартал, год). Например, если периодичность установлена год, то с нового года система опять начнет нумерацию указанных документов с 1.

Важными характеристиками документа являются дата и время. Они позволяют установить строгую временную последовательность совершения операций. Таким образом, документы могут отличаться друг от друга не только номером, но и своим положением на временной оси. В результате всегда можно сказать, какая из двух операций была совершена раньше.

Помимо номера, даты и времени, каждый документ, как правило, содержит некоторую дополнительную информацию, которая подробно описывает этот документ. Например, для документа **Поступление товаров и услуг** это может быть информация о поставщике товаров (контрагенте), складе, на который приходуются товары и т. п. Набор такой информации является одинаковым для всех документов конкретного вида, и для ее хранения служат реквизиты документа:



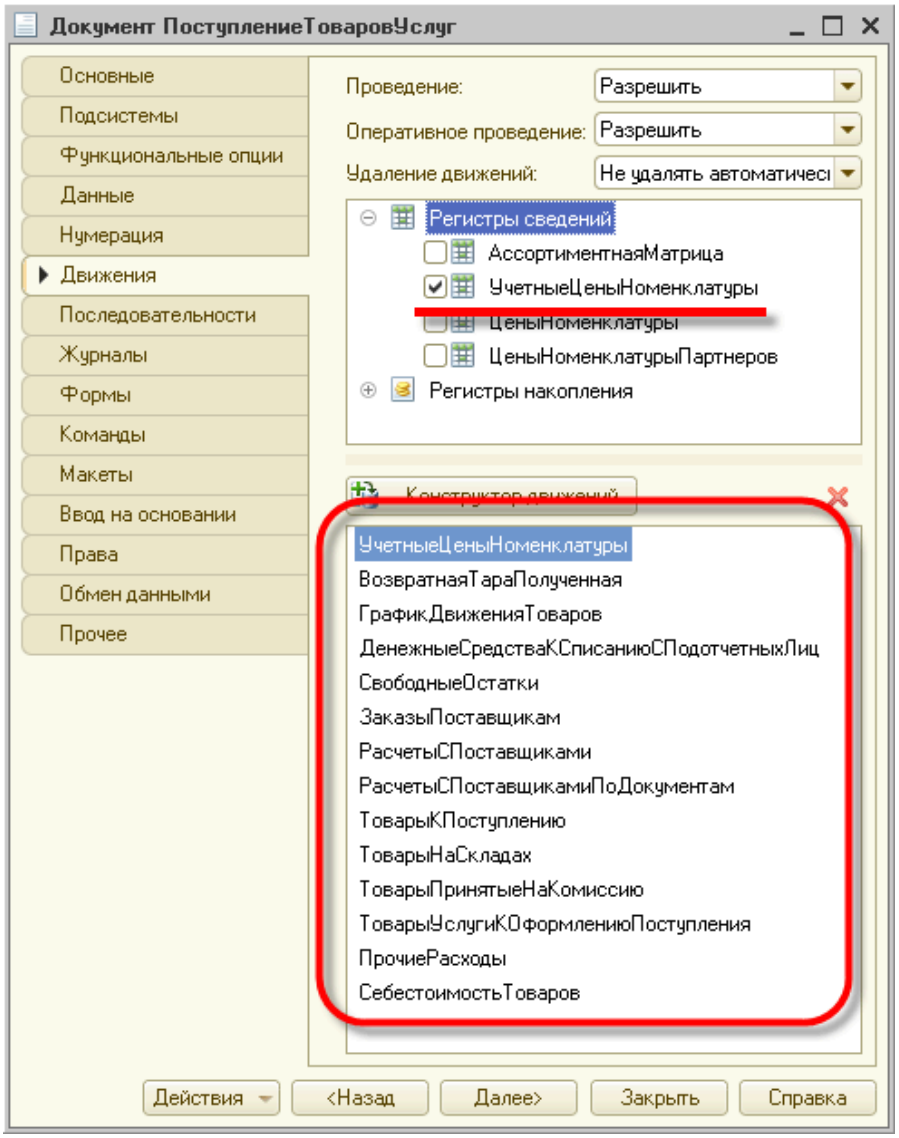
Кроме этого, каждый документ может содержать некоторый набор информации, которая одинакова по своей структуре, но различна по количеству, для разных документов. Например, для документа **Поступление товаров и услуг** это может быть информация о товарах, поступивших на предприятие (наименование, количество, и т. д.), серийных номерах и возвратной таре. Для хранения подобных данных служат табличные части документа:



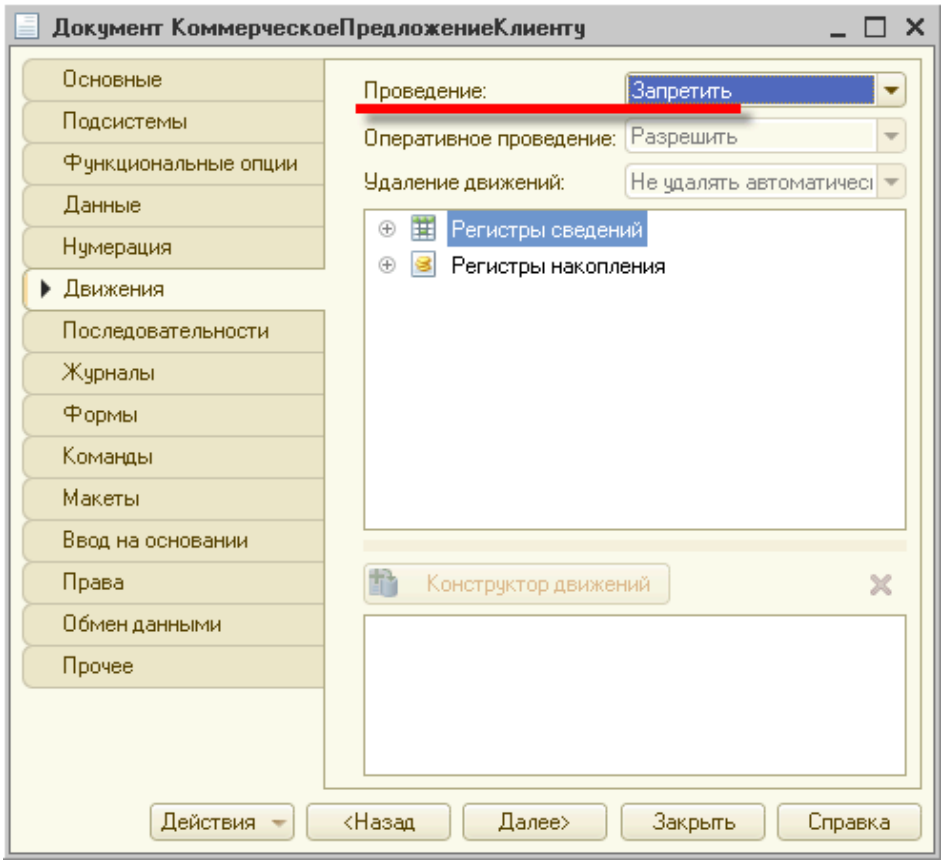
Проведение документа

Важным свойством документа является возможность его проведения. Если документ проводится, то он может изменить состояние тех или иных учитываемых данных. Если же документ не является «проводимым» это значит, что событие, которое он отражает, не влияет на состояние учета, который ведется в данном прикладном решении.

Например, документ **Поступление товаров и услуг** при своем проведении может вносить изменения в состояние расчетов с поставщиками, в учет остатков товаров, в состояние заказов покупателей и другие учетные данные:



Для документа **Коммерческое предложение контрагенту** напротив, проведение может быть запрещено, потому что этот документ никак не влияет на состояние учета:

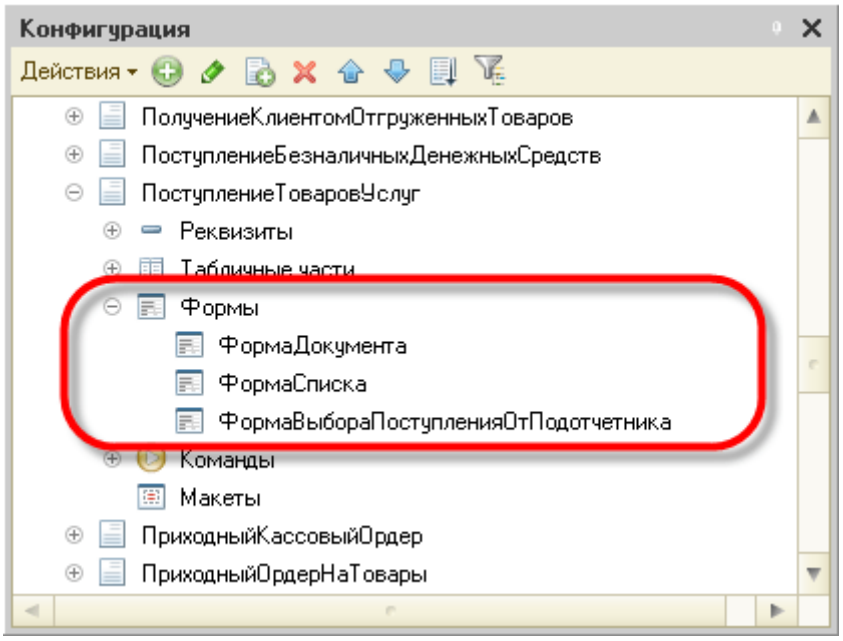


Конструктор движений

Алгоритм, на основании которого документ вносит те или иные изменения в состояние учетных данных при своем проведении, описывается средствами встроенного языка на этапе разработки прикладного решения. Система содержит конструктор движений, который помогает разработчику создавать алгоритмы проведения документа. Подробнее...

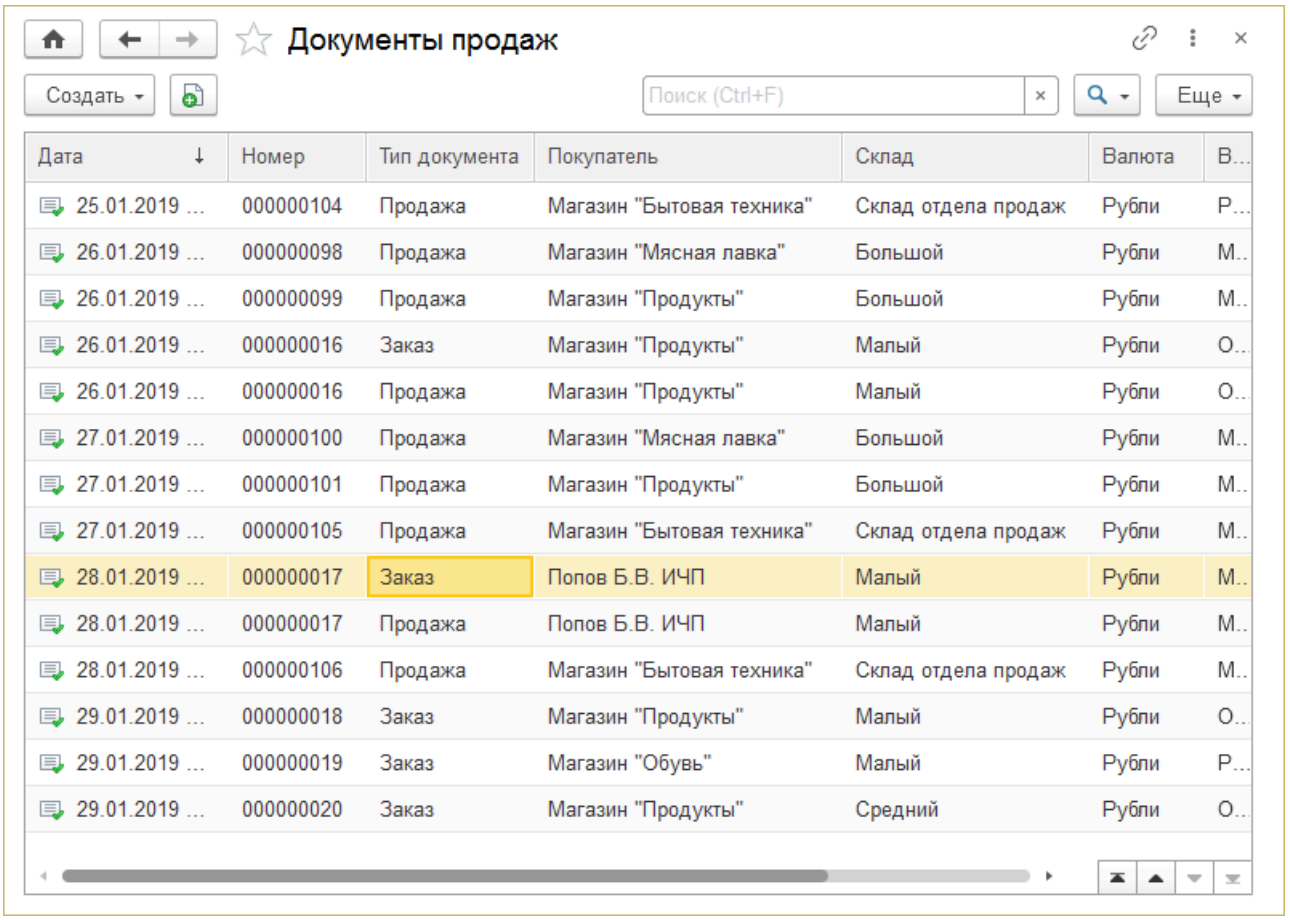
Формы документа

Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в документах, система поддерживает несколько форм представления документа. Система может автоматически генерировать все нужные формы документа. Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию:



Форма списка

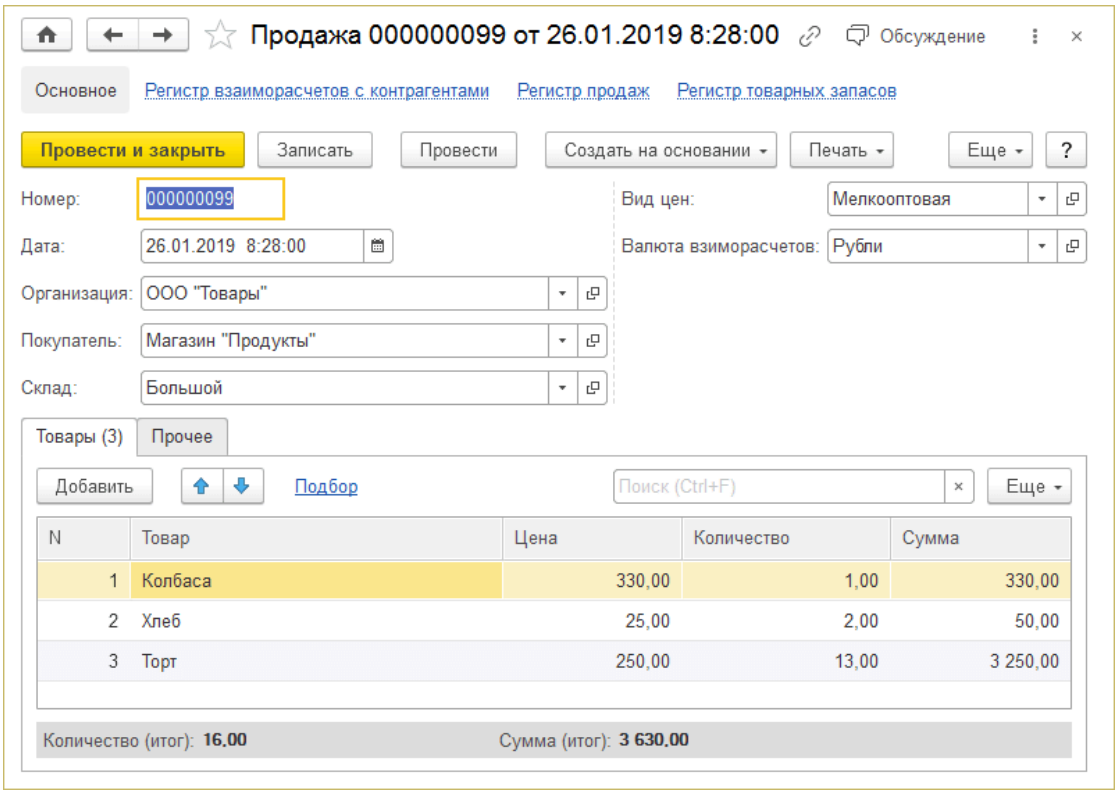
Для просмотра документов одного вида используется форма списка. Она позволяет выполнять навигацию по списку документов, задавать временной интервал отображаемых документов, добавлять, помечать на удаление и удалять документы. Форма списка позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



Дата	Номер	Тип документа	Покупатель	Склад	Валюта	В...
25.01.2019 ...	000000104	Продажа	Магазин "Бытовая техника"	Склад отдела продаж	Рубли	Р...
26.01.2019 ...	000000098	Продажа	Магазин "Мясная лавка"	Большой	Рубли	М...
26.01.2019 ...	000000099	Продажа	Магазин "Продукты"	Большой	Рубли	М...
26.01.2019 ...	000000016	Заказ	Магазин "Продукты"	Малый	Рубли	О...
26.01.2019 ...	000000016	Продажа	Магазин "Продукты"	Малый	Рубли	О...
27.01.2019 ...	000000100	Продажа	Магазин "Мясная лавка"	Большой	Рубли	М...
27.01.2019 ...	000000101	Продажа	Магазин "Продукты"	Большой	Рубли	М...
27.01.2019 ...	000000105	Продажа	Магазин "Бытовая техника"	Склад отдела продаж	Рубли	М...
28.01.2019 ...	000000017	Заказ	Попов Б.В. ИЧП	Малый	Рубли	М...
28.01.2019 ...	000000017	Продажа	Попов Б.В. ИЧП	Малый	Рубли	М...
28.01.2019 ...	000000106	Продажа	Магазин "Бытовая техника"	Склад отдела продаж	Рубли	М...
29.01.2019 ...	000000018	Заказ	Магазин "Продукты"	Малый	Рубли	О...
29.01.2019 ...	000000019	Заказ	Магазин "Обувь"	Малый	Рубли	Р...
29.01.2019 ...	000000020	Заказ	Магазин "Продукты"	Средний	Рубли	О...

Форма документа

Для просмотра и изменения данных отдельных документов используется форма документа. Как правило, она представляет данные в удобном для восприятия и редактирования виде:

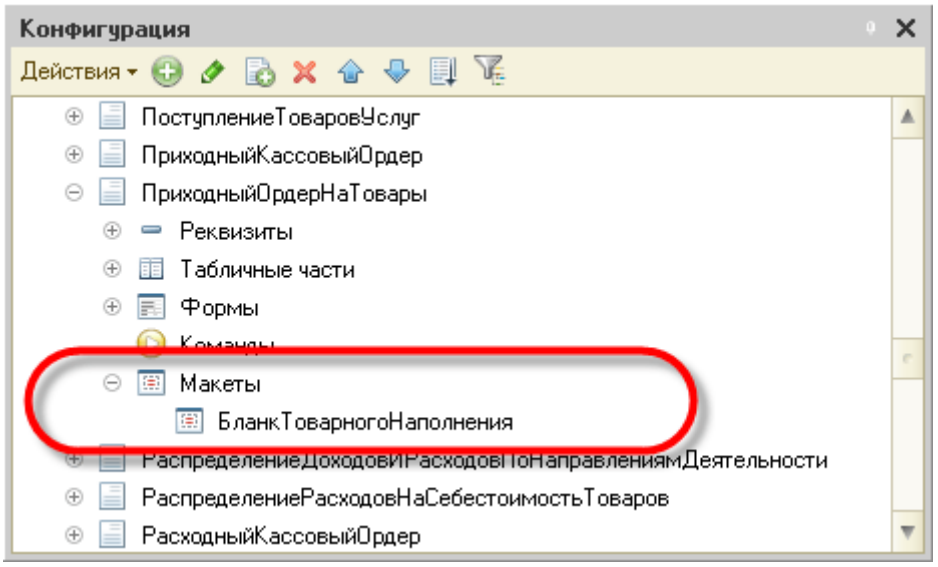


Форма выбора

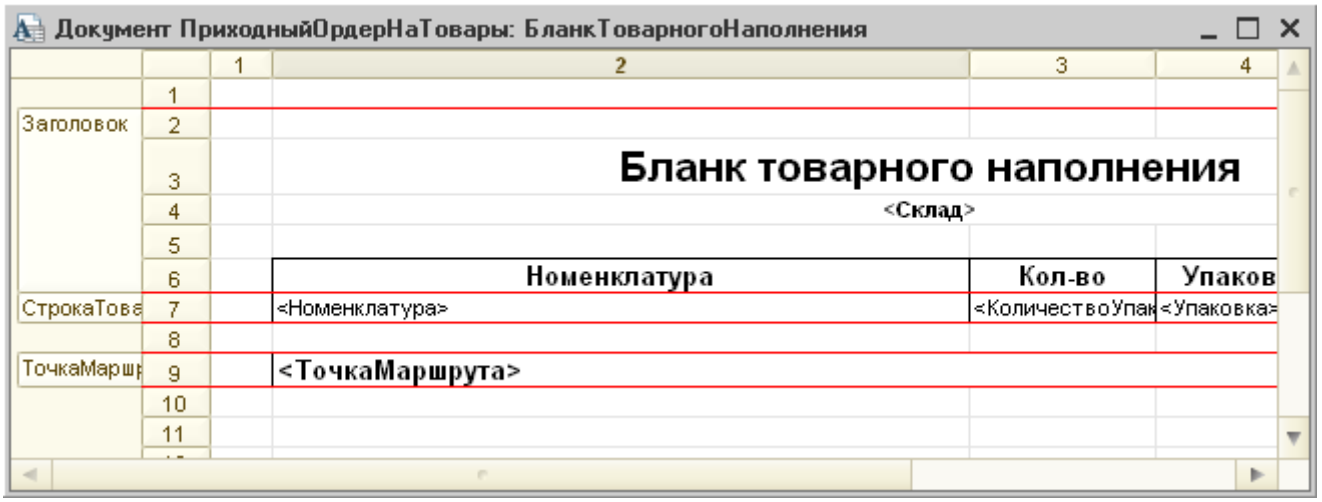
Кроме этих двух форм для документа поддерживается форма выбора конкретных документов из списка. Она, обычно, содержит минимальный набор информации, необходимой для выбора того или иного документа и может использоваться, например, при создании документа **Поступление товаров и услуг**, в котором должен быть указан заказ поставщику, на основании которого формируется поступление товаров.

Макеты документа

Документу могут быть сопоставлены несколько макетов, содержащих данные, необходимые для обеспечения работы документа:

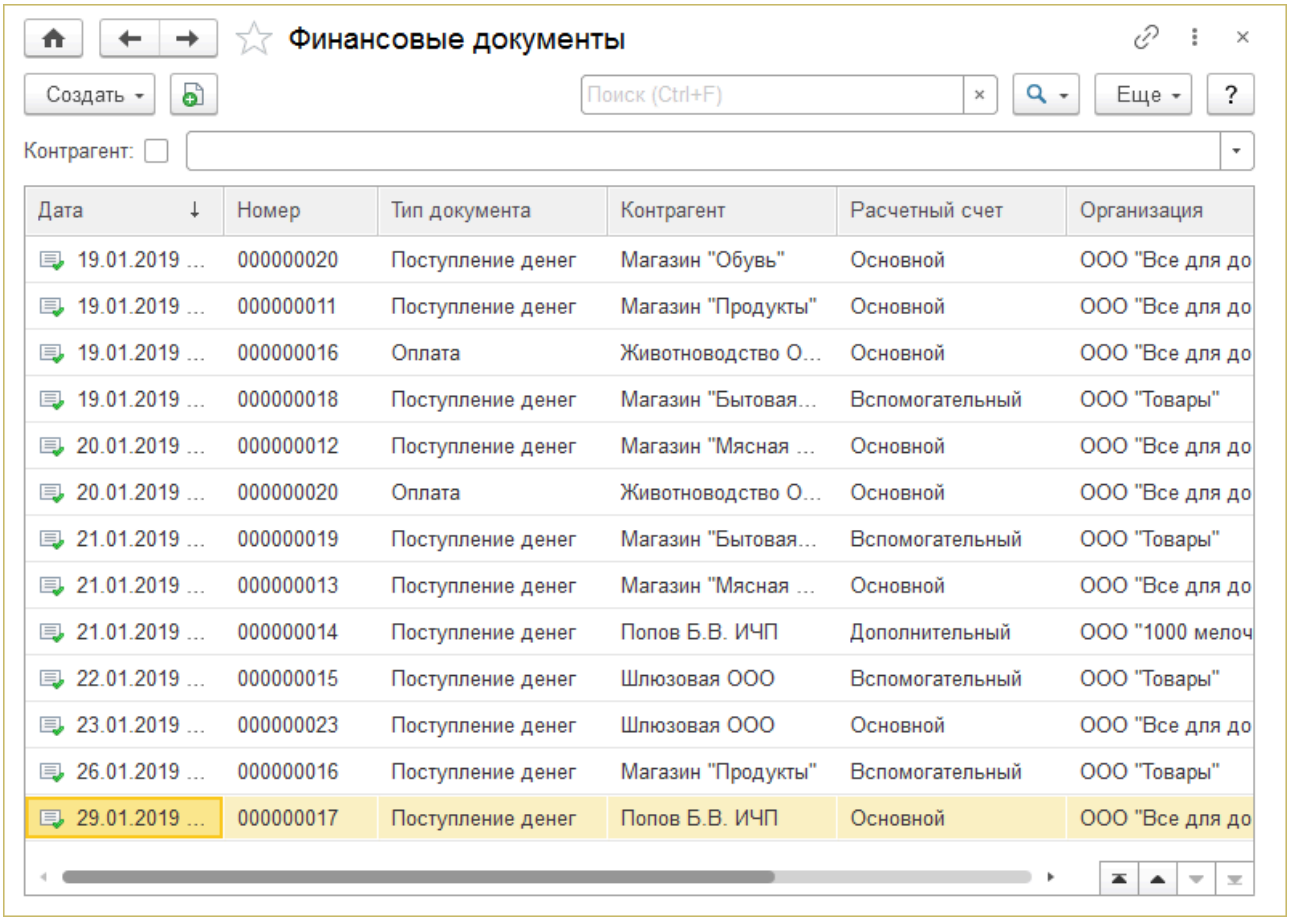


Макеты могут использоваться для формирования печатных форм документа или для отображения дополнительной информации, имеющей отношение к документу:



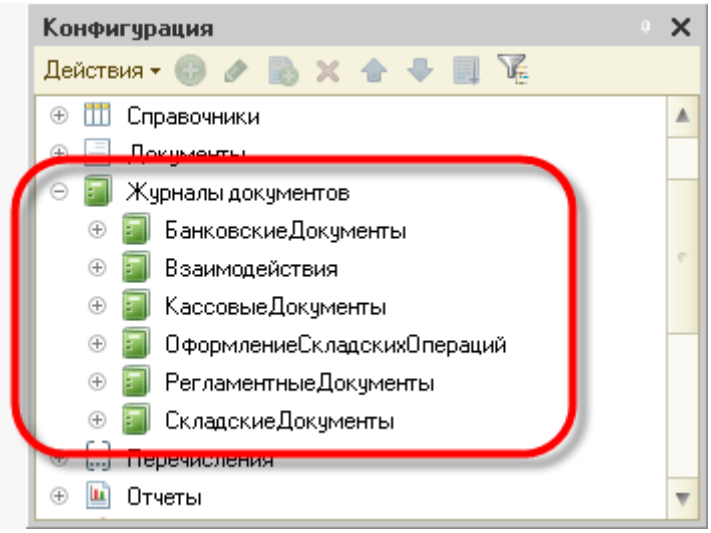
Журнал документов

Журналы документов — это прикладные объекты конфигурации. Они предназначены для просмотра документов разных видов. Для журнала документов могут быть определены графы, предназначенные для отображения реквизитов документов разного вида, отнесенных к данному журналу:

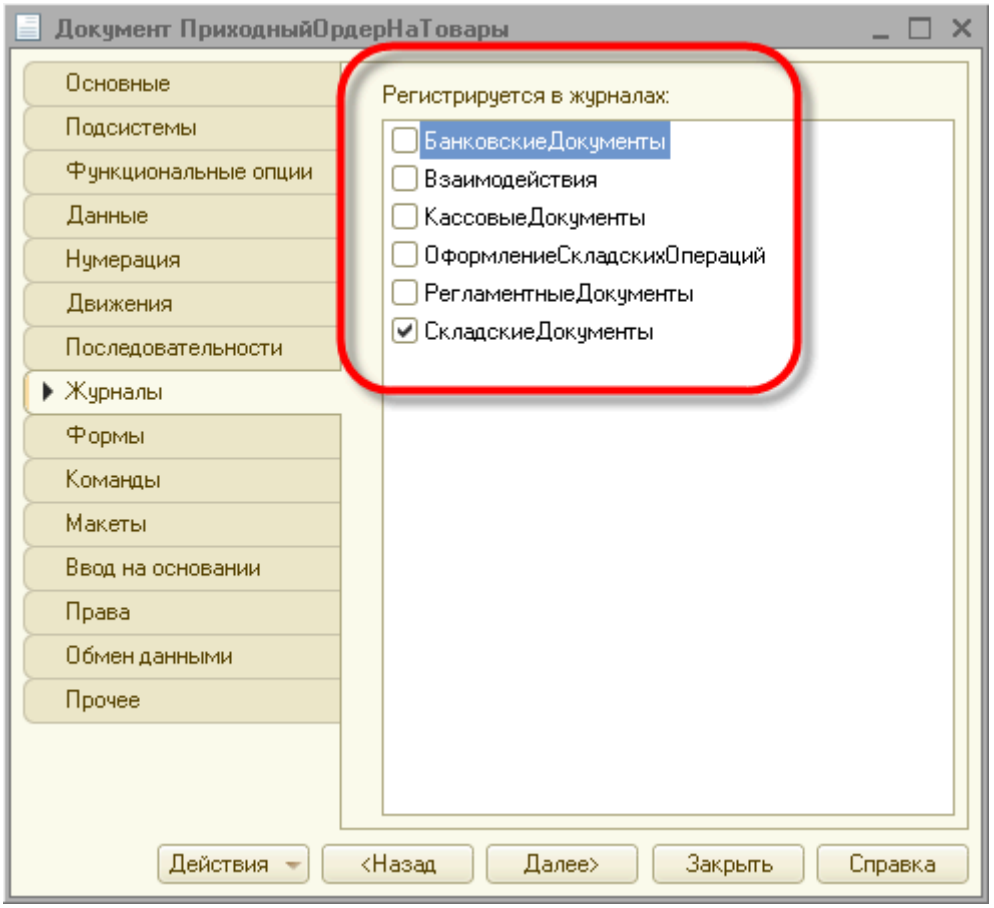


Дата	Номер	Тип документа	Контрагент	Расчетный счет	Организация
19.01.2019 ...	000000020	Поступление денег	Магазин "Обувь"	Основной	ООО "Все для до
19.01.2019 ...	000000011	Поступление денег	Магазин "Продукты"	Основной	ООО "Все для до
19.01.2019 ...	000000016	Оплата	Животноводство О...	Основной	ООО "Все для до
19.01.2019 ...	000000018	Поступление денег	Магазин "Бытовая...	Вспомогательный	ООО "Товары"
20.01.2019 ...	000000012	Поступление денег	Магазин "Мясная ...	Основной	ООО "Все для до
20.01.2019 ...	000000020	Оплата	Животноводство О...	Основной	ООО "Все для до
21.01.2019 ...	000000019	Поступление денег	Магазин "Бытовая...	Вспомогательный	ООО "Товары"
21.01.2019 ...	000000013	Поступление денег	Магазин "Мясная ...	Основной	ООО "Все для до
21.01.2019 ...	000000014	Поступление денег	Попов Б.В. ИЧП	Дополнительный	ООО "1000 мелоч
22.01.2019 ...	000000015	Поступление денег	Шлюзовая ООО	Вспомогательный	ООО "Товары"
23.01.2019 ...	000000023	Поступление денег	Шлюзовая ООО	Основной	ООО "Все для до
26.01.2019 ...	000000016	Поступление денег	Магазин "Продукты"	Вспомогательный	ООО "Товары"
29.01.2019 ...	000000017	Поступление денег	Попов Б.В. ИЧП	Основной	ООО "Все для до

В прикладном решении может быть создано произвольное количество журналов документов:



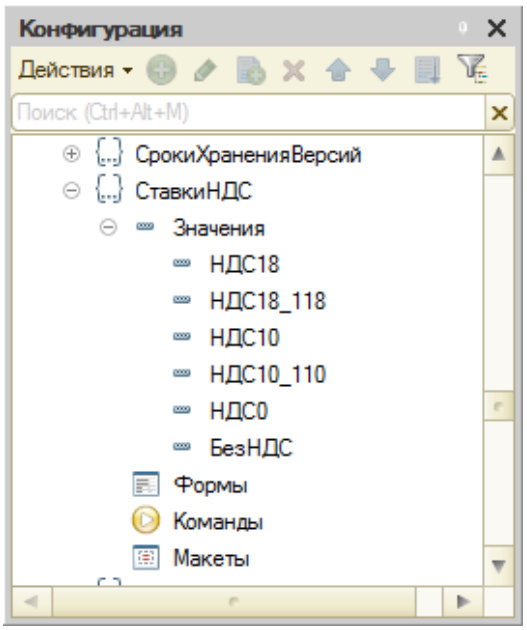
Каждый вид документа может быть показан в нескольких журналах:



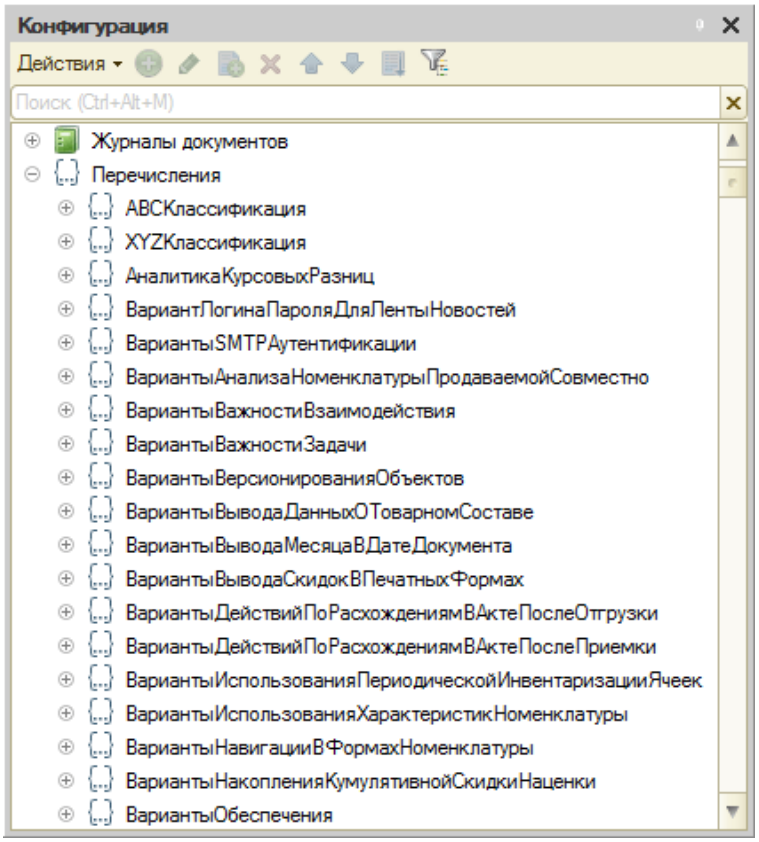
Перечисления

Перечисления — это прикладные объекты конфигурации. Они позволяют хранить в информационной базе наборы значений, которые не изменяются в процессе работы прикладного решения.

Например, это может быть перечисление возможных ставок НДС (БезНДС, НДС18 и т. д.), перечисление состояния заказов (Запланировано, ВРаботе, Выполнено) и пр.:



В прикладном решении может быть создано произвольное количество перечислений:



Отчет

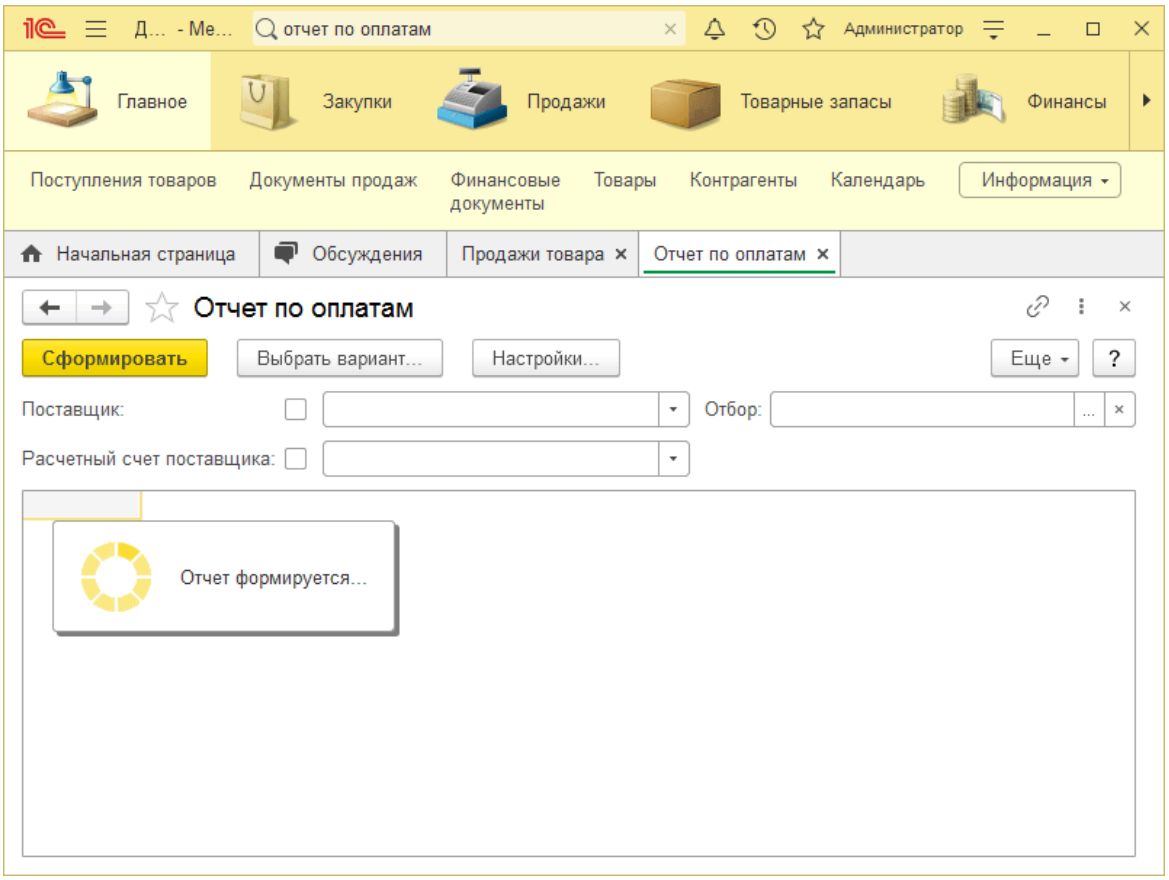
Отчеты — это прикладные объекты конфигурации. Они предназначены для обработки накопленной информации и получения сводных данных в удобном для просмотра и анализа виде. Конфигуратор позволяет формировать набор различных отчетов, достаточных для удовлетворения потребности пользователей системы в достоверной и подробной выходной информации.

Как правило, для формирования выходных данных отчет использует систему компоновки данных. Но, вообще говоря, отчет может содержать произвольный алгоритм формирования «бумажного» или «электронного» отчета на встроенном языке.

Отчет может содержать одну или несколько форм, с помощью которых, при необходимости, можно организовать ввод каких-либо параметров, влияющих на ход алгоритма.

Выполнение отчетов в фоновом режиме

Отчеты, использующие систему компоновки данных, могут формироваться в фоновом режиме. Запустив отчет пользователь может продолжать свою работу с программой, а на время формирования отчета вместо данных будет отображаться специальная картинка.


Д... - Ме...
отчет по оплатам
Администратор
Главное
Закупки
Продажи
Товарные запасы
Финансы
Поступления товаров
Документы продаж
Финансовые документы
Товары
Контрагенты
Календарь
Информация
Начальная страница
Обсуждения
Продажи товара
Отчет по оплатам
Отчет по оплатам
Сформировать
Выбрать вариант...
Настройки...
Еще
?
Поставщик:
Отбор:
Расчетный счет поставщика:
Отчет формируется...

Обработка

Обработки — это прикладные объекты конфигурации. Они предназначены для выполнения различных действий над информацией.

Например, с их помощью можно выполнять удаление из системы устаревших данных, импорт информации из других систем и многое другое. Характер выполняемых в этом случае действий отражает название объекта конфигурации — **Обработка**, так как в результате информация, хранящаяся в системе, претерпевает какие-либо изменения.

Обработка может содержать одну или несколько форм, с помощью которых, при необходимости, можно организовать ввод каких-либо параметров, влияющих на ход алгоритма. Вывод результатов выполнения алгоритма на экран и принтер осуществляется с помощью конструктора запроса с обработкой результата.

Основное отличие обработки от отчета заключается в том, что отчет может использовать схему компоновки данных. В остальном обработка не отличается от отчета.

План видов характеристик

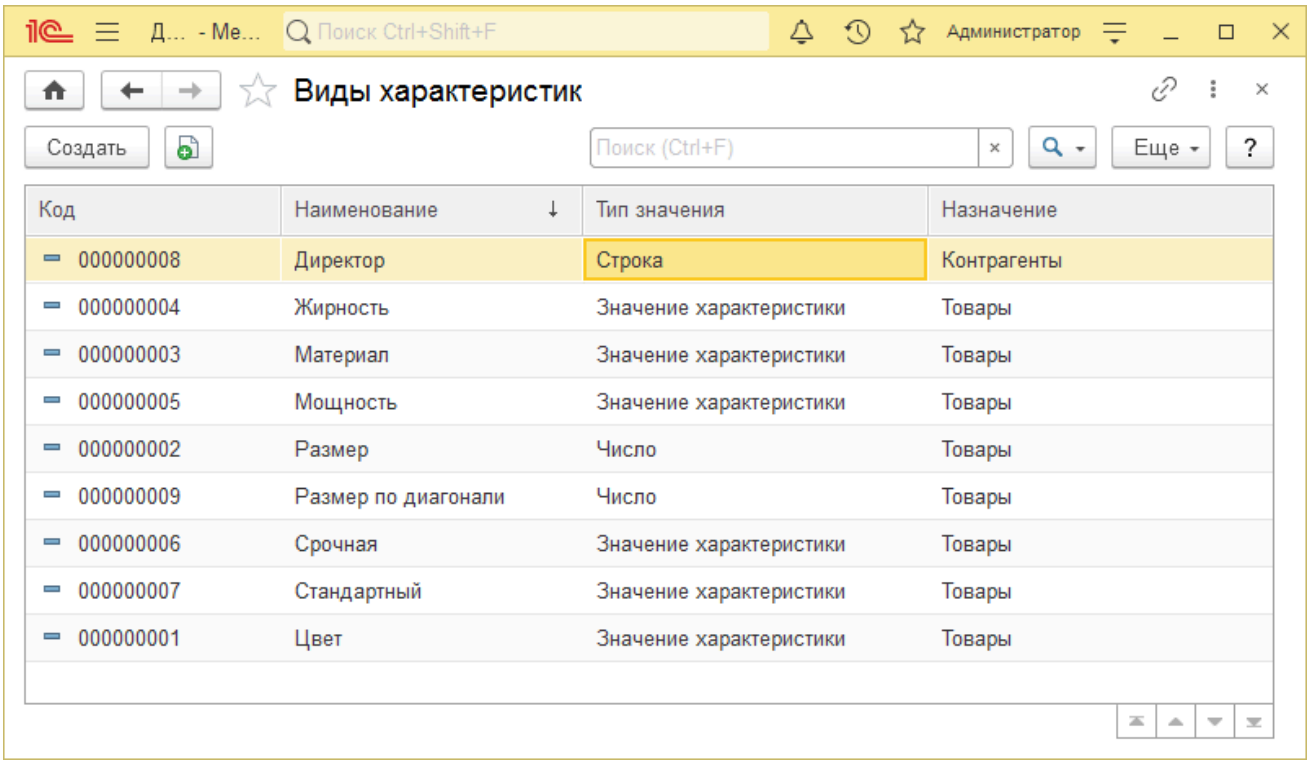
Планы видов характеристик — это прикладные объекты конфигурации. Они предназначены для хранения информации о характеристиках различных объектов.

С их помощью пользователь может создавать всевозможные характеристики, описывать тип этих характеристик и задавать их значения. Например, для того, чтобы описывать товары произвольным количеством произвольных характеристик (цвет, размер, запах и т. д.).

Структура

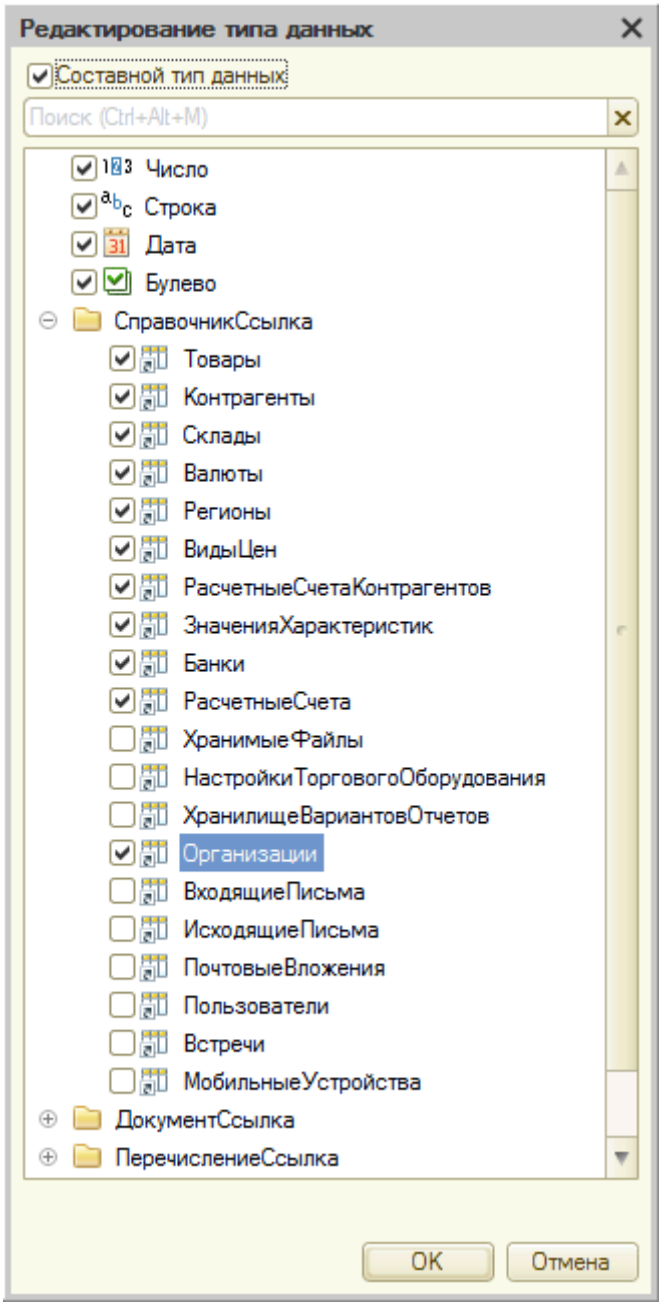
По своей структуре план видов характеристик напоминает справочник: в плане видов характеристик хранятся элементы, — характеристики, которые могут иметь некоторый набор реквизитов и табличных частей и, также как и элементы справочника, могут образовывать иерархические структуры. В плане видов характеристик могут существовать предопределенные характеристики, заданные разработчиком.

Однако основное назначение плана видов характеристик заключается в том, чтобы для каждой характеристики хранить тип значения, который она может принимать:



Код	Наименование	Тип значения	Назначение
000000008	Директор	Строка	Контрагенты
000000004	Жирность	Значение характеристики	Товары
000000003	Материал	Значение характеристики	Товары
000000005	Мощность	Значение характеристики	Товары
000000002	Размер	Число	Товары
000000009	Размер по диагонали	Число	Товары
000000006	Срочная	Значение характеристики	Товары
000000007	Стандартный	Значение характеристики	Товары
000000001	Цвет	Значение характеристики	Товары

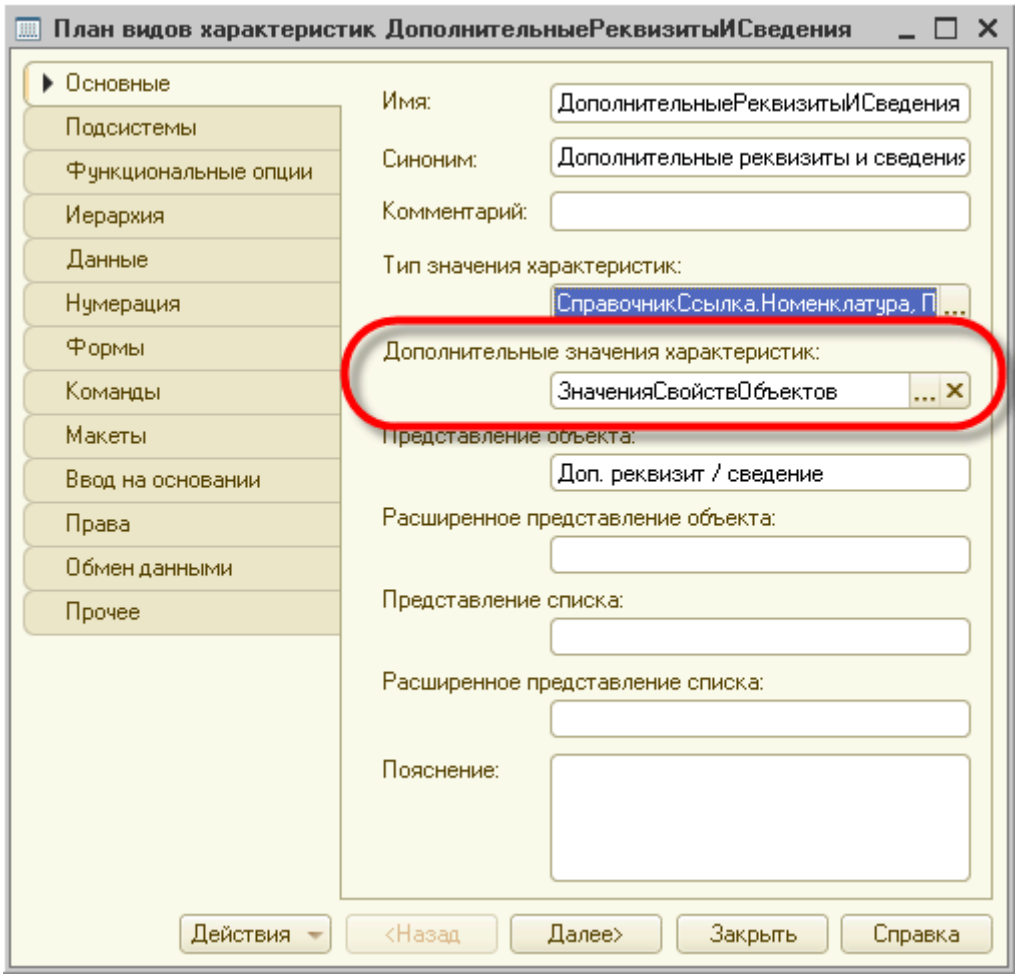
Возможный перечень типов, которые могут принимать характеристики, указывается разработчиком в процессе создания прикладного решения:



Создавая новые характеристики (или редактируя существующие), пользователь сможет выбрать для них один из типов, входящих в этот перечень.

Однако не исключена ситуация, когда для создания очередной характеристики пользователю понадобится тип, не существующий в прикладном решении. Например, пользователь решит создать характеристику **Запах**, которая должна иметь значения справочника **Запахи**, но такого справочника в прикладном решении нет.

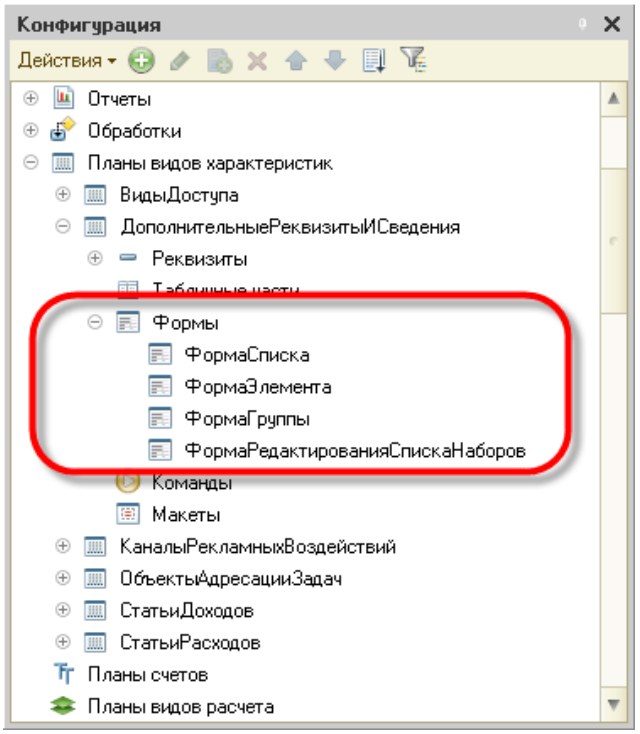
Специально для таких ситуаций разработчик может создать специальный пустой справочник, и указать, что в нем будут храниться дополнительные значения характеристик:



Теперь, создавая характеристику **Запах**, пользователь сможет выбрать для нее тип значения этого специального справочника, а в самом справочнике создать нужные ему значения: **Сладкий**, **Резкий**, **Кислый** и т. д.

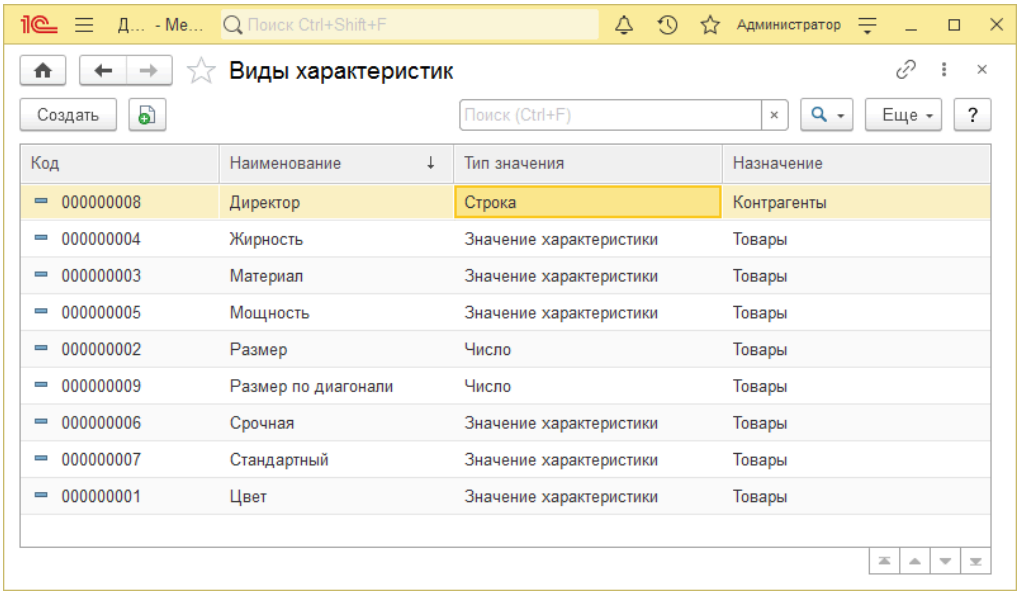
Формы плана видов характеристик

Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в плане видов характеристик, система поддерживает несколько форм его представления. Система может автоматически генерировать все нужные формы плана видов характеристик. Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию:



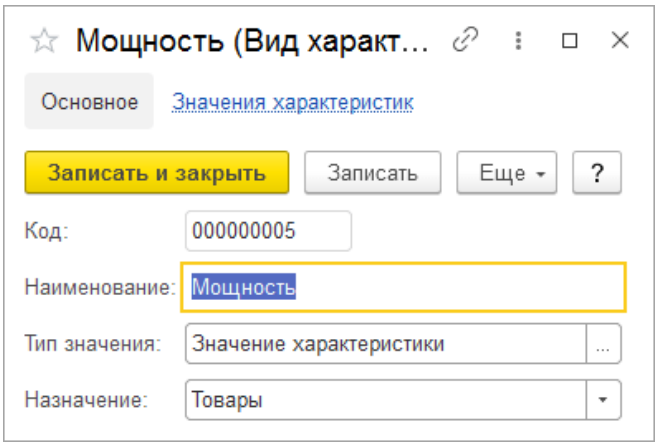
Форма списка

Для просмотра данных, содержащихся в плане видов характеристик, используется форма списка. Она позволяет выполнять навигацию по плану, добавлять, помечать на удаление и удалять характеристики и группы характеристик, перемещать характеристики и группы. Форма списка может представлять данные в иерархическом и не иерархическом виде и позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



Форма характеристики

Для просмотра и изменения данных отдельных характеристик используется форма характеристики. Как правило, она представляет данные в удобном для восприятия и редактирования виде:



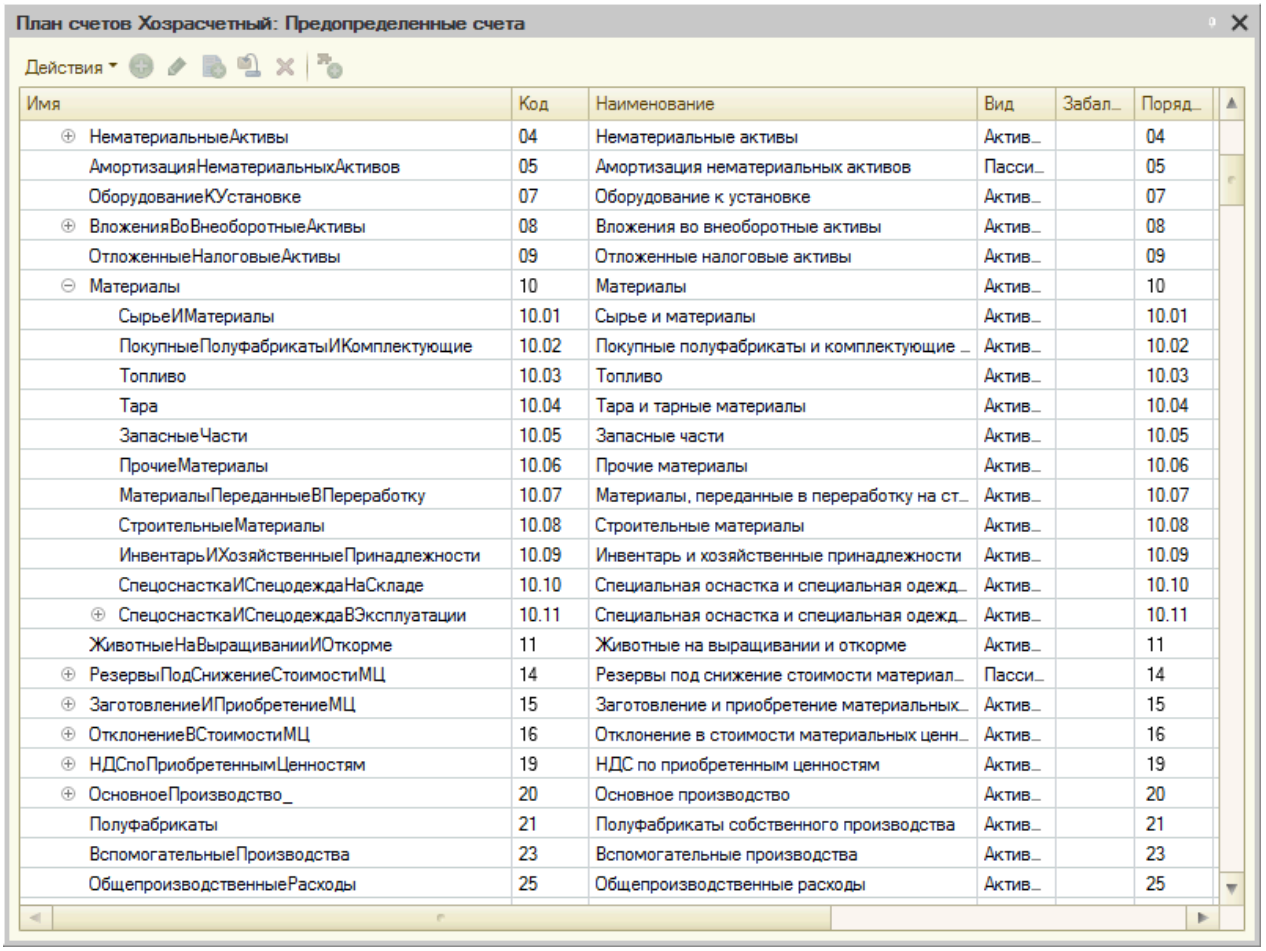
Кроме этого для планов видов характеристик, также как и для справочников, поддерживаются формы группы, выбора и выбора группы.

План счетов

Планы счетов — это прикладные объекты конфигурации. Каждый из них позволяет описать совокупность синтетических счетов, предназначенных для группировки информации о хозяйственной деятельности предприятия. Путем настройки плана счетов организуется, собственно, требуемая система учета.

Структура

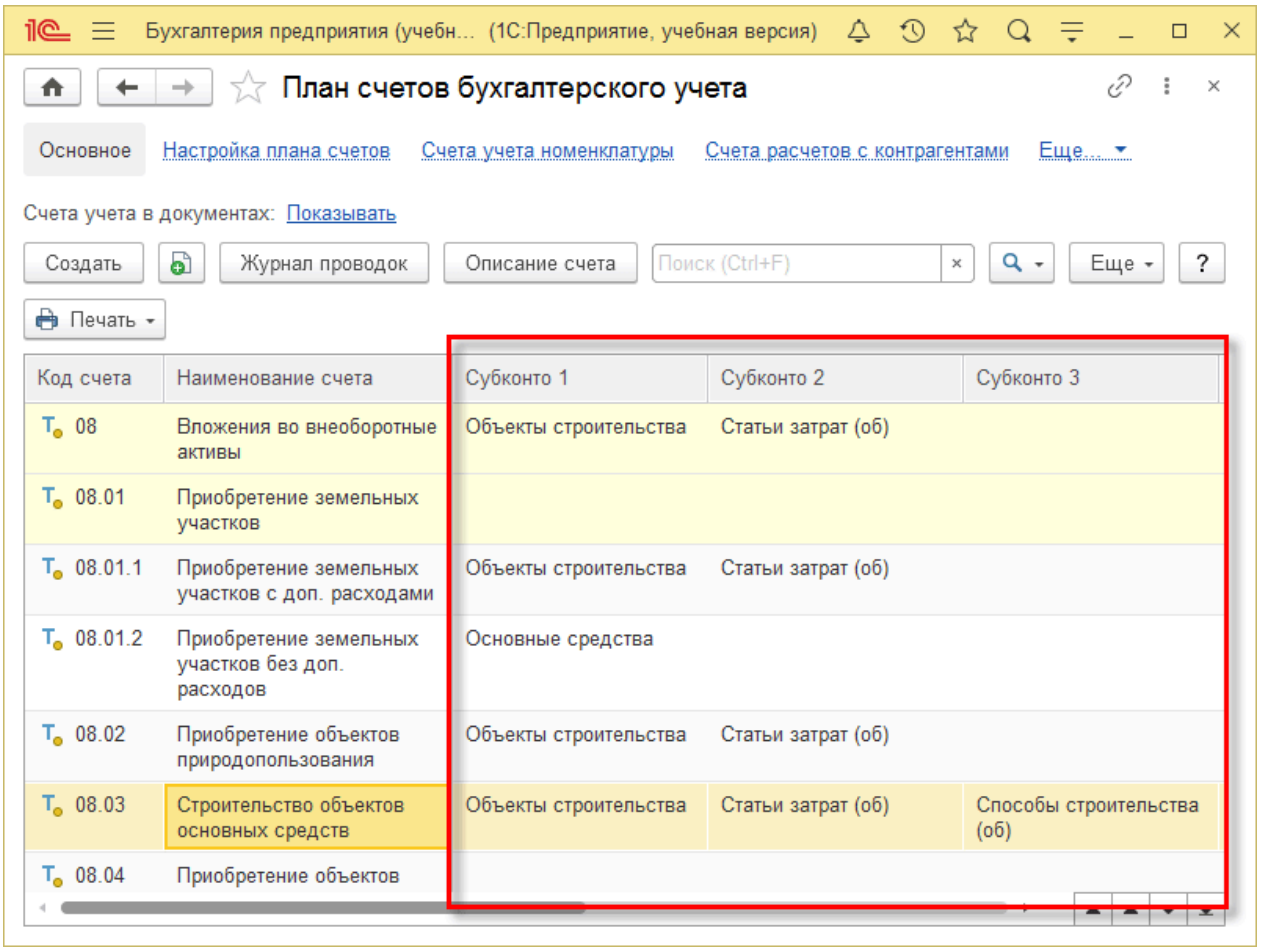
Планы счетов поддерживают многоуровневую иерархию «счет — субсчета». Каждый план счетов может включать необходимое количество счетов первого уровня. К каждому счету может быть открыто нужное количество субсчетов. В свою очередь, каждый субсчет может иметь свои субсчета — и т. д. Количество уровней вложенности субсчетов в системе 1С:Предприятие 8 неограничено:



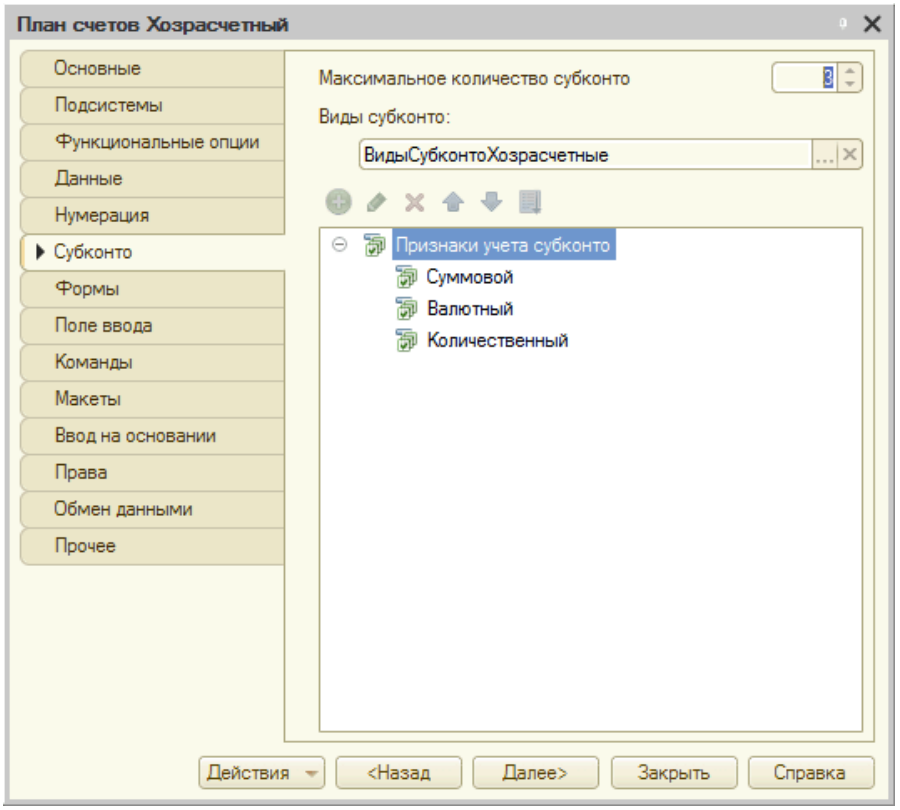
Имя	Код	Наименование	Вид	Забал...	Поряд...
НематериальныеАктивы	04	Нематериальные активы	Актив...		04
АмортизацияНематериальныхАктивов	05	Амортизация нематериальных активов	Пасси...		05
ОборудованиеКУстановке	07	Оборудование к установке	Актив...		07
ВложенияВоВнеоборотныеАктивы	08	Вложения во внеоборотные активы	Актив...		08
ОтложенныеНалоговыеАктивы	09	Отложенные налоговые активы	Актив...		09
Материалы	10	Материалы	Актив...		10
СырьеИМатериалы	10.01	Сырье и материалы	Актив...		10.01
ПокупныеПолуфабрикатыИКомплектующие	10.02	Покупные полуфабрикаты и комплектующие ...	Актив...		10.02
Топливо	10.03	Топливо	Актив...		10.03
Тара	10.04	Тара и тарные материалы	Актив...		10.04
ЗапасныеЧасти	10.05	Запасные части	Актив...		10.05
ПрочиеМатериалы	10.06	Прочие материалы	Актив...		10.06
МатериалыПереданныеВПереработку	10.07	Материалы, переданные в переработку на ст...	Актив...		10.07
СтроительныеМатериалы	10.08	Строительные материалы	Актив...		10.08
ИнвентарьИХозяйственныеПринадлежности	10.09	Инвентарь и хозяйственные принадлежности	Актив...		10.09
СпецоснасткаИСпецодеждаНаСкладе	10.10	Специальная оснастка и специальная одежд...	Актив...		10.10
СпецоснасткаИСпецодеждаВЭксплуатации	10.11	Специальная оснастка и специальная одежд...	Актив...		10.11
ЖивотныеНаВыращиванииИОткорме	11	Животные на выращивании и откорме	Актив...		11
РезервыПодСнижениеСтоимостиМЦ	14	Резервы под снижение стоимости материал...	Пасси...		14
ЗаготовлениеИПриобретениеМЦ	15	Заготовление и приобретение материальных...	Актив...		15
ОтклонениеВСтоимостиМЦ	16	Отклонение в стоимости материальных ценн...	Актив...		16
НДСпоПриобретеннымЦенностям	19	НДС по приобретенным ценностям	Актив...		19
ОсновноеПроизводство_	20	Основное производство	Актив...		20
Полуфабрикаты	21	Полуфабрикаты собственного производства	Актив...		21
ВспомогательныеПроизводства	23	Вспомогательные производства	Актив...		23
ОбщепроизводственныеРасходы	25	Общепроизводственные расходы	Актив...		25

Создание и редактирование счетов может выполняться как разработчиком (предопределенные счета), так и пользователем, в процессе работы с прикладным решением. Однако пользователь не может удалять счета, созданные разработчиком.

По любому счету или субсчету возможно ведение аналитического учета. При создании и редактировании плана счетов требуемому счету или субсчету можно поставить в соответствие необходимое количество субконто — объектов аналитического учета:

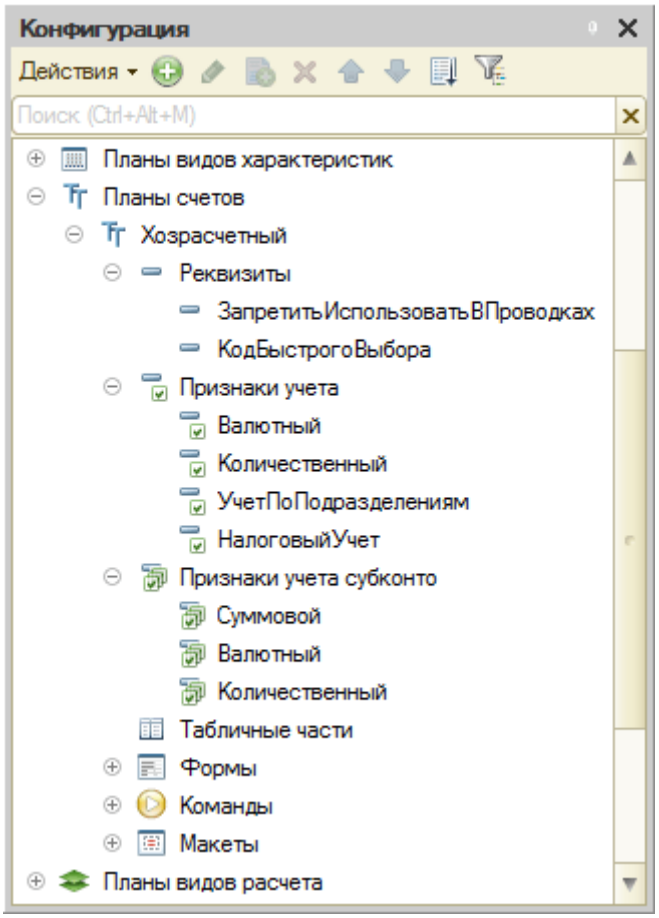


Для того чтобы иметь возможность указания субконто для счетов, план счетов связывается с планом видов характеристик, в котором описаны виды субконто, применяемые для этого плана счетов:



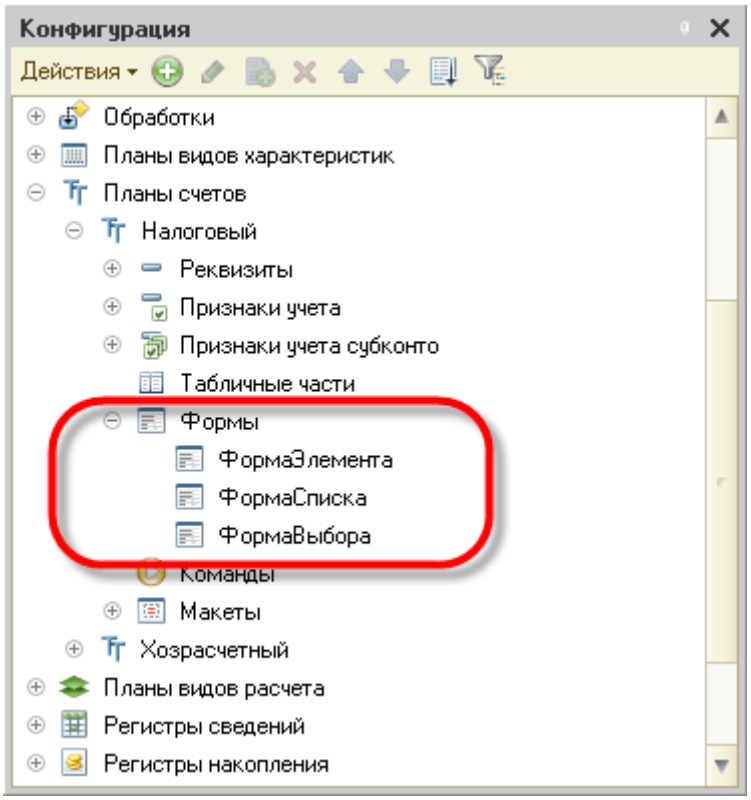
Для каждого счета можно задать несколько признаков учета, например количественный и валютный. Кроме этого можно задать несколько признаков учета субконто (например, суммовой, количественный, валютный). Признаки учета субконто позволяют установить признак использования того или иного вида субконто при формировании предопределенных счетов.

Также с каждым счетом и субсчетом может быть связан ряд дополнительной информации, которая хранится в реквизитах. Например, это может быть признак, запрещающий использовать счет в проводках:



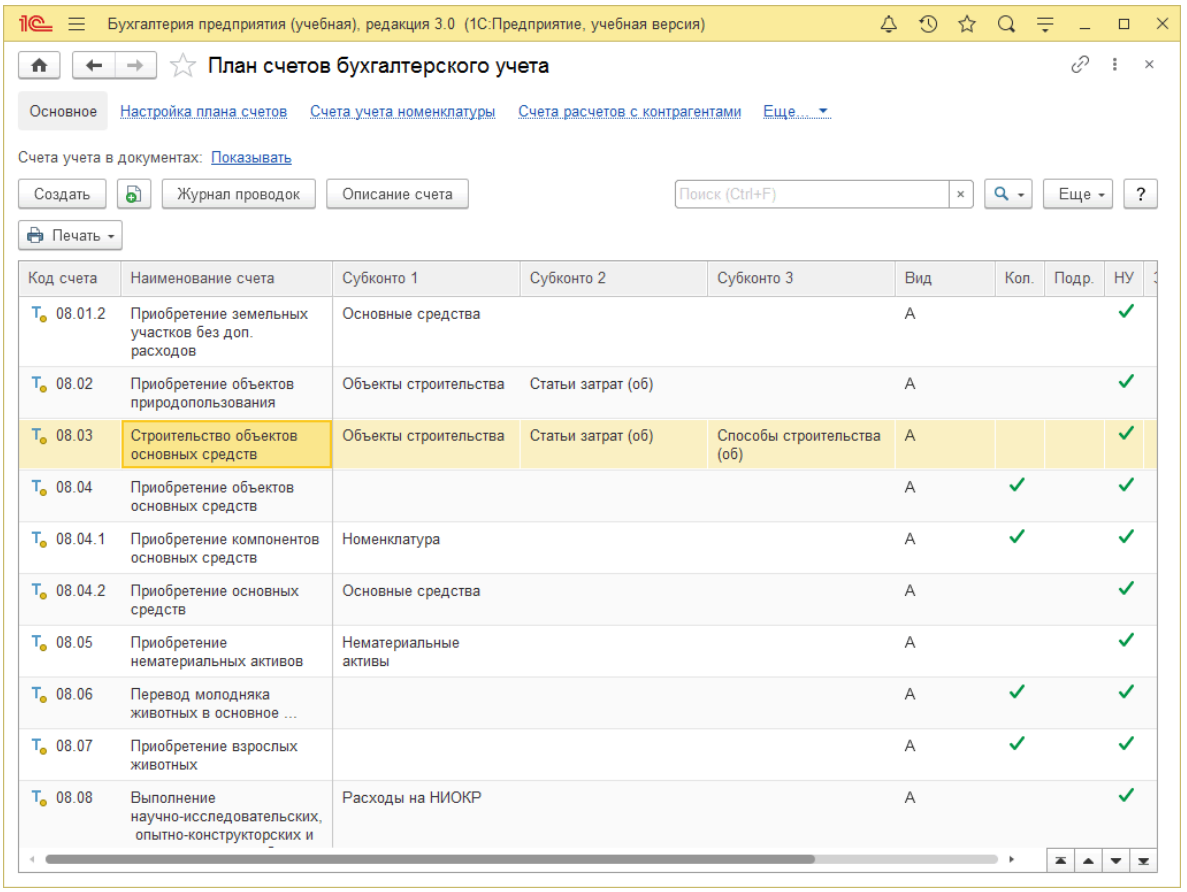
Формы плана счетов

Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в плане счетов, система поддерживает несколько форм представления плана счетов. Система может автоматически генерировать все нужные формы; наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию:



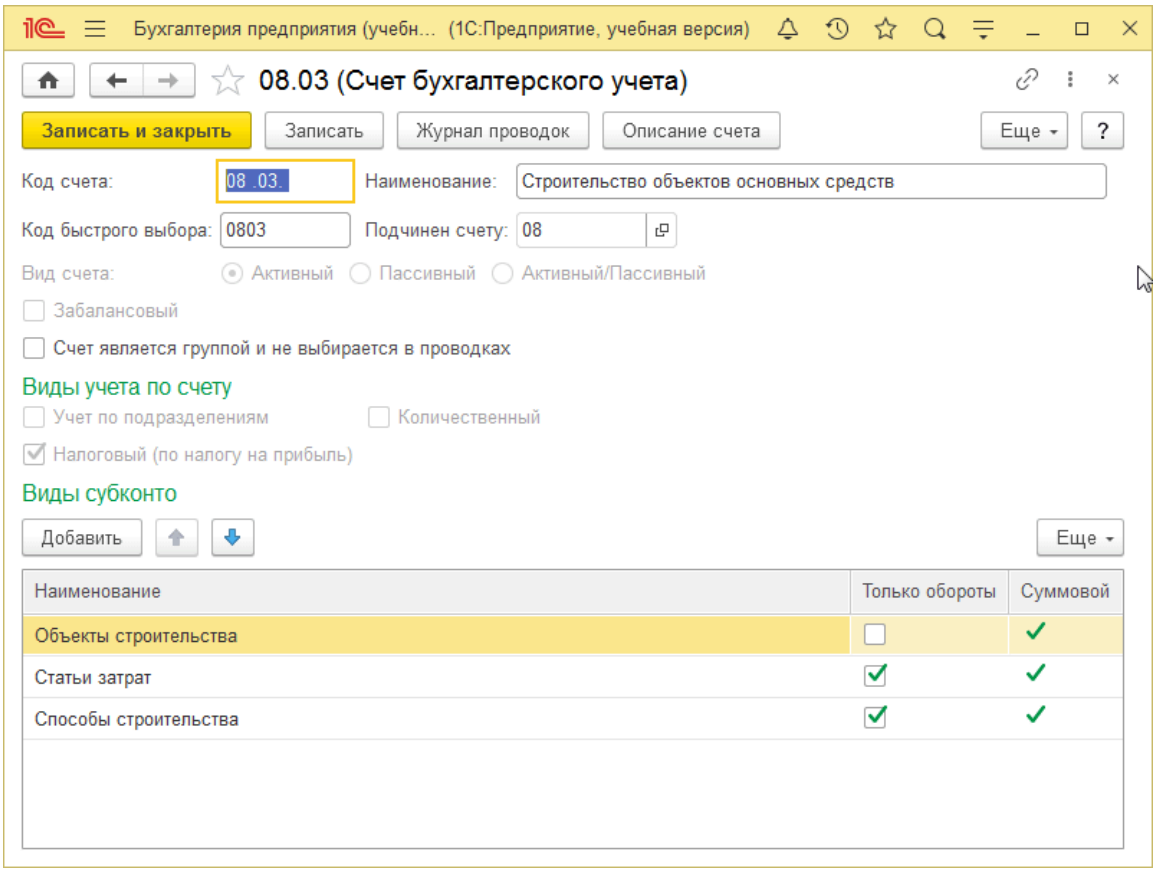
Форма списка

Для просмотра данных, содержащихся в плане счетов, используется форма списка. Она позволяет выполнять навигацию по плану, просматривать счета в иерархии и в виде простого списка, добавлять, помечать на удаление и удалять счета. Форма списка позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



Форма счета

Для просмотра и изменения отдельных счетов используется форма счета. Как правило, она представляет данные в удобном для восприятия и редактирования виде:



Форма выбора

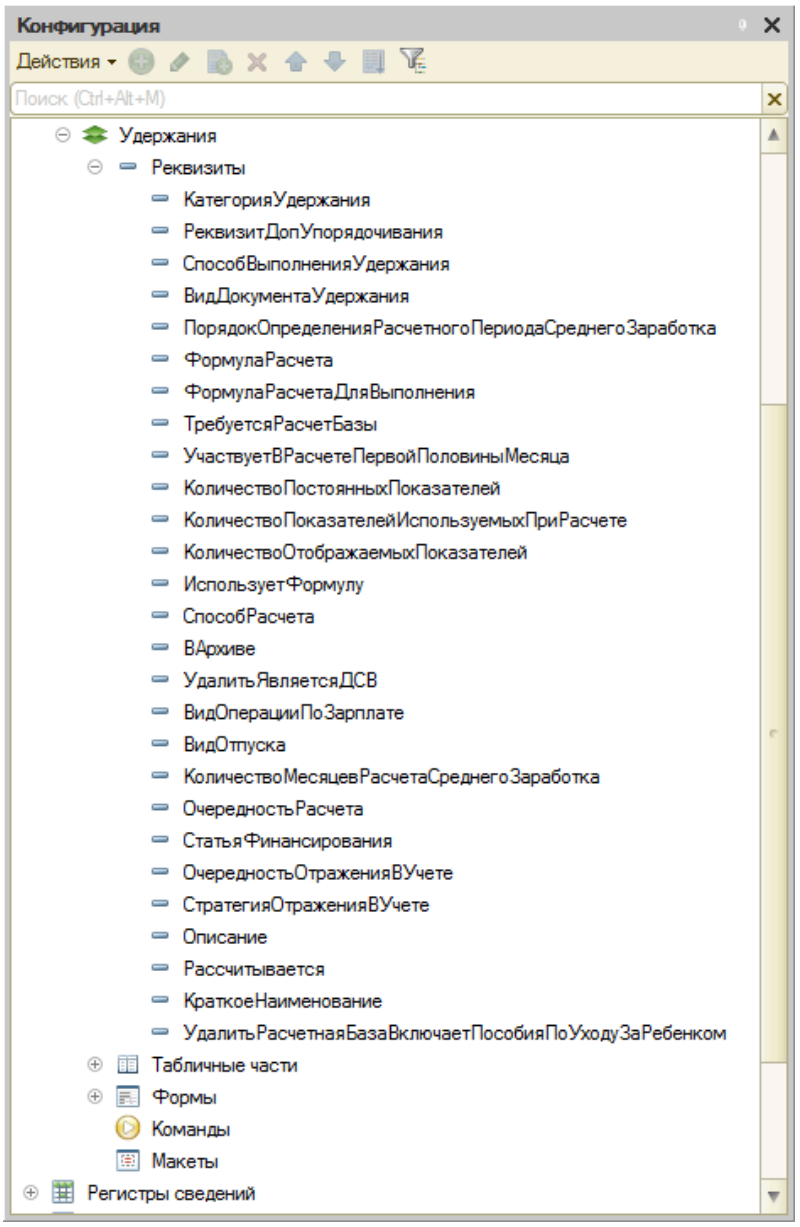
Кроме этих двух форм для плана счетов поддерживается форма выбора конкретного счета из списка. Она, обычно, содержит минимальный набор информации, необходимой для выбора того или иного счета и может использоваться, например, для выбора счета при создании бухгалтерской операции.

План видов расчета

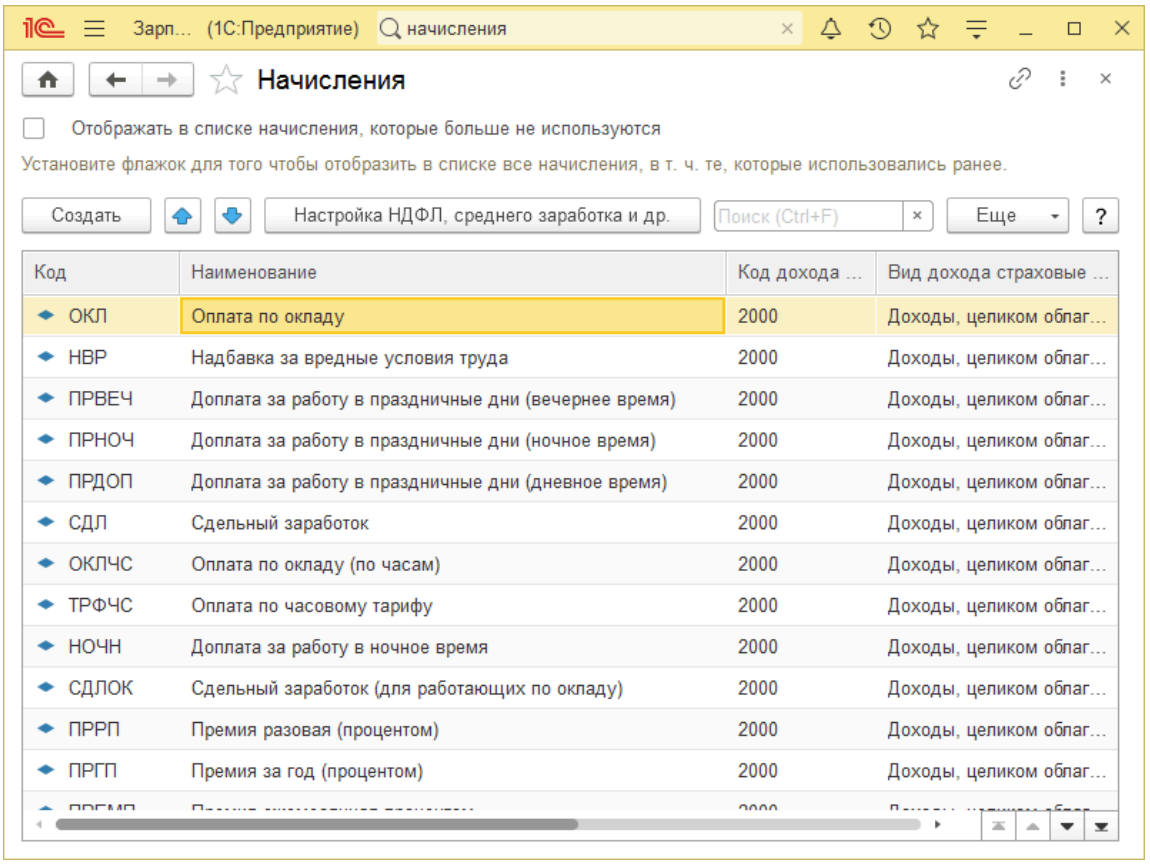
Планы видов расчета — это прикладные объекты конфигурации. Они используются в механизме сложных периодических расчетов и служат для описания видов расчета и их взаимного влияния друг на друга.

Структура

План видов расчета представляет собой список видов расчета. Каждый вид расчета имеет код, наименование и набор **реквизитов**, содержащих дополнительную информацию о данном виде расчета:



Например, план видов расчета **Начисления** может выглядеть следующим образом:



Код	Наименование	Код дохода ...	Вид дохода страховые ...
ОКЛ	Оплата по окладу	2000	Доходы, целиком облаг...
НВР	Надбавка за вредные условия труда	2000	Доходы, целиком облаг...
ПРВЕЧ	Доплата за работу в праздничные дни (вечернее время)	2000	Доходы, целиком облаг...
ПРНОЧ	Доплата за работу в праздничные дни (ночное время)	2000	Доходы, целиком облаг...
ПРДОП	Доплата за работу в праздничные дни (дневное время)	2000	Доходы, целиком облаг...
СДЛ	Сдельный заработок	2000	Доходы, целиком облаг...
ОКЛЧС	Оплата по окладу (по часам)	2000	Доходы, целиком облаг...
ТРФЧС	Оплата по часовому тарифу	2000	Доходы, целиком облаг...
НОЧН	Доплата за работу в ночное время	2000	Доходы, целиком облаг...
СДЛОК	Сдельный заработок (для работающих по окладу)	2000	Доходы, целиком облаг...
ПРРП	Премия разовая (процентом)	2000	Доходы, целиком облаг...
ПРГП	Премия за год (процентом)	2000	Доходы, целиком облаг...

Создание и редактирование видов расчета может выполняться как разработчиком (предопределенные виды расчета), так и пользователем, в процессе работы с прикладным решением. Однако пользователь не может удалять виды расчета, созданные разработчиком.

Виды расчета, созданные в плане видов расчета, могут влиять друг на друга. Система поддерживает два вида такого влияния: **зависимость по базовому периоду** и **вытеснение по периоду действия**.

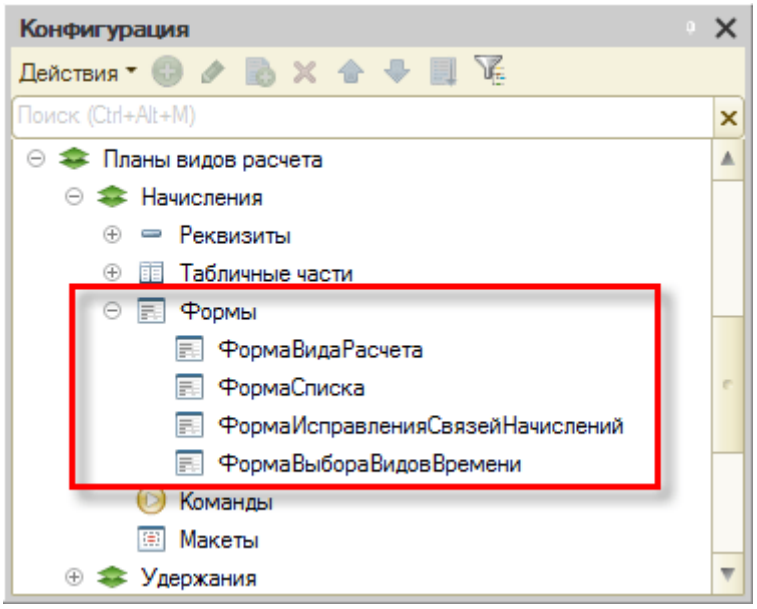
Для каждого вида расчета можно указать перечень видов расчета, от которых он будет зависеть по базовому периоду, и которые будут вытеснять его по периоду действия.

Кроме этих зависимостей, для вида расчета могут быть указаны так называемые ведущие виды расчета, — те, от которых он не зависит напрямую, но которые могут оказывать на него влияние через другие виды расчетов.

Формы плана видов расчета

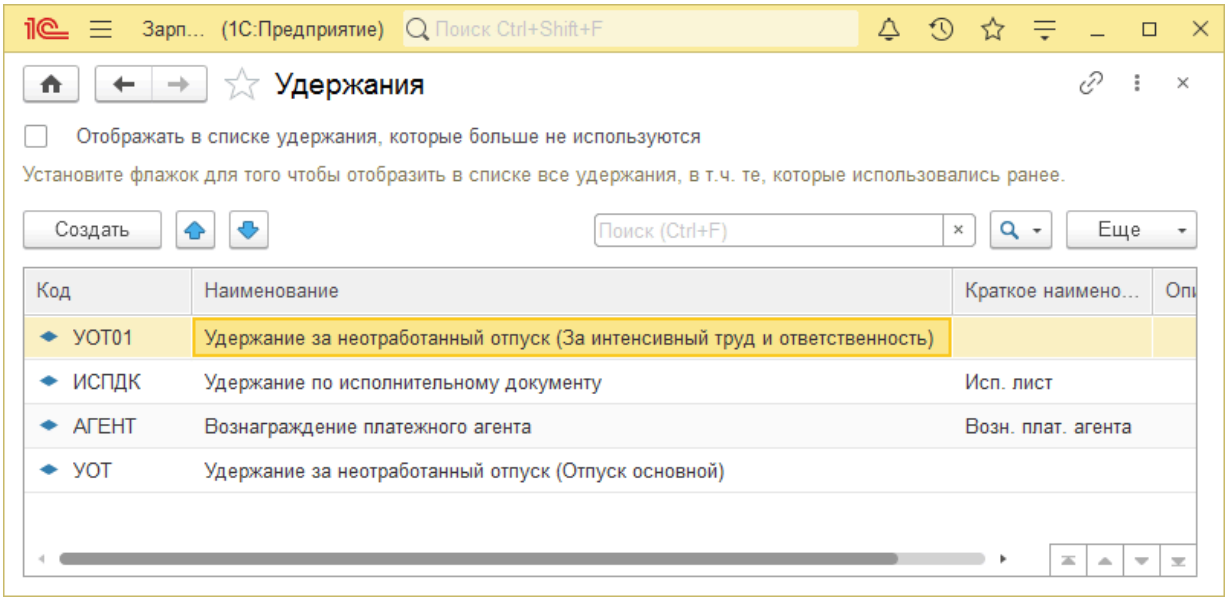
Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в плане видов расчета, система поддерживает несколько форм его представления. Система может автоматически генерировать все нужные

формы; наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию:



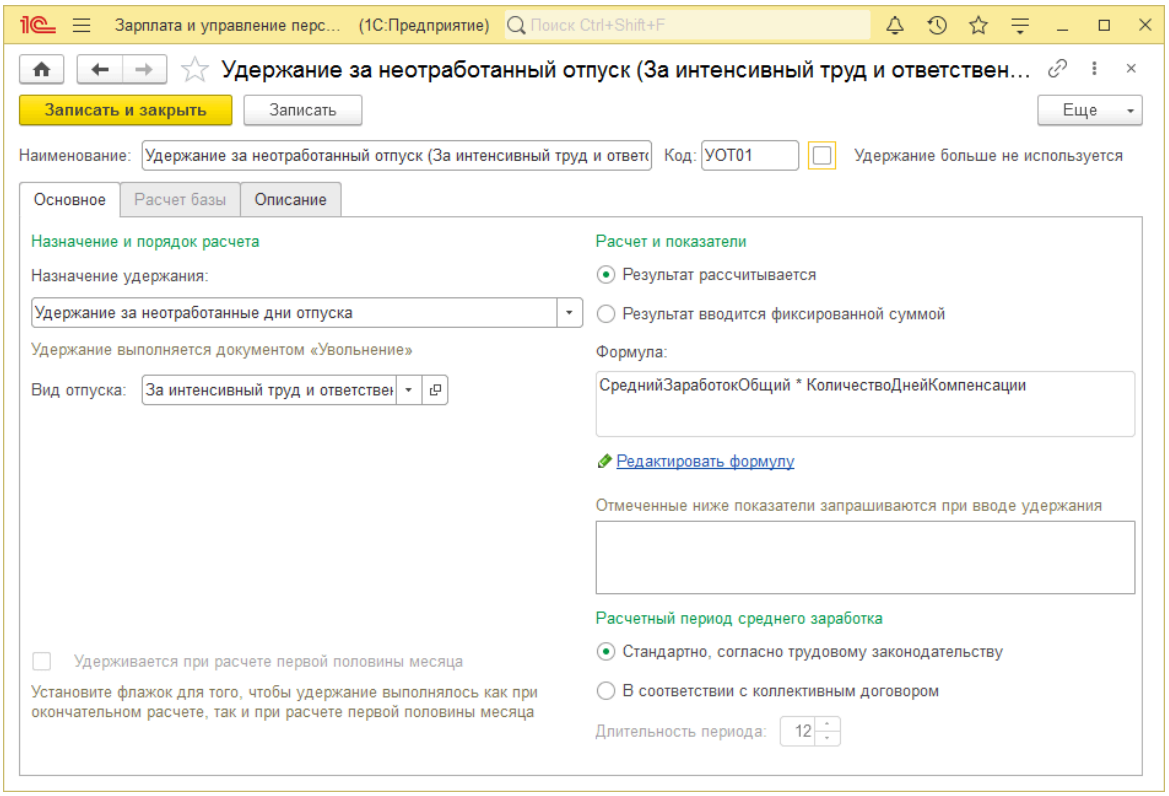
Форма списка

Для просмотра видов расчета используется форма списка. Она позволяет выполнять навигацию по списку, добавлять, помечать на удаление и удалять виды расчетов. Форма списка позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



Форма вида расчета

Для просмотра и изменения данных отдельных видов расчета используется форма вида расчета. Как правило, она представляет данные в удобном для восприятия и редактирования виде:



Кроме этих двух форм для видов расчета поддерживается форма выбора конкретных видов расчета из списка. Она, обычно, содержит минимальный набор информации, необходимой для выбора того или иного вида расчета.

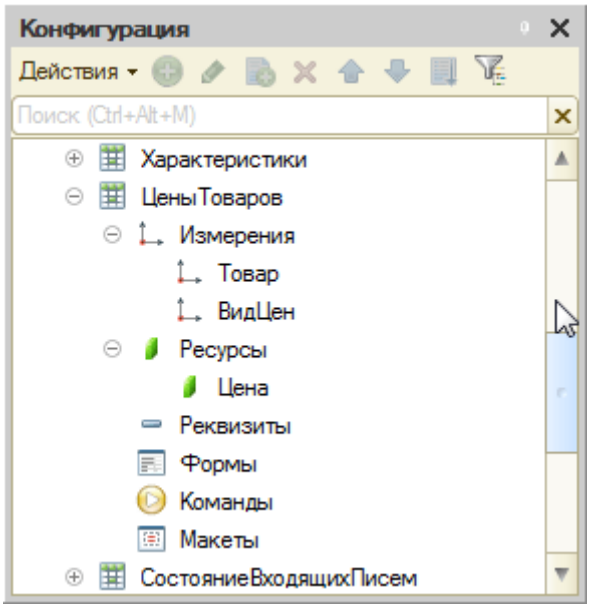
Регистр сведений

Регистры сведений — это прикладные объекты конфигурации. Они позволяют хранить в прикладном решении произвольные данные в разрезе нескольких измерений. Например, в регистре сведений можно хранить курсы валют в разрезе валют, или цены предприятия в разрезе номенклатуры и типа цен.

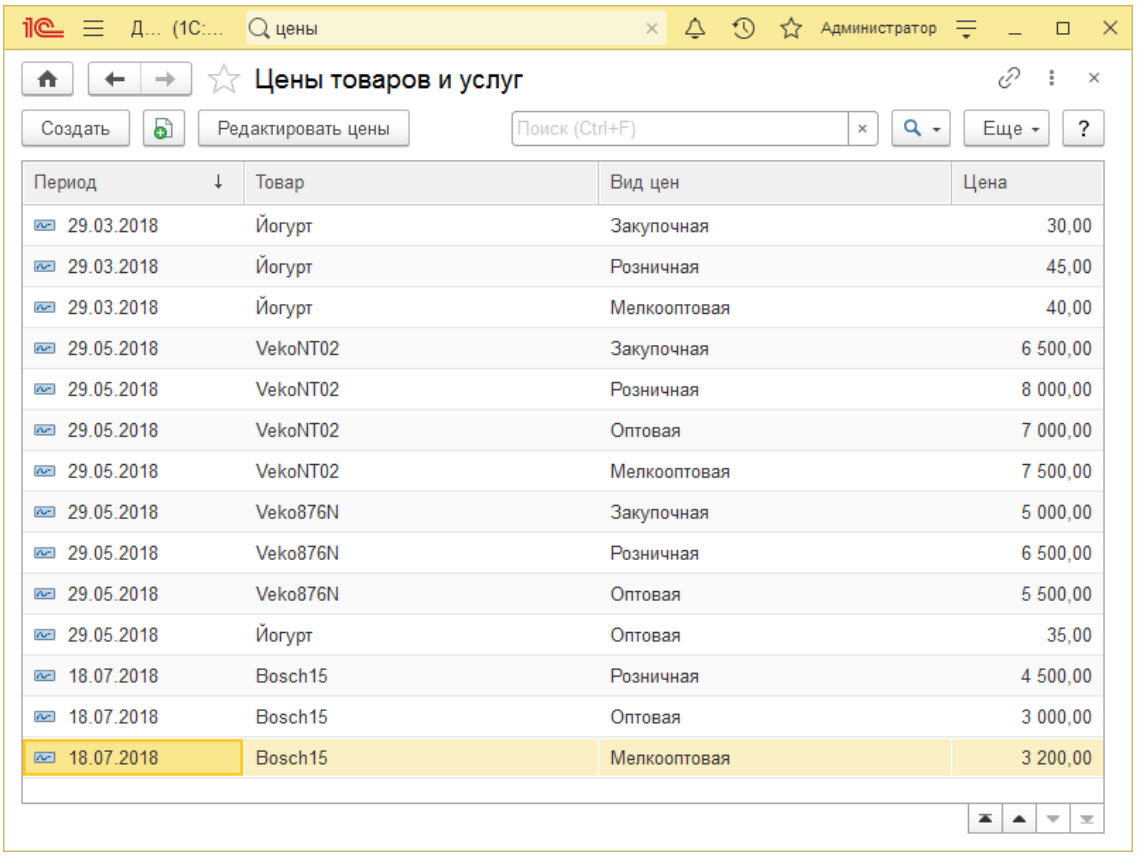
Структура

Информация в регистре сведений хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов.

Измерения регистра описывают разрезы, в которых хранится информация, а **ресурсы** регистра непосредственно содержат хранимую информацию. Например, для регистра сведений **Цены товаров**, который имеет следующую структуру:



записи, хранимые в базе данных, будут выглядеть следующим образом:

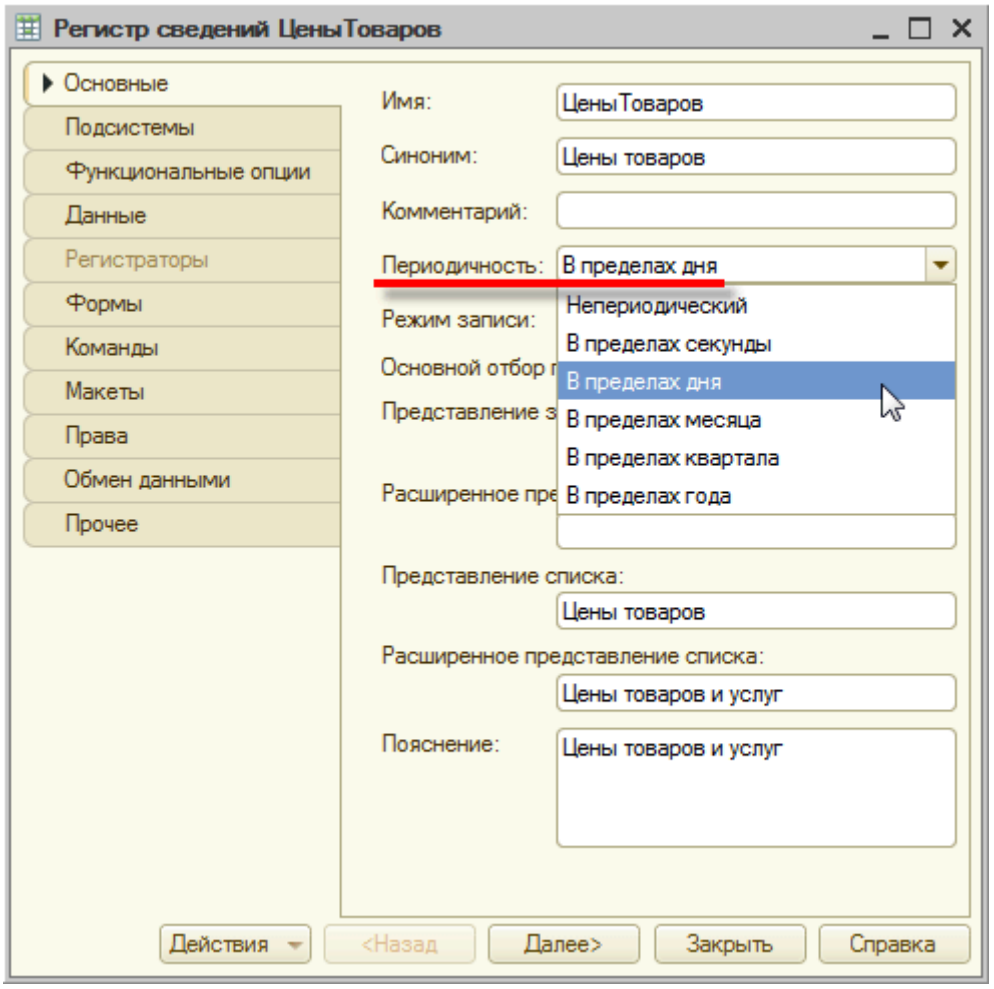


Период	Товар	Вид цен	Цена
29.03.2018	Йогурт	Закупочная	30,00
29.03.2018	Йогурт	Розничная	45,00
29.03.2018	Йогурт	Мелкооптовая	40,00
29.05.2018	VekoNT02	Закупочная	6 500,00
29.05.2018	VekoNT02	Розничная	8 000,00
29.05.2018	VekoNT02	Оптовая	7 000,00
29.05.2018	VekoNT02	Мелкооптовая	7 500,00
29.05.2018	Veko876N	Закупочная	5 000,00
29.05.2018	Veko876N	Розничная	6 500,00
29.05.2018	Veko876N	Оптовая	5 500,00
29.05.2018	Йогурт	Оптовая	35,00
18.07.2018	Bosch15	Розничная	4 500,00
18.07.2018	Bosch15	Оптовая	3 000,00
18.07.2018	Bosch15	Мелкооптовая	3 200,00

Вместе с каждой записью, находящейся в регистре сведений, можно хранить дополнительную произвольную информацию. Для этого служат реквизиты регистра сведений.

Периодичность

Одной из возможностей регистра сведений является хранение данных не только в разрезе указанных измерений, но и в разрезе времени. Разработчик может указать минимальную периодичность, с которой записи будут заноситься в регистр:

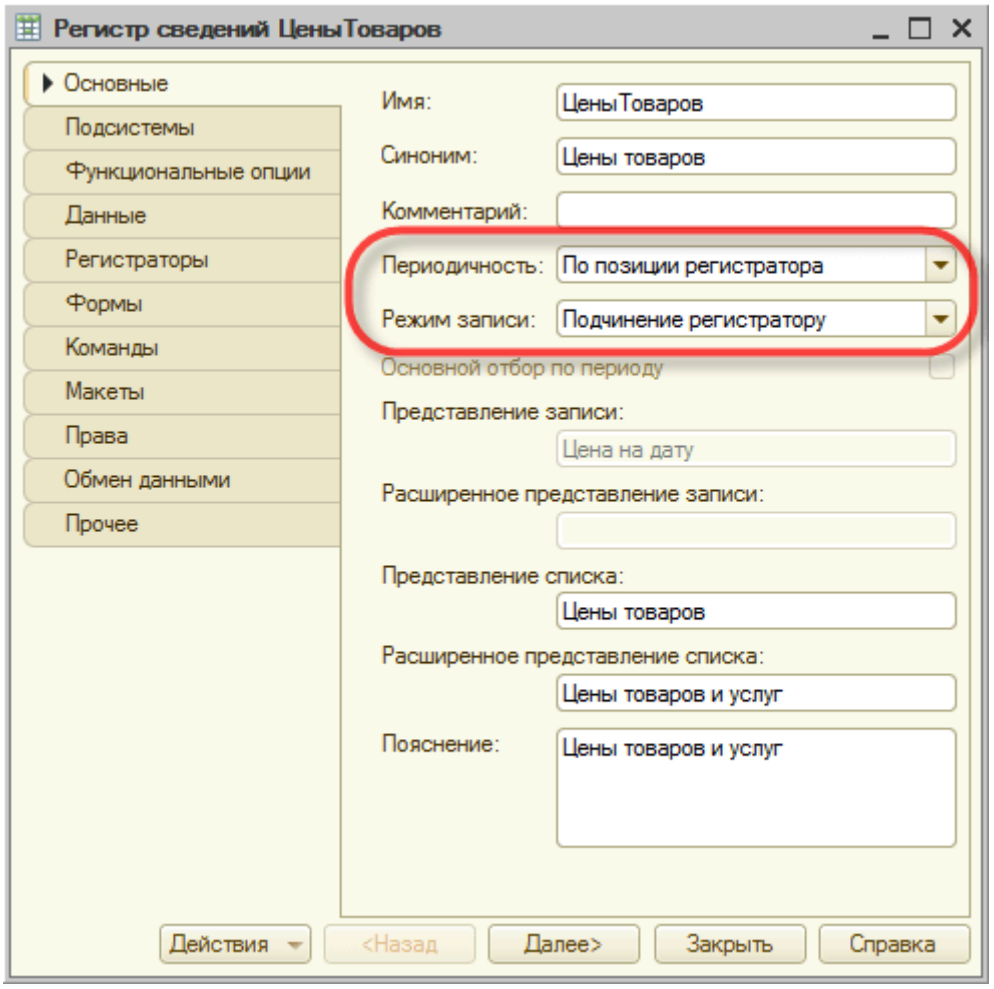


В этом случае к каждой записи регистра будет добавляться поле **Период**, хранящее дату, которой были внесены записи в регистр. Использование периодичности регистра сведений позволяет не просто хранить статические данные, но и отслеживать их изменение во времени.

Например, периодический регистр сведений **Цены товаров** может не только хранить информацию о том, какова цена на определенную номенклатуру сейчас, но и о том, как она изменялась в прошлом (или будет изменяться в будущем):

Подчинение регистратору

Внесение изменений в регистр сведений может выполняться как вручную, так и при помощи документов. В случае, когда изменения в регистр сведений вносятся с помощью документов, к каждой записи регистра добавляется специальное поле, в котором хранится информация о регистраторе — документе, с которым связана эта запись. В процессе создания прикладного решения разработчик указывает, какой именно режим записи будет использоваться данным регистром сведений:



Использование режима записи **Подчинение регистратору** может потребоваться в случае, когда логика работы прикладного решения требует того, чтобы изменения, выполняемые в регистре сведений, были жестко связаны с документами, фиксирующими факты хозяйственной деятельности.

Например, изменение цен компании может производиться только определенным кругом лиц, и каждое такое изменение должно сопровождаться «бумажным» документом. В этом случае можно использовать режим подчинения регистратору, при котором изменение цен может быть выполнено только специальным документом — **Изменение цен товаров**.

Уникальность записей

Система обеспечивает контроль уникальности записей, хранящихся в регистре сведений. Таким образом, в регистре сведений не может находиться двух одинаковых записей. Одинаковыми считаются записи, у которых совпадает ключ записи. Ключ записи формируется системой автоматически, на основании значений, содержащихся в полях записи, и зависит от вида регистра сведений.

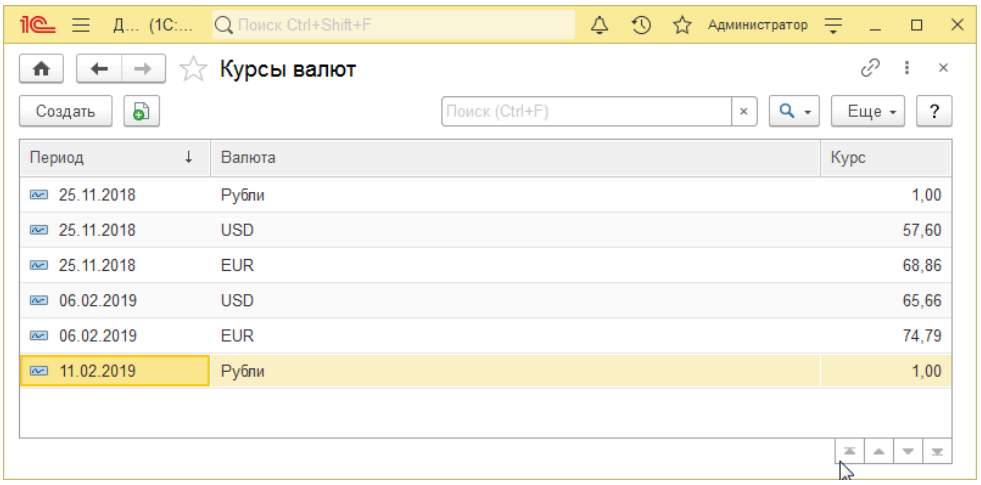
В общем случае в формировании ключа записи будут участвовать значения регистратора, периода и значения измерений. Таким образом, например, в непериодическом регистре сведений **Цены товаров** с независимым режимом записи не может существовать двух записей о розничной цене конфет ассорти. Точно так же, как в периодическом регистре сведений **Цены товаров**, подчиненном регистратору, не может существовать двух записей о розничной цене конфет ассорти, внесенных одной и той же датой, одним и тем же документом **Изменение цен товаров**.

Формы

Для того чтобы пользователь мог просматривать и изменять данные, содержащиеся в регистре сведений, система поддерживает несколько форм представления регистра. Система может автоматически генерировать все нужные формы регистра. Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо форм по умолчанию:

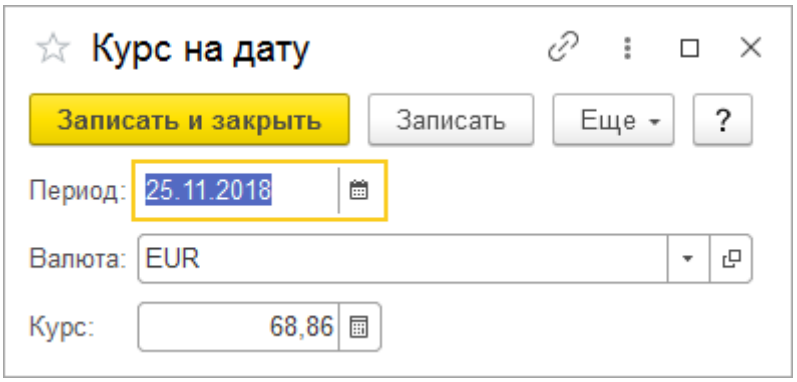
Форма списка

Для просмотра данных, содержащихся в регистре сведений, используется форма списка. Она позволяет выполнять навигацию по регистру, добавлять, помечать на удаление и удалять записи регистра. Форма списка позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



Форма записи

Для просмотра и изменения отдельных записей регистра сведений используется форма записи. Как правило, она представляет данные в удобном для восприятия и редактирования виде:



Функциональные возможности регистра сведений

Основными функциональными возможностями, которые предоставляет регистр сведений разработчику, являются:

- создание, изменение и удаление записей;
- выбор записей в заданном интервале по заданным критериям;
- выбор записей по регистратору;
- получение значений ресурсов записей, соответствующих указанному периоду и значениям измерений;
- получение значений ресурсов наиболее ранних и наиболее поздних записей регистра, соответствующих указанному периоду и значениям измерений.

Регистр накопления

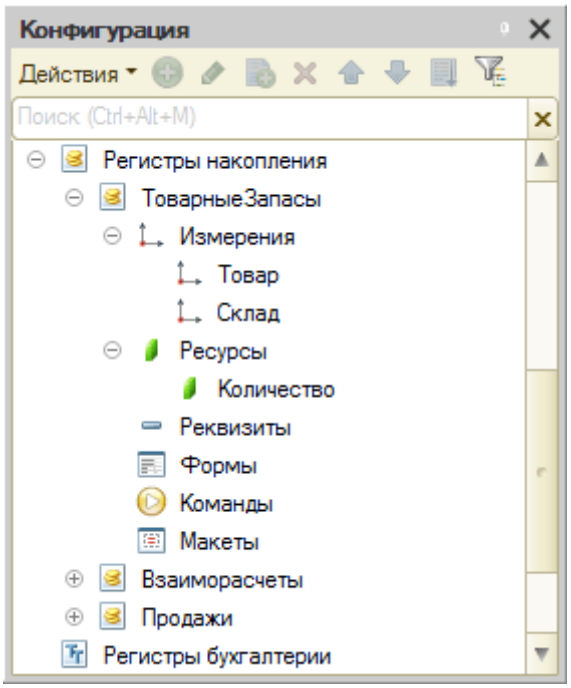
Регистры накопления — это прикладные объекты конфигурации. Они составляют основу механизма учета движения средств (финансов, товаров, материалов и т. д.), который позволяет автоматизировать такие направления, как складской учет, взаиморасчеты, планирование.

Регистр накопления образует многомерную систему измерений и позволяет «накапливать» числовые данные в разрезе нескольких измерений. Например, в таком регистре можно накапливать информацию об остатках товаров в разрезе номенклатуры и склада, или информацию об объемах продаж в разрезе номенклатуры и подразделения компании.

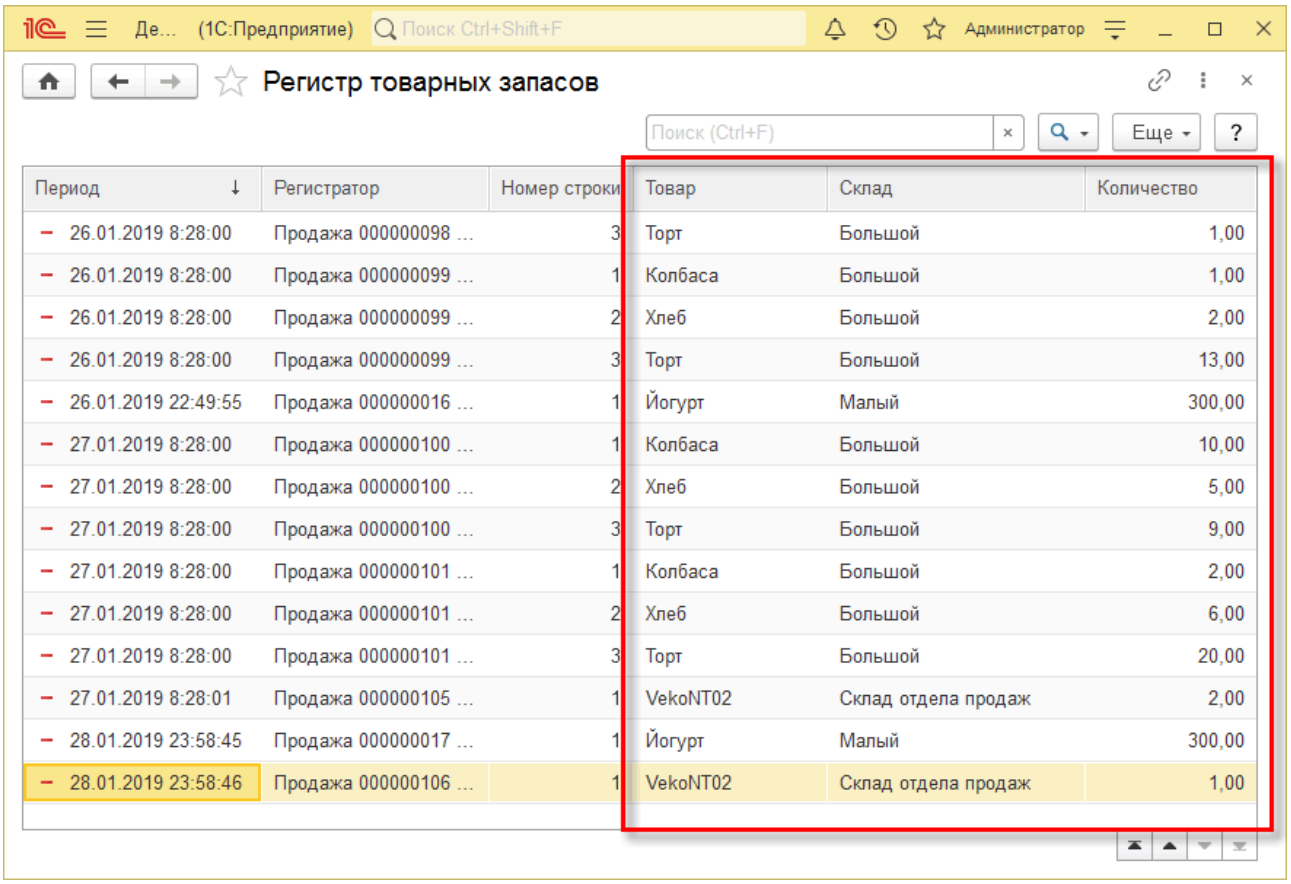
Структура

Информация в регистре накопления хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов.

Измерения регистра описывают разрезы, в которых хранится информация, а в ресурсах регистра накапливаются нужные числовые данные. Например, для регистра ТоварныеЗапасы, который имеет следующую структуру:



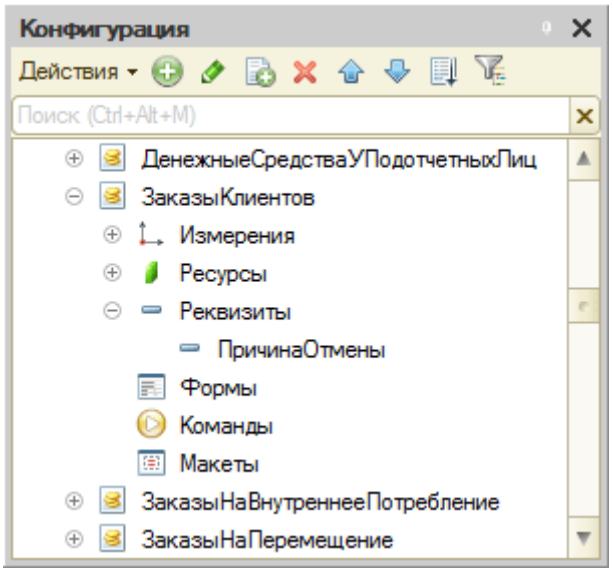
записи, производящие изменение ресурсов регистра в базе данных, будут выглядеть следующим образом:



Период	Регистратор	Номер строки	Товар	Склад	Количество
26.01.2019 8:28:00	Продажа 000000098 ...	3	Торт	Большой	1,00
26.01.2019 8:28:00	Продажа 000000099 ...	1	Колбаса	Большой	1,00
26.01.2019 8:28:00	Продажа 000000099 ...	2	Хлеб	Большой	2,00
26.01.2019 8:28:00	Продажа 000000099 ...	3	Торт	Большой	13,00
26.01.2019 22:49:55	Продажа 000000016 ...	1	Йогурт	Малый	300,00
27.01.2019 8:28:00	Продажа 000000100 ...	1	Колбаса	Большой	10,00
27.01.2019 8:28:00	Продажа 000000100 ...	2	Хлеб	Большой	5,00
27.01.2019 8:28:00	Продажа 000000100 ...	3	Торт	Большой	9,00
27.01.2019 8:28:00	Продажа 000000101 ...	1	Колбаса	Большой	2,00
27.01.2019 8:28:00	Продажа 000000101 ...	2	Хлеб	Большой	6,00
27.01.2019 8:28:00	Продажа 000000101 ...	3	Торт	Большой	20,00
27.01.2019 8:28:01	Продажа 000000105 ...	1	VekoNT02	Склад отдела продаж	2,00
28.01.2019 23:58:45	Продажа 000000017 ...	1	Йогурт	Малый	300,00
28.01.2019 23:58:46	Продажа 000000106 ...	1	VekoNT02	Склад отдела продаж	1,00

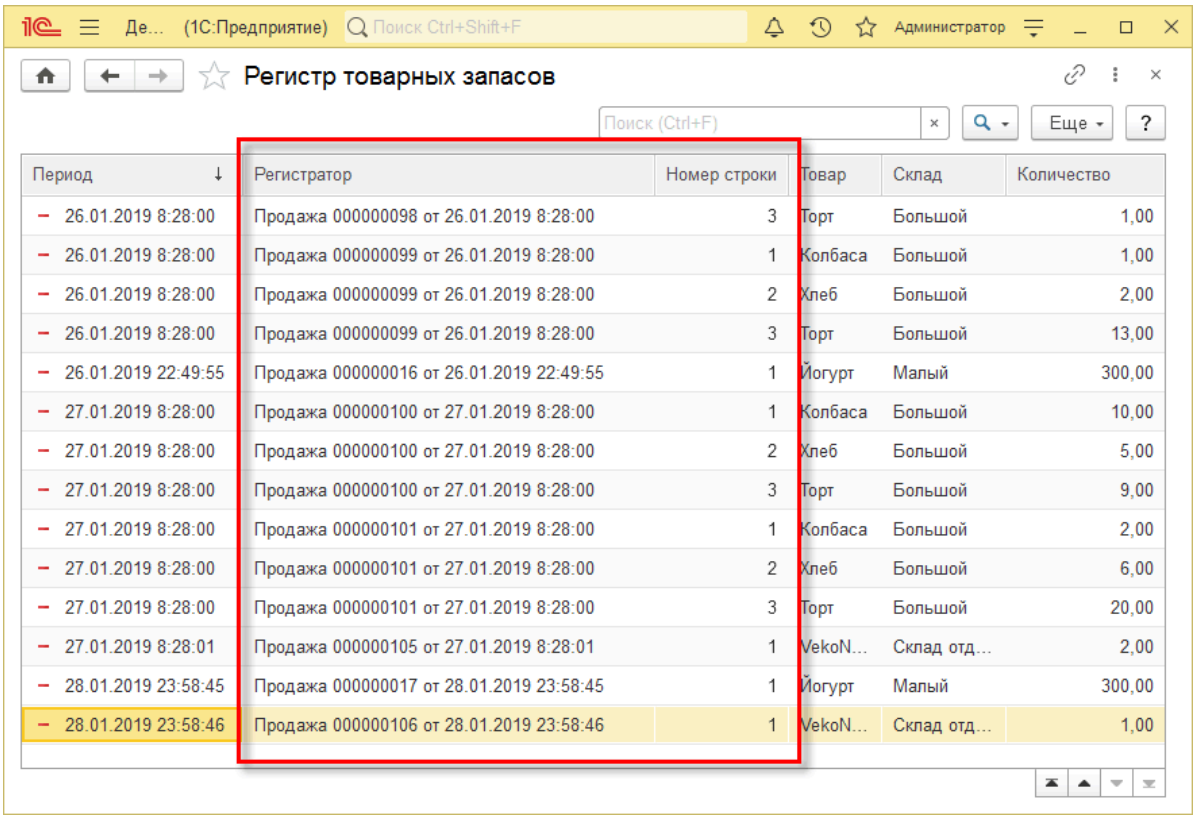
Поскольку регистр накопления служит для накопления числовых значений, каждая запись выполняет изменение хранимых ресурсов — движение. Движения, в общем случае, могут либо добавлять некоторые приращения к хранимым ресурсам, либо отнимать их. Если должно выполняться увеличение хранимых ресурсов, — такое движение называется движением прихода («+»), если уменьшение хранимых ресурсов — движением расхода («-»).

Вместе с каждой записью, находящейся в регистре накопления, можно хранить дополнительную произвольную информацию. Для этого служат реквизиты регистра накопления.



Связь с регистратором

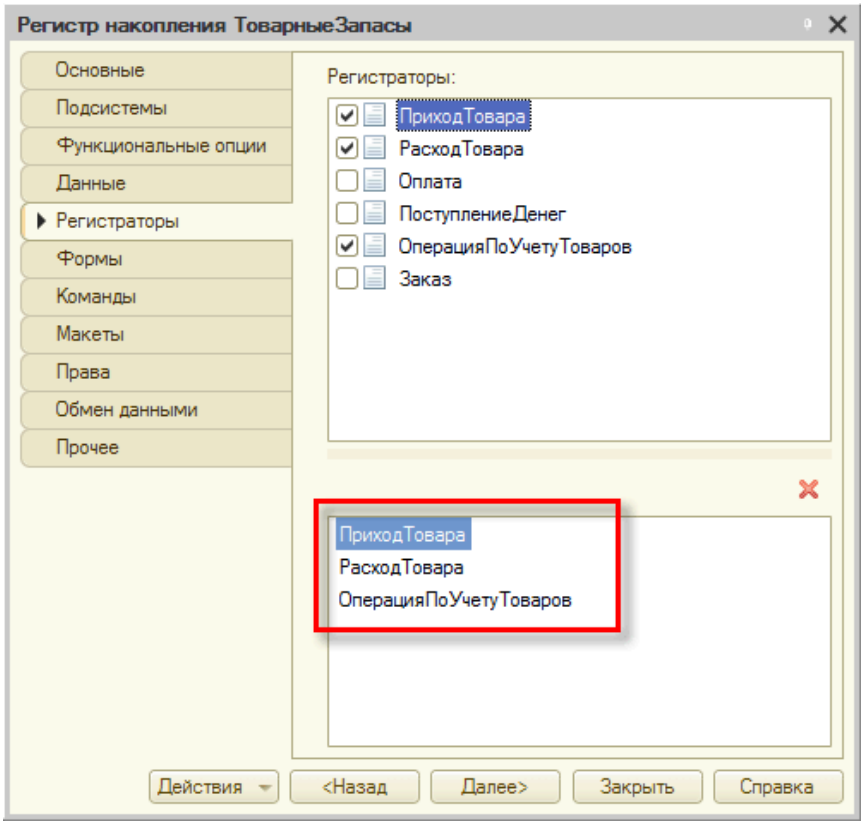
Изменение состояния регистра накопления происходит, как правило, при проведении документа. Поэтому каждая запись регистра связана с определенным документом — регистратором, номером строки этого документа, и датой — периодом:



Период	Регистратор	Номер строки	Товар	Склад	Количество
26.01.2019 8:28:00	Продажа 000000098 от 26.01.2019 8:28:00	3	Торт	Большой	1,00
26.01.2019 8:28:00	Продажа 000000099 от 26.01.2019 8:28:00	1	Колбаса	Большой	1,00
26.01.2019 8:28:00	Продажа 000000099 от 26.01.2019 8:28:00	2	Хлеб	Большой	2,00
26.01.2019 8:28:00	Продажа 000000099 от 26.01.2019 8:28:00	3	Торт	Большой	13,00
26.01.2019 22:49:55	Продажа 000000016 от 26.01.2019 22:49:55	1	Йогурт	Малый	300,00
27.01.2019 8:28:00	Продажа 000000100 от 27.01.2019 8:28:00	1	Колбаса	Большой	10,00
27.01.2019 8:28:00	Продажа 000000100 от 27.01.2019 8:28:00	2	Хлеб	Большой	5,00
27.01.2019 8:28:00	Продажа 000000100 от 27.01.2019 8:28:00	3	Торт	Большой	9,00
27.01.2019 8:28:00	Продажа 000000101 от 27.01.2019 8:28:00	1	Колбаса	Большой	2,00
27.01.2019 8:28:00	Продажа 000000101 от 27.01.2019 8:28:00	2	Хлеб	Большой	6,00
27.01.2019 8:28:00	Продажа 000000101 от 27.01.2019 8:28:00	3	Торт	Большой	20,00
27.01.2019 8:28:01	Продажа 000000105 от 27.01.2019 8:28:01	1	VekoN...	Склад отд...	2,00
28.01.2019 23:58:45	Продажа 000000017 от 28.01.2019 23:58:45	1	Йогурт	Малый	300,00
28.01.2019 23:58:46	Продажа 000000106 от 28.01.2019 23:58:46	1	VekoN...	Склад отд...	1,00

В общем случае значение поле Период может не совпадать с датой документа. Например, документ План продаж может внести в регистр накопления записи о предполагаемых продажах компании несколькими разными датами.

Состав документов, которые могут создавать записи в регистре накопления, задается разработчиком в процессе создания прикладного решения:



Конструктор движений

Алгоритмы, по которым формируются записи в регистре, описываются средствами встроенного языка в процедурах соответствующих документов. Система содержит конструктор движений, который помогает разработчику создавать алгоритмы проведения документа. Подробнее...

Уникальность записей

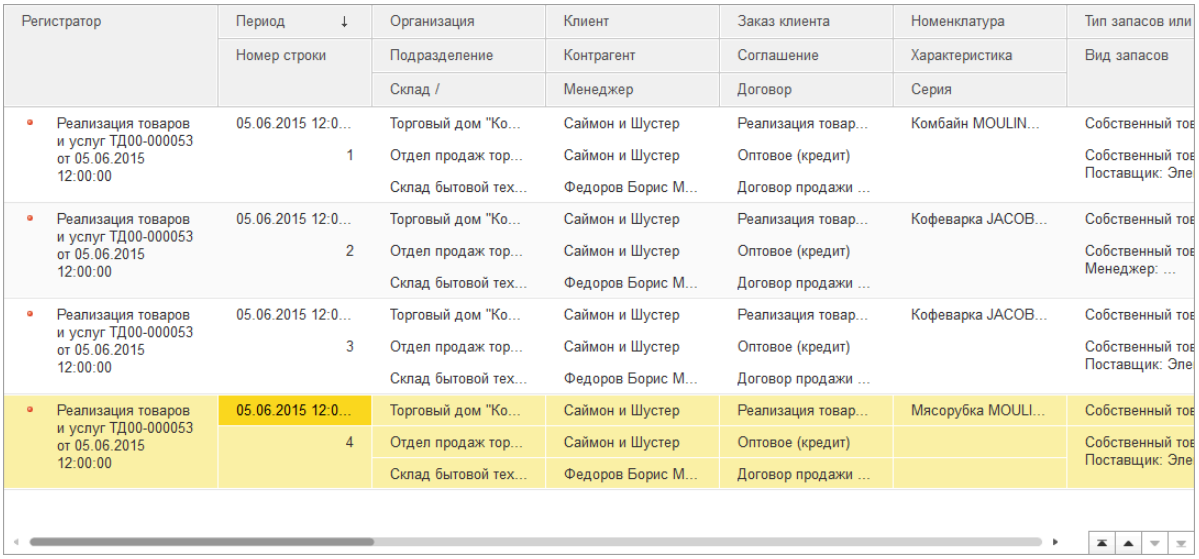
Система обеспечивает контроль уникальности записей, хранящихся в регистре накопления. Благодаря этому в регистре накоплений не может находиться двух записей, относящихся к одной и той же строке одного и того же документа.

Регистры остатков и регистры оборотов

Существует два вида регистров накопления: регистры накопления остатков и регистры накопления оборотов. Регистр хранить как итоговые значения ресурсов — остатки, так и изменения этих ресурсов — обороты. Регистр

накопления оборотов является более «специализированным» видом регистра накопления и позволяет хранить только изменения ресурсов — обороты.

Существование регистра накопления оборотов связано с тем, что при автоматизации экономической деятельности существует большое количество ситуаций, когда требуется накапливать только обороты, а значения остатков не имеют смысла. Типичным примером использования регистра накопления оборотов является регистр Выручка и себестоимость продаж, хранящий информацию об объемах продаж:



Регистратор	Период ↓	Организация	Клиент	Заказ клиента	Номенклатура	Тип запасов или
	Номер строки	Подразделение	Контрагент	Соглашение	Характеристика	Вид запасов
		Склад /	Менеджер	Договор	Серия	
Реализация товаров и услуг ТД00-000053 от 05.06.2015 12:00:00	05.06.2015 12:0...	Торговый дом "Ко...	Саймон и Шустер	Реализация товар...	Комбайн MOULIN...	Собственный тов
	1	Отдел продаж тор...	Саймон и Шустер	Оптовое (кредит)		Собственный тов
		Склад бытовой тех...	Федоров Борис М...	Договор продажи ...		Поставщик: Эле
Реализация товаров и услуг ТД00-000053 от 05.06.2015 12:00:00	05.06.2015 12:0...	Торговый дом "Ко...	Саймон и Шустер	Реализация товар...	Кофеварка JACOB...	Собственный тов
	2	Отдел продаж тор...	Саймон и Шустер	Оптовое (кредит)		Собственный тов
		Склад бытовой тех...	Федоров Борис М...	Договор продажи ...		Менеджер: ...
Реализация товаров и услуг ТД00-000053 от 05.06.2015 12:00:00	05.06.2015 12:0...	Торговый дом "Ко...	Саймон и Шустер	Реализация товар...	Кофеварка JACOB...	Собственный тов
	3	Отдел продаж тор...	Саймон и Шустер	Оптовое (кредит)		Собственный тов
		Склад бытовой тех...	Федоров Борис М...	Договор продажи ...		Поставщик: Эле
Реализация товаров и услуг ТД00-000053 от 05.06.2015 12:00:00	05.06.2015 12:0...	Торговый дом "Ко...	Саймон и Шустер	Реализация товар...	Мясорубка MOULI...	Собственный тов
	4	Отдел продаж тор...	Саймон и Шустер	Оптовое (кредит)		Собственный тов
		Склад бытовой тех...	Федоров Борис М...	Договор продажи ...		Поставщик: Эле

Поскольку регистр накопления оборотов не накапливает остатки ресурсов, для него не имеет смысла «направление» движения ресурсов (приход или расход); накапливается только величина изменения ресурсов. Поэтому все записи регистра накопления оборотов отмечены одинаковыми пиктограммами.

Агрегаты

Для оборотных регистров накопления платформа поддерживает специальный механизм агрегатов, который позволяет значительно ускорить получение данных из регистров, содержащих большое количество записей — сотни тысяч и миллионы записей. Подробнее...

Форма списка и форма набора записей

Для того чтобы пользователь мог просматривать данные, содержащиеся в регистре накопления, система поддерживает форму представления регистра накопления — форму списка. Она позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям.

Система может автоматически генерировать эту форму. Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо формы умолчанию, в том числе и форму набора записей, которая позволяет добавлять, изменять и удалять записи регистра сведений.

Функциональные возможности регистра накопления

Основными функциональными возможностями, которые предоставляет регистр накопления разработчику, являются:

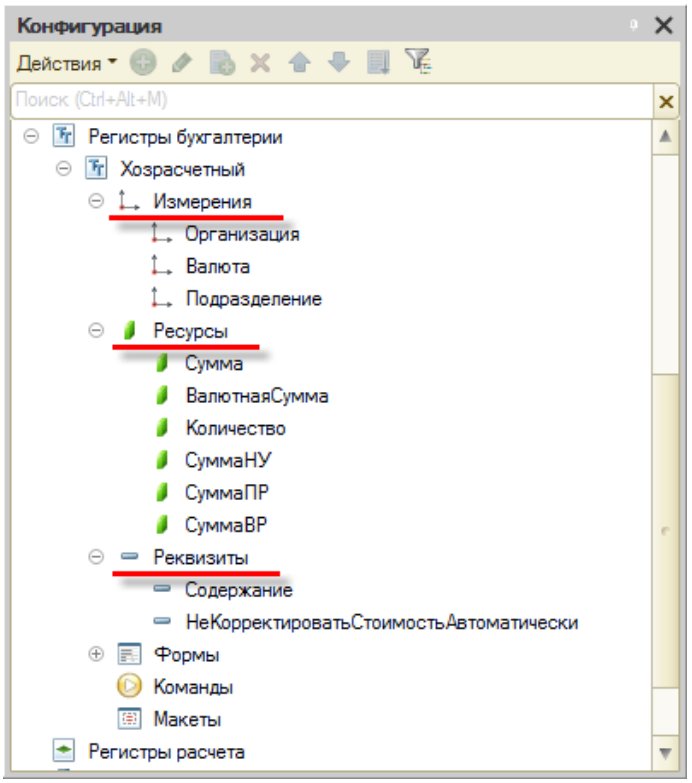
- выбор записей в заданном интервале по заданным критериям;
- выбор записей по регистратору;
- получение остатков и оборотов на указанный момент времени по заданным значениям измерений;
- режим работы с разделением итогов, который обеспечивает более высокую параллельность записи в регистр;
- отключение использования текущих итогов;
- расчет итогов на указанную дату;
- чтение, изменение и запись набора записей в регистр;
- возможность записи в регистр без пересчета итогов;
- полный пересчет итогов и пересчет итогов за указанный период.

Регистр бухгалтерии

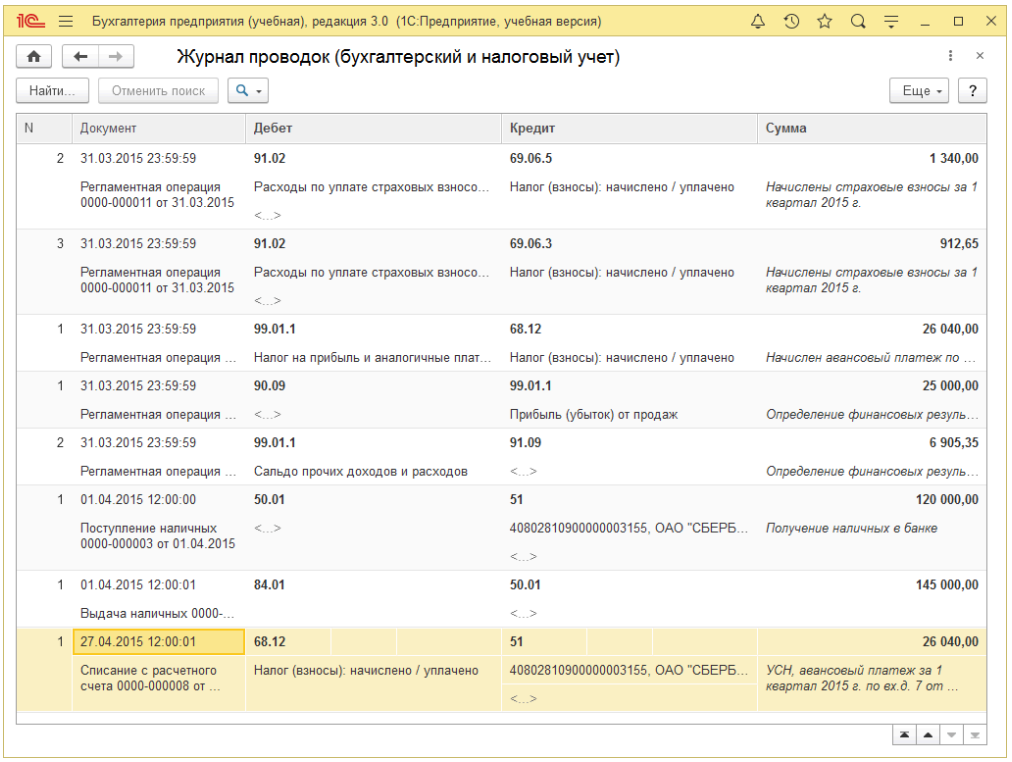
Регистры бухгалтерии — это прикладные объекты конфигурации. Они используются в механизме бухгалтерского учета и позволяют вести многоуровневый и многомерный аналитический учет, учет по нескольким планам счетов, опциональное ведение количественного, суммового и валютного учета по отдельным разрезам аналитики и т. д.

Структура

По своему виду регистр бухгалтерии напоминает регистр накопления: он может иметь набор **измерений**, в разрезе которых накапливаются числовые значения **ресурсов**. Также он может содержать набор **реквизитов** для хранения дополнительной информации, связанной с записями регистра:

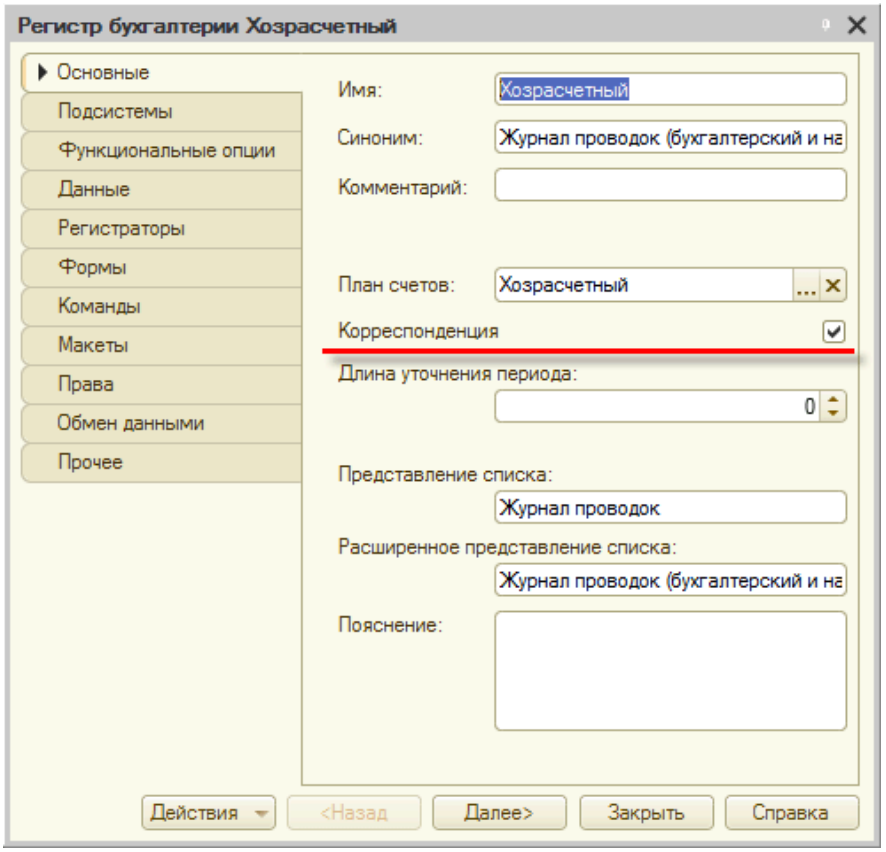


Регистр бухгалтерии связывается с одним из используемых планов счетов, и хранит бухгалтерские итоги в соответствии с его структурой. Информация в регистре бухгалтерии хранится в виде записей, структура которых строится системой динамически, в зависимости от настроек различных элементов учета, выполненной при редактировании плана счетов. Например, запись может содержать поля для ввода корреспондирующих счетов, сумм, субконто, количества, вида валюты и суммы в валюте:



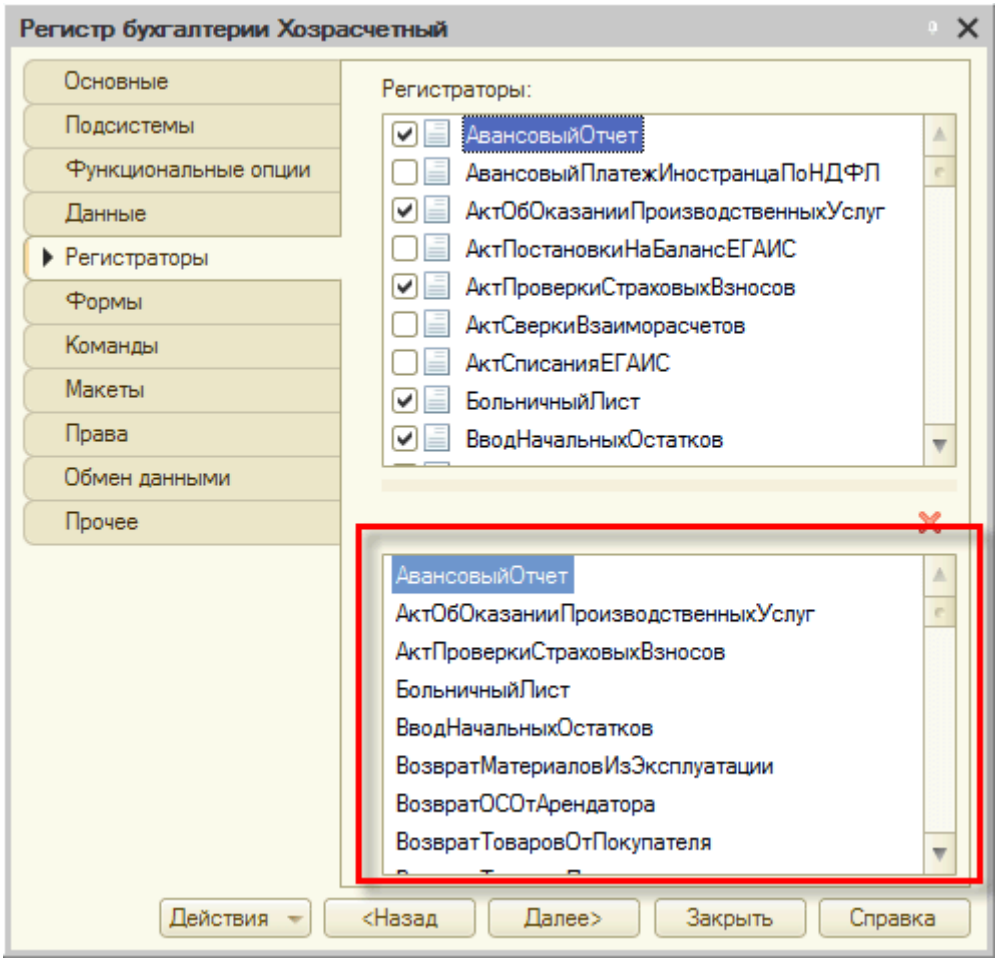
N	Документ	Дебет	Кредит	Сумма
2	31.03.2015 23:59:59	**91.02**	**69.06.5**	**1 340,00**
	Регламентная операция 0000-000011 от 31.03.2015	Расходы по уплате страховых взносо… <…>	Налог (взносы): начислено / уплачено	*Начислены страховые взносы за 1 квартал 2015 г.*
3	31.03.2015 23:59:59	**91.02**	**69.06.3**	**912,65**
	Регламентная операция 0000-000011 от 31.03.2015	Расходы по уплате страховых взносо… <…>	Налог (взносы): начислено / уплачено	*Начислены страховые взносы за 1 квартал 2015 г.*
1	31.03.2015 23:59:59	**99.01.1**	**68.12**	**26 040,00**
	Регламентная операция …	Налог на прибыль и аналогичные плат…	Налог (взносы): начислено / уплачено	*Начислен авансовый платеж по …*
1	31.03.2015 23:59:59	**90.09**	**99.01.1**	**25 000,00**
	Регламентная операция …	<…>	Прибыль (убыток) от продаж	*Определение финансовых резуль…*
2	31.03.2015 23:59:59	**99.01.1**	**91.09**	**6 905,35**
	Регламентная операция …	Сальдо прочих доходов и расходов	<…>	*Определение финансовых резуль…*
1	01.04.2015 12:00:00	**50.01**	**51**	**120 000,00**
	Поступление наличных 0000-000003 от 01.04.2015	<…>	40802810900000003155, ОАО "СБЕРБ… <…>	*Получение наличных в банке*
1	01.04.2015 12:00:01	**84.01**	**50.01**	**145 000,00**
	Выдача наличных 0000-…		<…>	
1	27.04.2015 12:00:01	**68.12**	**51**	**26 040,00**
	Списание с расчетного счета 0000-000008 от …	Налог (взносы): начислено / уплачено	40802810900000003155, ОАО "СБЕРБ… <…>	*УСН, авансовый платеж за 1 квартал 2015 г. по вх.д. 7 от …*

Регистр бухгалтерии может работать в двух режимах: с поддержкой корреспонденции и без. В случае если выбран режим с поддержкой корреспонденции, будет задействован механизм двойной записи и каждая запись регистра будет содержать обязательные поля **СчетДт** (счет дебета) и **СчетКт** (счет кредита). Если же регистр не поддерживает корреспонденцию, каждая запись регистра будет содержать только одно поле — **Счет**:



Связь с регистратором

Поскольку записи регистра бухгалтерии отражают те или иные хозяйственные операции, выполняемые на предприятии, каждая из них связана с документом, — регистратором, номером строки этого документа, и датой — периодом. Состав документов, которые могут создавать записи в регистре бухгалтерии, задается разработчиком в процессе создания прикладного решения:



Конструктор движений

Алгоритмы, по которым формируются записи в регистре, описываются средствами встроенного языка в процедурах соответствующих документов. Система содержит конструктор движений, который помогает разработчику создавать алгоритмы проведения документа. Подробнее...

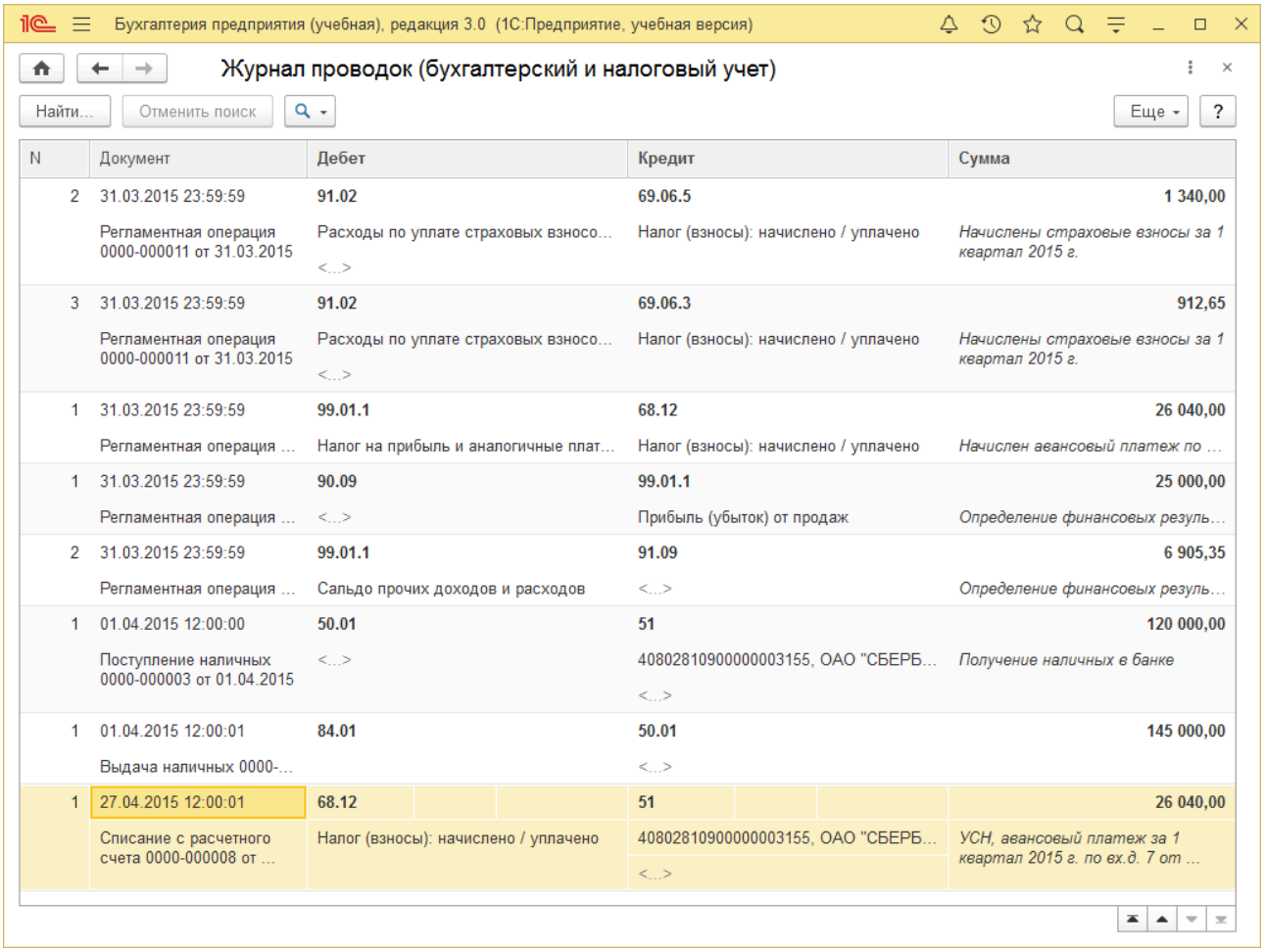
Уникальность записей

Система обеспечивает контроль уникальности записей, хранящихся в регистре бухгалтерии. Благодаря этому в регистре бухгалтерии не может находиться двух записей, относящихся к одной и той же строке одного и того же документа.

Форма списка

Для того чтобы пользователь мог просматривать данные, содержащиеся в регистре бухгалтерии, система поддерживает форму представления

регистра бухгалтерии — форму списка. Она позволяет выполнять сортировку и отбор отображаемой информации по нескольким критериям:



N	Документ	Дебет	Кредит	Сумма
2	31.03.2015 23:59:59	91.02	69.06.5	1 340,00
	Регламентная операция 0000-000011 от 31.03.2015	Расходы по уплате страховых взносо... <...>	Налог (взносы): начислено / уплачено	*Начислены страховые взносы за 1 квартал 2015 г.*
3	31.03.2015 23:59:59	91.02	69.06.3	912,65
	Регламентная операция 0000-000011 от 31.03.2015	Расходы по уплате страховых взносо... <...>	Налог (взносы): начислено / уплачено	*Начислены страховые взносы за 1 квартал 2015 г.*
1	31.03.2015 23:59:59	99.01.1	68.12	26 040,00
	Регламентная операция ...	Налог на прибыль и аналогичные плат...	Налог (взносы): начислено / уплачено	*Начислен авансовый платеж по ...*
1	31.03.2015 23:59:59	90.09	99.01.1	25 000,00
	Регламентная операция ...	<...>	Прибыль (убыток) от продаж	*Определение финансовых резуль...*
2	31.03.2015 23:59:59	99.01.1	91.09	6 905,35
	Регламентная операция ...	Сальдо прочих доходов и расходов	<...>	*Определение финансовых резуль...*
1	01.04.2015 12:00:00	50.01	51	120 000,00
	Поступление наличных 0000-000003 от 01.04.2015	<...>	40802810900000003155, ОАО "СБЕРБ... <...>	*Получение наличных в банке*
1	01.04.2015 12:00:01	84.01	50.01	145 000,00
	Выдача наличных 0000-...		<...>	
1	27.04.2015 12:00:01	68.12	51	26 040,00
	Списание с расчетного счета 0000-000008 от ...	Налог (взносы): начислено / уплачено	40802810900000003155, ОАО "СБЕРБ... <...>	*УСН, авансовый платеж за 1 квартал 2015 г. по вх.д. 7 от ...*

Система может автоматически генерировать эту форму. Наряду с этим разработчик имеет возможность создать собственные формы, которые система будет использовать вместо формы по умолчанию, в том числе и форму набора записей, которая позволяет добавлять, изменять и удалять записи регистра бухгалтерии.

Функциональные возможности регистра бухгалтерии

Основными функциональными возможностями, которые предоставляет регистр бухгалтерии разработчику, являются:

- выбор записей в заданном интервале по заданным критериям;
- выбор записей по регистратору;
- получение остатков и оборотов на указанный момент времени по заданным значениям параметров (счет, субконто, измерения, кор. счет, кор. субконто, кор. измерения);

- получение остатков на указанный момент времени по заданным значениям параметров (счетДт, субконтоДт, счетКт, субконтоКт, измеренияКт, измерения (для балансовых) и измеренияДт, измеренияКт (для не балансовых));
- режим работы с разделением итогов, который обеспечивает более высокую параллельность записи в регистр;
- отключение использования текущих итогов;
- расчет итогов на указанную дату;
- чтение, изменение и запись набора записей в регистр;
- возможность записи в регистр без пересчета итогов;
- полный пересчет итогов и пересчет итогов за указанный период.

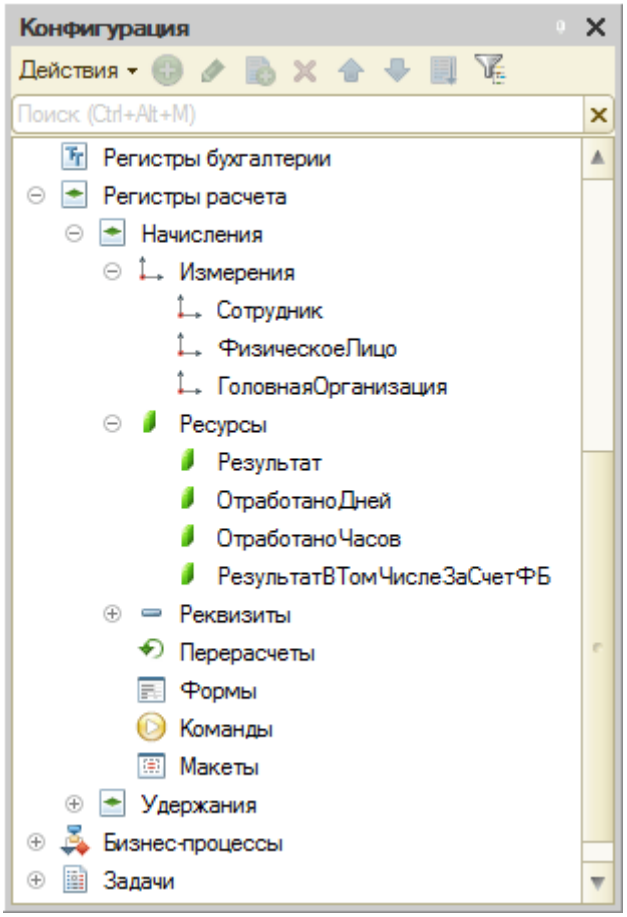
Регистр расчета

Регистры расчета — это прикладные объекты конфигурации. Они используются в механизме сложных периодических расчетов и служат для хранения записей о тех или иных видах расчета, которые необходимо выполнить, а также для хранения промежуточных данных и самих результатов выполненных расчетов.

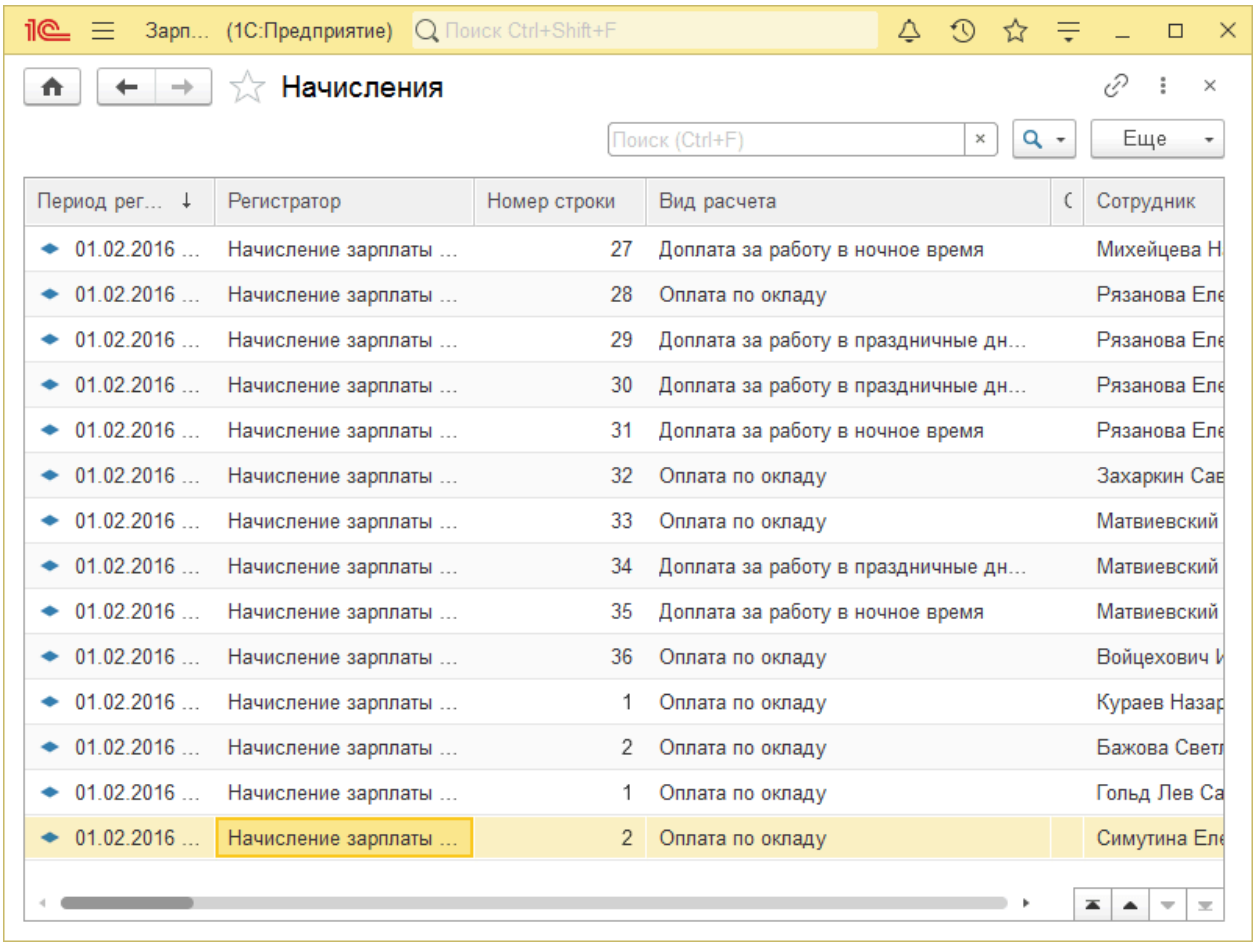
Структура

Информация в регистре расчета хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов.

Измерения регистра описывают разрезы, в которых хранится информация, а **ресурсы** регистра непосредственно содержат хранимую информацию. Например, для регистра расчета **Начисления**, который имеет следующую структуру:



записи, хранимые в базе данных, будут выглядеть следующим образом:

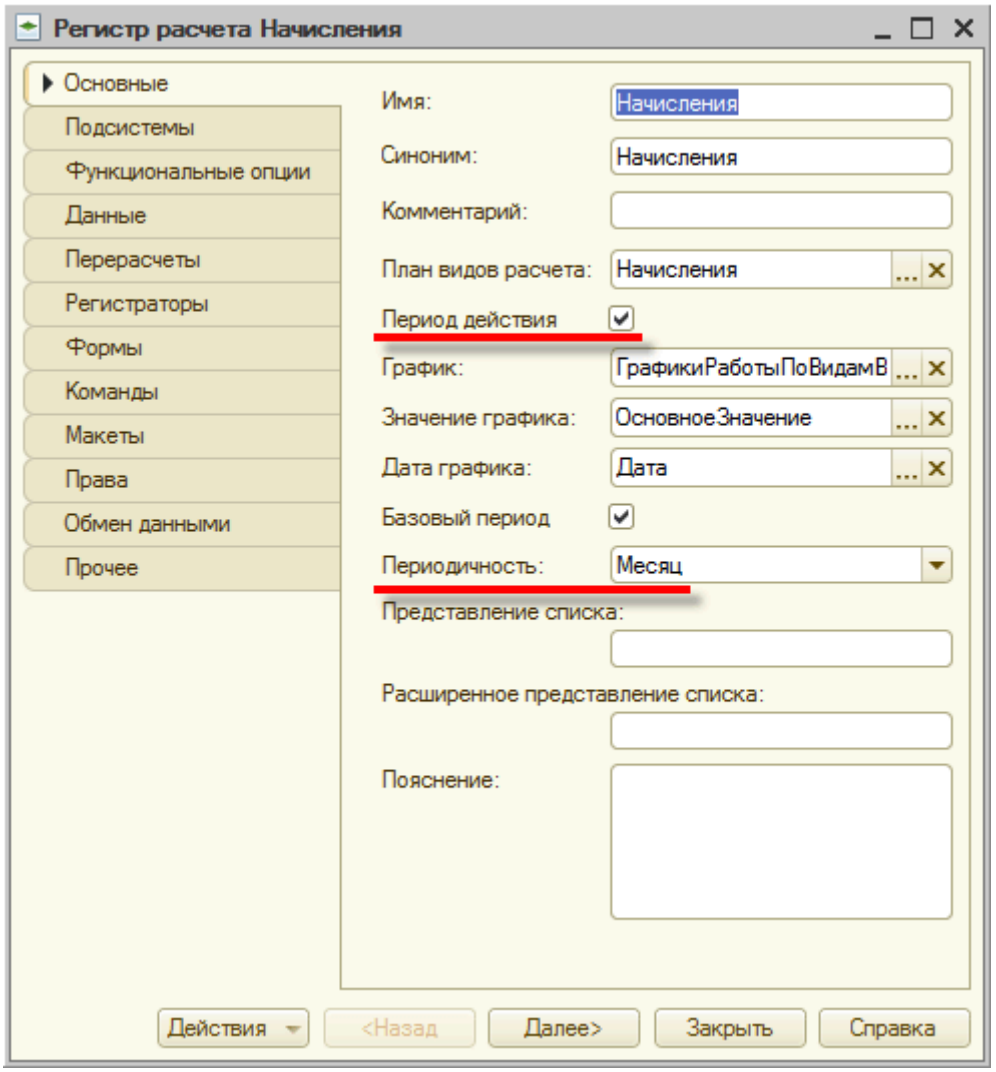


Связь с планом видов расчета

Регистр расчета связан с одним из планов видов расчета, существующих в прикладном решении. Эта связь обуславливает наличие у каждой записи регистра поля **Вид расчета**, благодаря которому механизмы регистра могут отслеживать взаимное влияние записей расчета друг на друга.

Периодичность

Регистр расчета хранит данные не только в разрезе созданных измерений, но и в разрезе времени. Это является причиной существования у каждой записи регистра расчета еще одного обязательного поля — **Период действия**. При создании регистра расчета разработчик может указать минимальную периодичность, с которой записи будут заноситься в регистр:



Подчинение регистратору

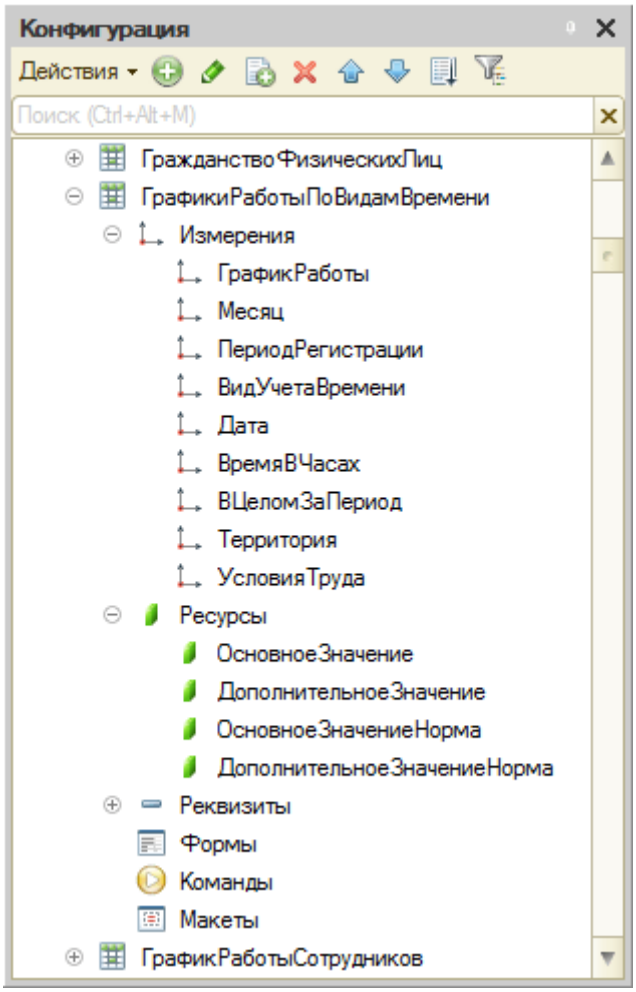
Изменение состояния регистра расчета происходит, как правило, при проведении документа. Поэтому каждая запись регистра связана с определенным документом — регистратором и номером строки этого документа. Добавление записей в регистр, их изменение и удаление возможно только одновременно для всех записей, относящихся к одному документу.

Связь с графиком времени

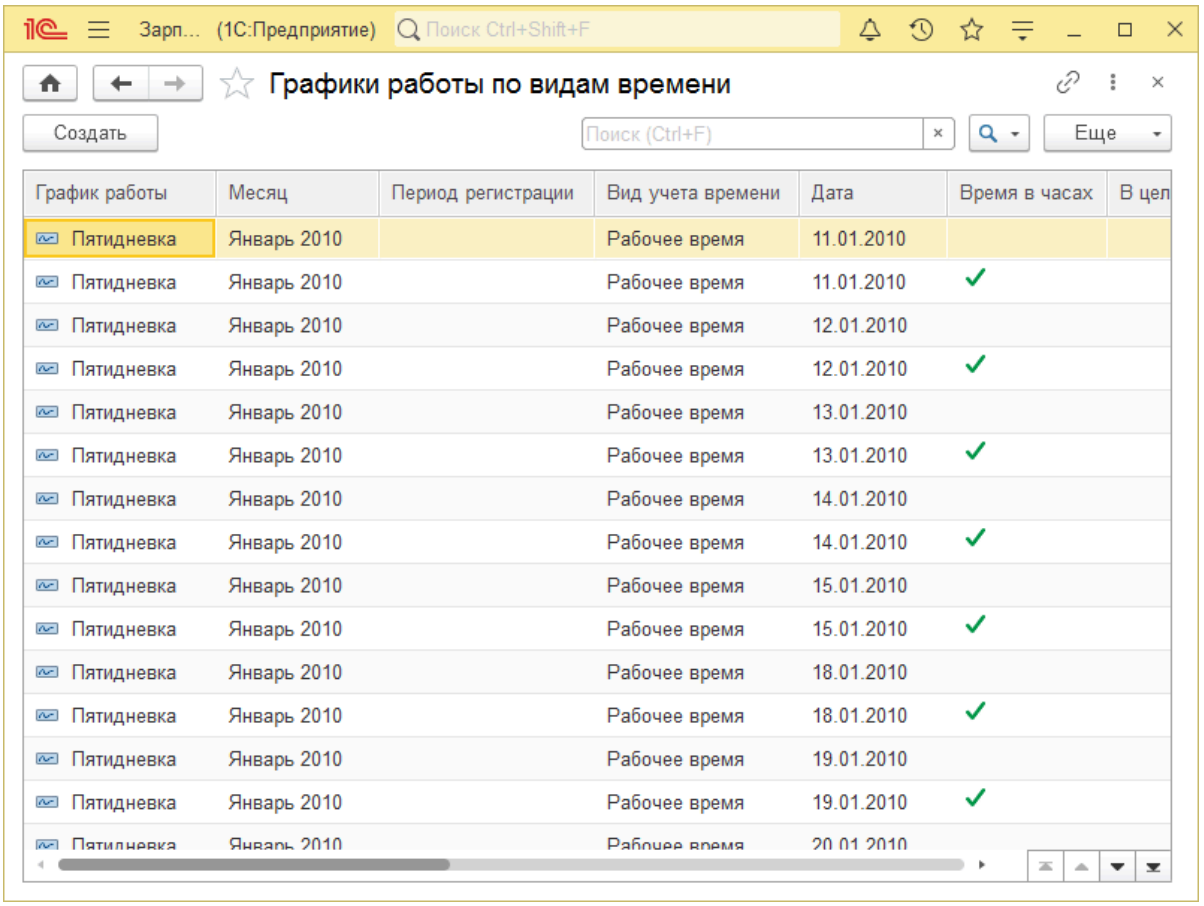
Для регистра расчета может быть указана связь с графиком времени. График времени представляет собой регистр сведений, в котором содержится временная схема исходных данных, участвующих в расчетах. Измерениями этого графика могут быть, например, график работы и дата, а ресурсом — количество рабочих часов в этой дате. Тогда можно будет связать запись регистра расчета с каким-либо конкретным графиком работы и в дальнейшем,

средствами встроенного языка получать информацию о количестве рабочих часов, необходимую для выполнения расчетов.

Например, график времени, имеющий следующую структуру:

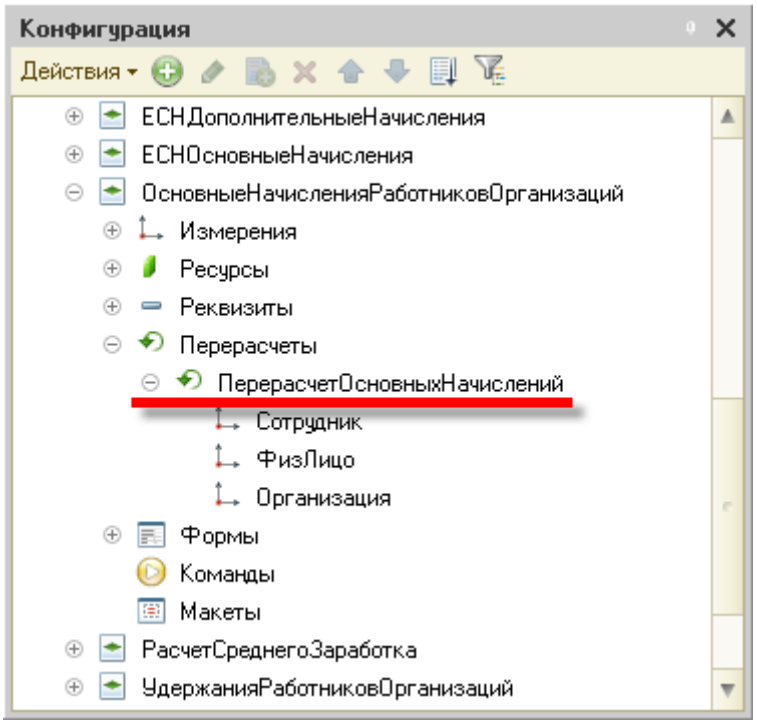


может содержать информацию о количестве рабочих часов:



Перерасчеты

Регистр расчета может включать в себя специальные объекты — **Перерасчеты**:



В этих объектах система будет хранить информацию о том, какие записи регистра расчета утратили свою актуальность и подлежат перерасчету в результате работы механизмов зависимости по базовому периоду и вытеснения по периоду действия.

Уникальность записей

Система обеспечивает контроль уникальности записей, хранящихся в регистре расчета. Поэтому в регистре расчета не может находиться двух записей, относящихся к одной и той же строке одного и того же документа.

Механизмы, реализуемые регистром расчета

Вытеснение по периоду действия

Механизм вытеснения по периоду действия позволяет рассчитывать фактический период действия записи регистра расчета на основании анализа других записей, содержащихся в регистре.

В общем случае, запись регистра расчета содержит две даты, определяющие период, на который распространяется действие данной записи. Этот период называется периодом действия записи. Однако, если вид расчета, к которому относится данная запись, может быть вытеснен другим видом расчета, то период действия данной записи является лишь «запрашиваемым» периодом, то есть «мы хотим, чтобы запись действовала в этом периоде». Реально, фактический период действия данной записи можно будет определить лишь после анализа всех записей видов расчета, которые вытесняют данный вид расчета по периоду действия. Фактический период действия будет представлять собой набор периодов, являющихся подмножеством исходного периода действия данной записи. Если не будет найдено ни одной записи, вытесняющей данную по периоду действия, то фактический период действия данной записи будет равен ее периоду действия. Другим крайним случаем вытеснения по периоду действия является полное вытеснение данной записи другими записями. В этом случае фактический период действия записи будет отсутствовать.

Каждая запись регистра расчета содержит вид расчета, к которому она относится. Чтобы определить, какие записи должны вытеснять данную запись по периоду действия, регистр расчета использует связь с планом видов расчета, в котором описано взаимное влияние видов расчета друг на друга. Использование этой связи позволяет регистру расчета определять фактический период действия каждой записи.

Зависимость по базовому периоду

Механизм зависимости по базовому периоду позволяет получать значение базы для записи регистра расчета на основании анализа других записей, содержащихся в регистре.

База представляет собой числовое значение, которое должно быть использовано для расчета результата данной записи. База рассчитывается путем анализа результатов расчета других записей, от которых данная запись зависит по базовому периоду. Таким образом, в общем случае, запись регистра расчета содержит две даты, определяющие период, в котором нужно анализировать записи видов расчета, от которых данный вид расчета зависит по базе — базовый период. Использование связи с планом видов расчета позволяет регистру расчета определять виды расчета, от которых данный вид расчета зависит по базовому периоду.

Регистр расчета поддерживает два вида зависимости по базовому периоду:

- зависимость по периоду действия;
- зависимость по периоду регистрации.

В случае зависимости по периоду действия для получения базы будут выбираться те записи, для которых найдено пересечение их фактического периода действия с базовым периодом данной записи. Значение базы, которая будет получена от конкретной влияющей записи, в общем случае не равно результату, который содержит эта запись. База будет рассчитана пропорционально тому, какую часть от фактического периода влияющей записи составляет перекрывающийся, с указанным базовым периодом, участок. При этом будут использованы данные графика, связанного с этой записью.

В случае зависимости по периоду регистрации для получения базы будут выбираться результаты расчета тех записей, которые попадают в базовый период данной записи значением своего поля «Период регистрации».

Наиболее сложным вариантом зависимости по базовому периоду является случай, когда для вида расчета данной записи установлено свойство «Период действия является базовым периодом». Это свойство означает, что в качестве базового периода данной записи будет использоваться не базовый период, который указан в соответствующих полях записи, а фактический период действия записи, получаемый в результате работы механизма вытеснения по периоду действия и являющийся, в общем случае, набором некоторых периодов.

Формирование записей перерасчета

Механизм формирования записей перерасчета отслеживает факт появления в регистре записей, влияющих на результат расчета уже существующих записей. Возможность влияния новых записей на существующие определяется в результате анализа взаимного влияния видов расчета и на основании работы механизмов вытеснения по периоду действия и зависимости по базовому периоду.

Результатом работы механизма формирования записей перерасчета является набор записей перерасчета, содержащий информацию о том, какие записи регистра должны быть рассчитаны заново (перерасчитаны).

Функциональные возможности регистра расчета

Основными функциональными возможностями, которые предоставляет регистр расчета разработчику, являются:

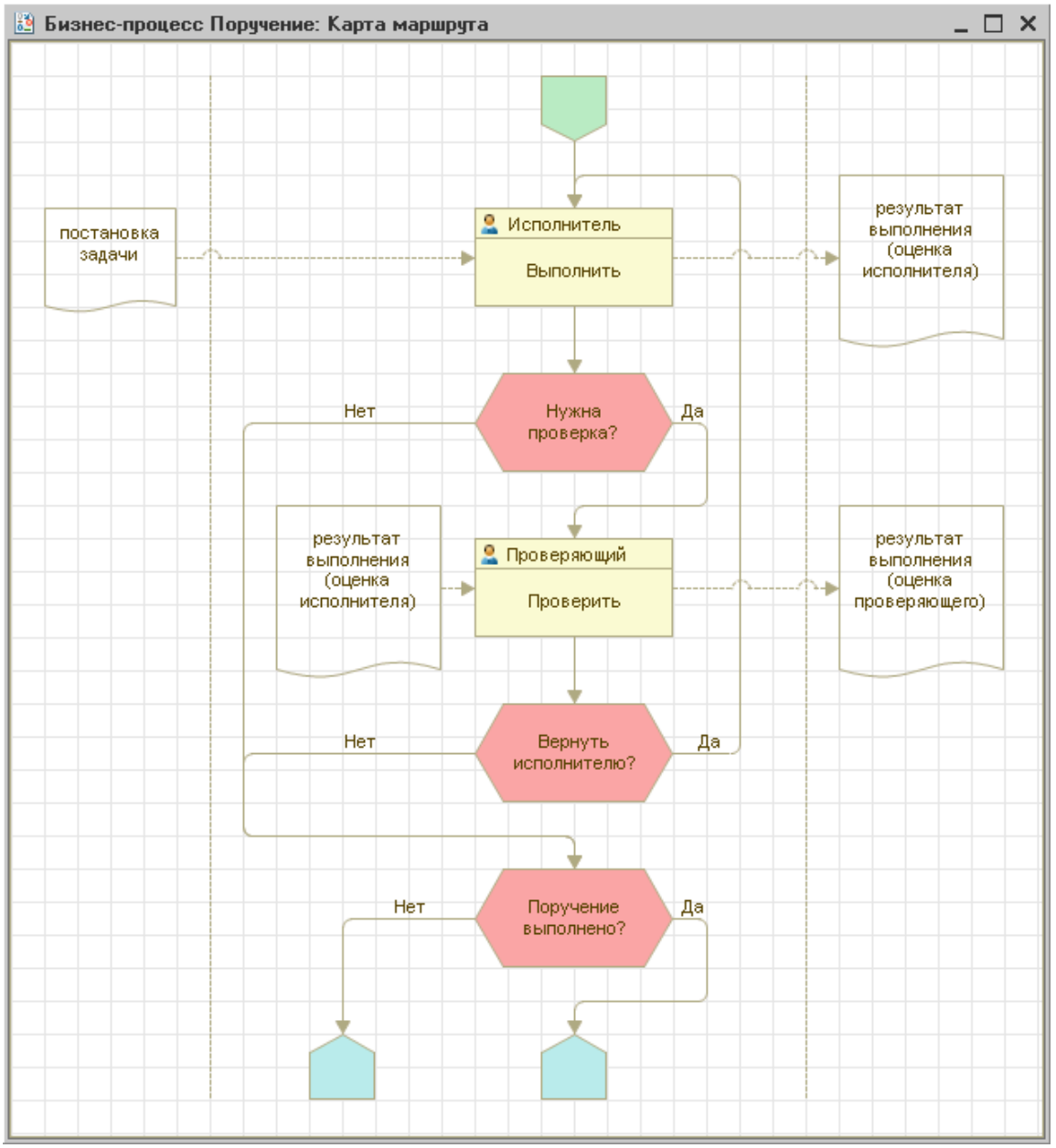
- выбор записей в заданном интервале по заданным критериям;
- выбор записей по регистратору;
- получение значения базы для записей регистра, удовлетворяющих заданному отбору;
- получение данных графика для записей регистра, удовлетворяющих заданному отбору;
- получение данных о записях, подлежащих перерасчету;
- чтение, изменение и запись набора записей в регистр.

Бизнес-процесс

Бизнес-процессы — это прикладные объекты конфигурации. Они описывают бизнес-логику в карте маршрута и управляют жизненным циклом созданных бизнес-процессов (экземпляров) от момента старта до момента завершения. Необходимым свойством описания бизнес-процесса является связь с задачей, которая задает систему адресации и позволяет проектировать карты маршрута в соответствии с поддерживаемой в прикладном решении организационной структурой.

Карта маршрута

Логика бизнес-процесса (взаимосвязь и последовательность обхода точек маршрута, условные переходы и пр.) наглядно описывается в виде карты маршрута, которая позволяет визуально описывать маршрут бизнес-процесса в виде связного графа и позволяет легко описывать алгоритмы условных переходов, и реакцию бизнес-процесса на различные события:

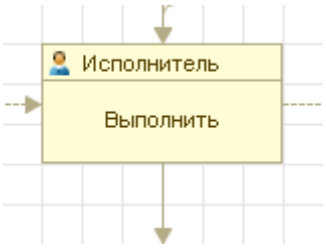


Для создания карты маршрута бизнес-процесса могут быть использованы все оформительские возможности, предоставляемые графической схемой.

При работе пользователя с системой предусмотрена возможность отображения актуальной карты маршрута для конкретных экземпляров бизнес-процессов с учетом пройденных и активных точек маршрута.

Точка действия

Точки этого вида описывают отдельную операцию (единицу работы), соответствующую определенному этапу (шагу) в жизненном цикле бизнес-процесса:



Точка действия содержит информацию кто и что должен сделать на данном этапе, например:

- бухгалтер — принять оплату наличными;
- кладовщик — выдать товар со склада по накладной;
- системный администратор — зарегистрировать нового сотрудника в сети и электронной почте.

Формулировка задания обычно составляется из глаголов и отглагольных существительных. Исполнитель может определяться персонально (Иванов), или с учетом ролевой маршрутизации («Кладовщик», «Руководитель отдела продаж»). При переходе бизнес-процесса на точку действия он автоматически формирует задачи, устанавливая в них реквизиты адресации, предусмотренные в точке действия.

Групповая и коллективная маршрутизация

- групповая — данное действие должны выполнить все члены группы. При этом формируется несколько задач (по количеству сотрудников, входящих в группу) и бизнес-процесс ожидает завершения всех задач для продвижения дальше по маршруту. Примером групповой маршрутизации может служить задача, например, всем менеджерам предоставить ежемесячный отчет;
- коллективная — только один из членов группы должен выполнить данное действие. Формируется одна задача. Пример коллективной задачи: кладовщики — выдать товар по данной накладной со склада.

В точке действия можно описать проверку необходимых условий выполнения задачи, интерактивный диалог с пользователем при переходе далее по маршруту, и указать, например, какие документы следует открывать при активации задач, связанных с этой точкой маршрута бизнес-процесса.

Персональная и ролевая маршрутизация

Ролевая маршрутизация позволяет назначать задания не только конкретным исполнителям, но и ролям, группам, подразделениям и т. д. как это определено в прикладном решении.

Ролевая маршрутизация построена на взаимодействии объектов задача и регистра сведений. Задача определяет состав реквизитов адресации (роли,

подразделения и т. д.), а регистр сведений отражает актуальную, т. е. соответствующую текущему моменту информацию о принадлежности сотрудников ролям, подразделениям, рабочим группам и т. д. и поэтому называется регистром адресации.

Регистр сведений можно использовать для реализации механизмов замещения или учета отсутствия сотрудников. Например, если в регистре сведений указано, что роль главного бухгалтера исполняет Иванов, и Иванов уходит в отпуск и его обязанности передаются Петрову, то запись в регистре сведений меняется так, чтобы роль главного бухгалтера исполнял Петров. По возвращении Иванова из отпуска запись в регистре сведений восстанавливается.

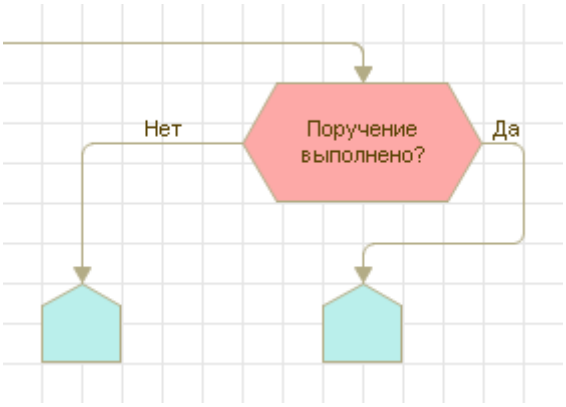
Условная маршрутизация

Механизм бизнес-процессов в 1С:Предприятии 8 позволяет автоматизировать не только последовательные цепочки работ, но и осуществлять условную маршрутизацию, управляя выбором заранее предусмотренных маршрутов, на основе описанной разработчиком карты маршрута.

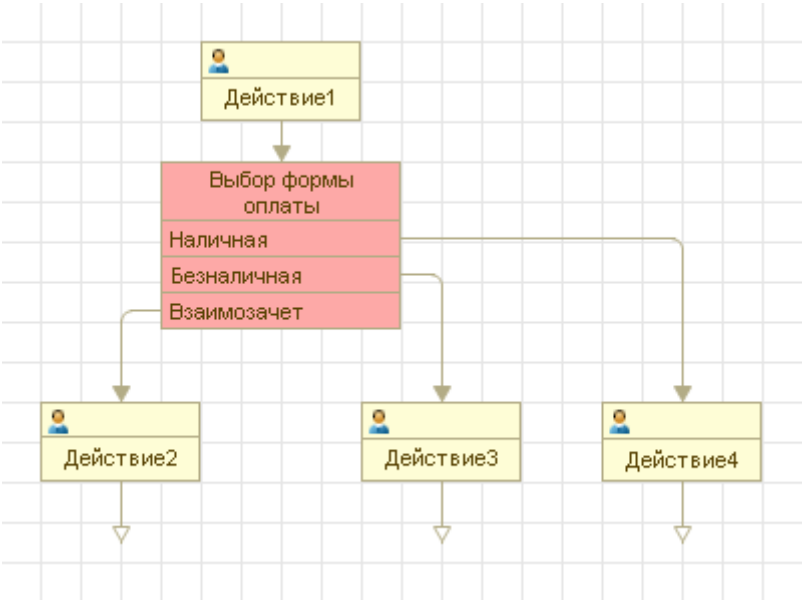
Условная маршрутизация обеспечивается точками маршрута двух видов:

- условный переход;
- выбор варианта.

Условный переход предоставляет возможность выбора одного из двух возможных вариантов дальнейшего маршрута (да/нет, больше/меньше, запрещено/разрешено и т. д.):



Точка выбора варианта предоставляет возможность выбора дальнейшего маршрута из неограниченного количества вариантов:



Использование в прикладных решениях

Для использования механизма бизнес-процессов не требуется изменения готовых решений, или эти изменения несущественны. Например, внесение незначительных изменений в конфигурацию позволит бизнес-процессам должным образом реагировать на запись важных для них объектов базы данных. В этом случае пользователи смогут работать привычным образом, не пользуясь списком заданий и не подозревая о существовании бизнес-процессов. Однако выполняемые ими операции будут приводить к продвижению соответствующих бизнес-процессов дальше по маршруту.

Задача

Задачи — это прикладные объекты конфигурации. Они предназначены для учета заданий и описывает способ их распределения по исполнителям, с учетом организационной структуры предприятия. Адресация заданий сотрудникам определяется реквизитами, в которых можно предусмотреть многомерную ролевую маршрутизацию, например, по ролям, рабочим группам, подразделениям, помещениям, филиалам и т. д.

Задачи являются «движущей силой» механизма бизнес-процессов. При выполнении задачи породивший ее бизнес-процесс осуществляет переход на следующую точку маршрута в соответствии с картой маршрута. При этом задачи имеют самостоятельную прикладную ценность как список заданий, назначенных конкретным исполнителям напрямую, или посредством ролевой маршрутизации, и могут использоваться отдельно от бизнес-процессов.

При формировании списка задач для конкретного сотрудника, используется регистр сведений, который обеспечивает нахождение соответствий роль-сотрудник согласно системе адресации, настроенной в задаче. Как правило, для всех бизнес-процессов реализуется единый список задач.

Внешний источник данных

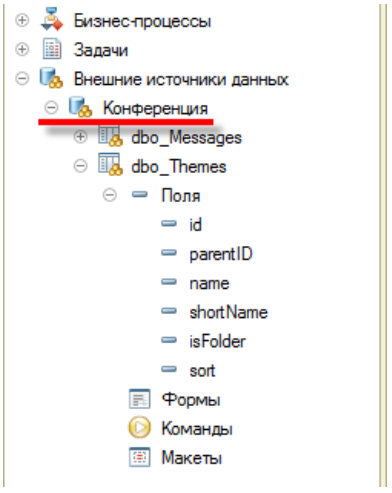
Внешние источники данных — это прикладные объекты конфигурации. Они позволяют работать с внешними базами данных, не основанными на 1С:Предприятии. Благодаря этим объектам конфигурации информацию из внешних баз можно использовать внутри прикладного решения так же, как будто бы она хранится в самой информационной базе.

Внешний источник может получать данные из ODBC-источников в операционных системах Windows и Linux, причем при работе с СУБД Microsoft SQL Server, IBM DB2, PostgreSQL и Oracle Database обеспечиваются полные возможности языка запросов. Кроме этого внешние источники данных позволяют подключить к прикладному решению многомерные источники данных, такие как:

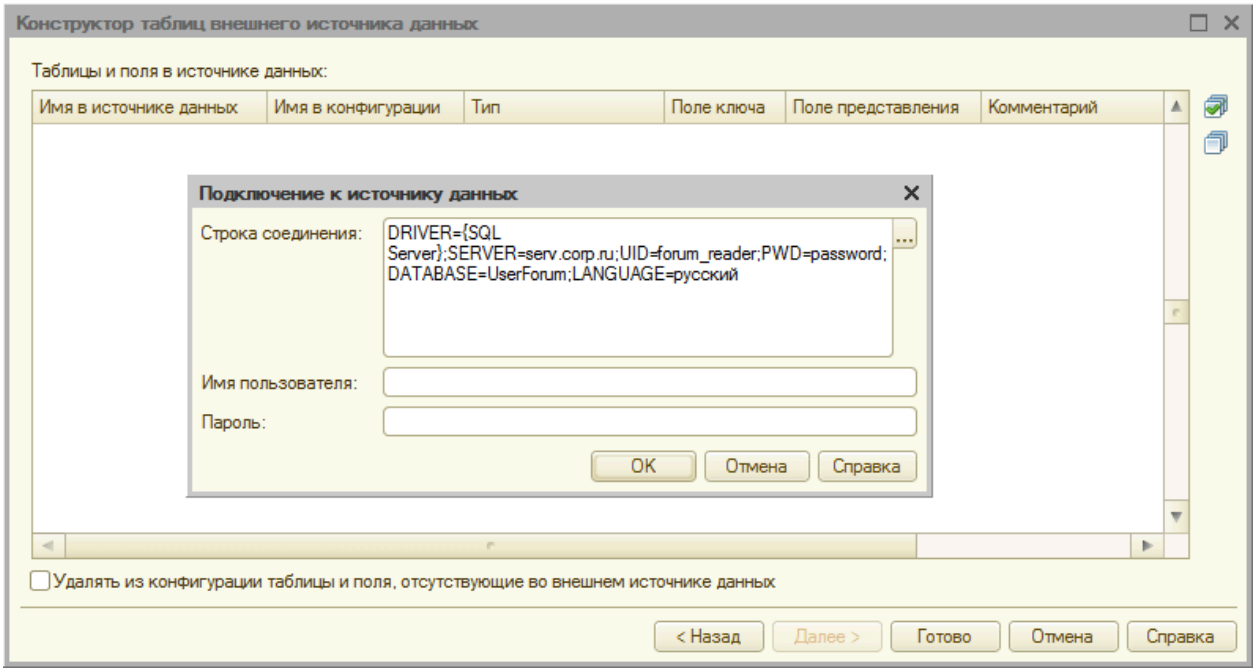
- Microsoft Analysis Services;
- Oracle Essbase;
- IBM InfoSphere Warehouse.

При работе с другими СУБД некоторые конструкции языка запросов могут не работать, т. к. они ограничены возможностями того ODBC-драйвера, который используется.

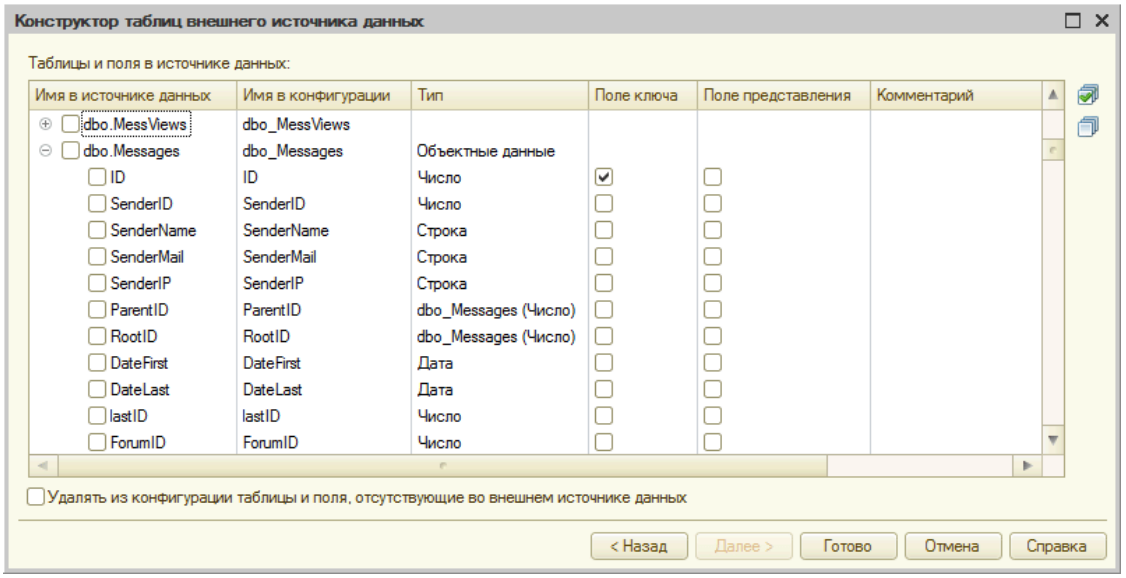
Внешний источник данных состоит из набора таблиц.



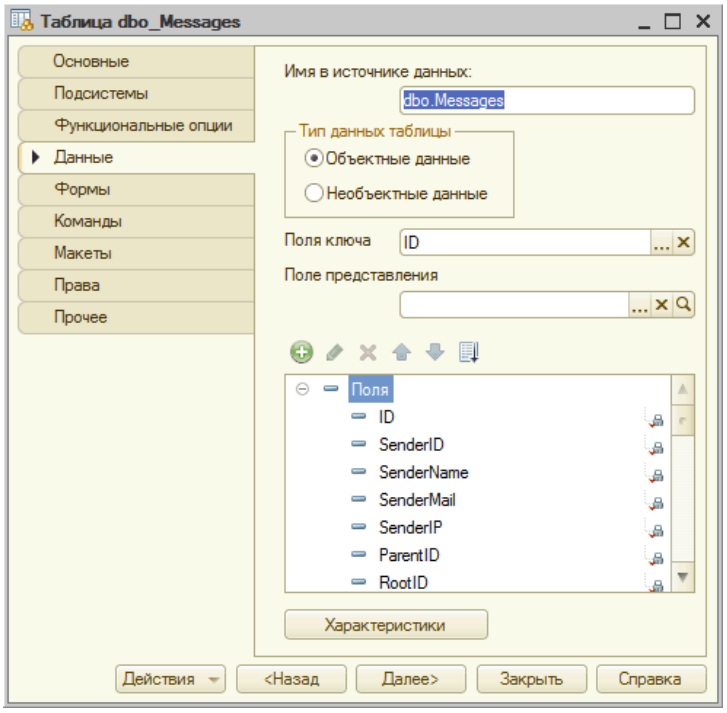
Таблицы внешнего источника данных можно описать вручную, или с помощью конструктора.



Конструктор, используя строку соединения, может подключится к внешней базе данных и предоставить список таблиц, доступных для добавления во внешний источник.

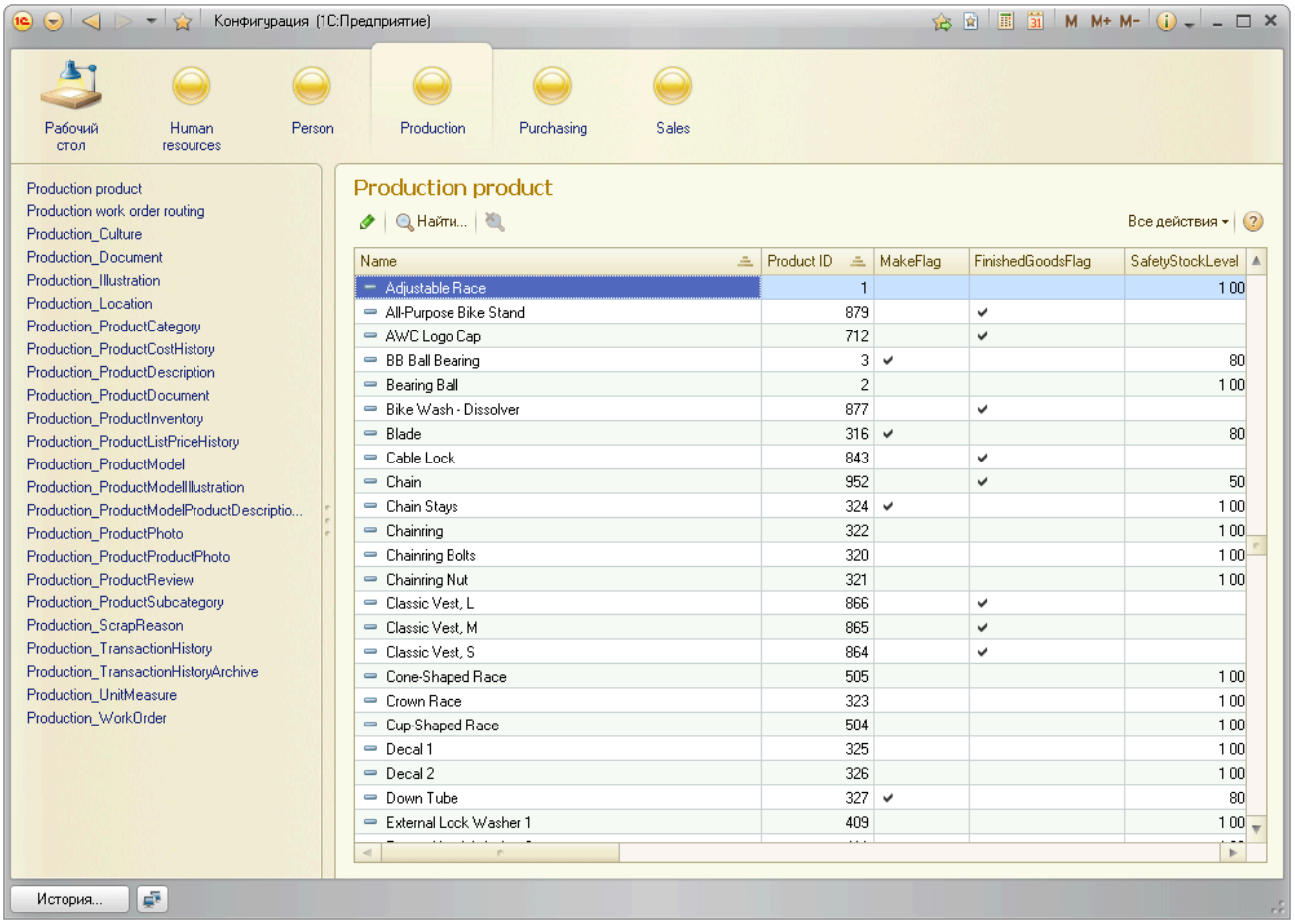


Каждая таблица, по сути, является самостоятельным объектом конфигурации. Она содержит набор полей, для неё могут быть определены формы, созданы команды и заданы макеты.

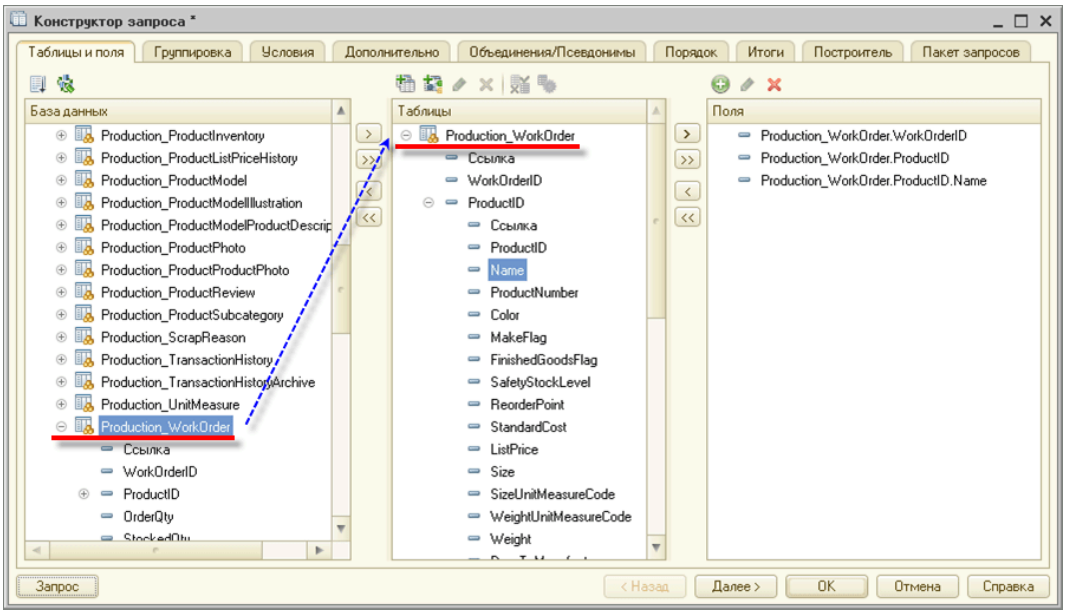


Таблицы внешних источников используются в конфигурации точно так же, как и таблицы любых других объектов конфигурации.

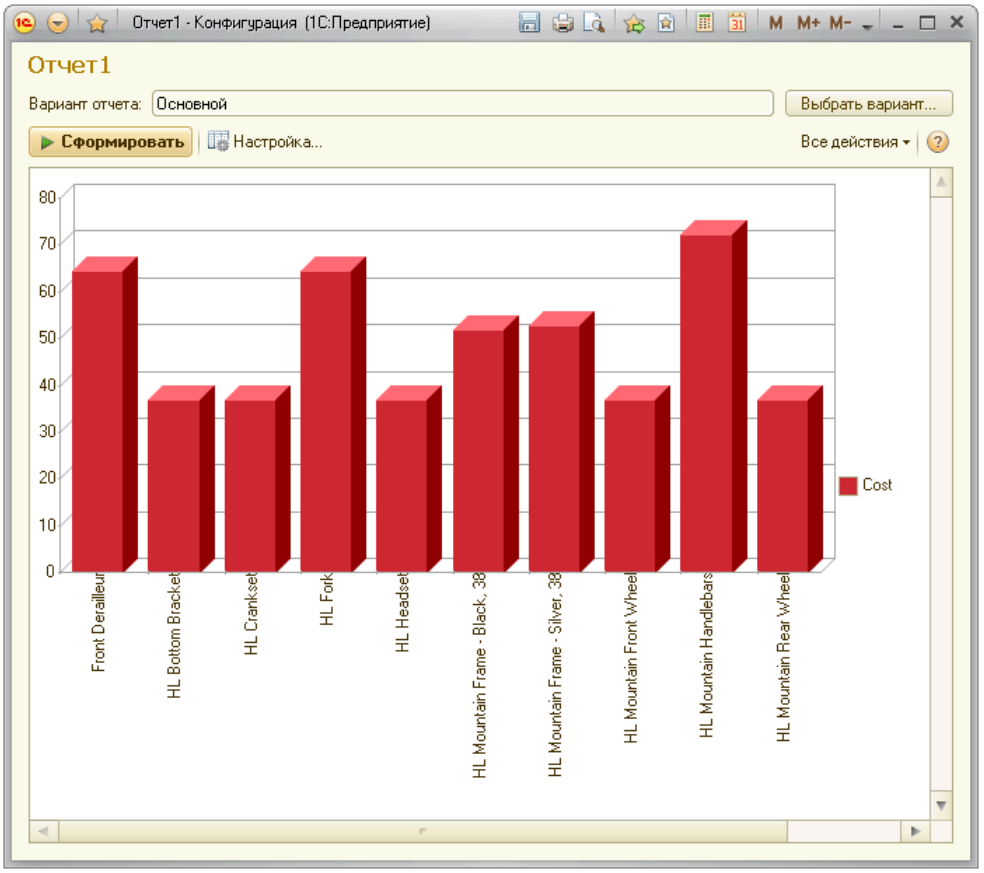
Например, платформа автоматически генерирует для них формы, или же форму разработчик может создать самостоятельно.



К данным этих таблиц можно обращаться с помощью языка запросов.



Эти таблицы можно использовать в отчётах на системе компоновки данных.



Также на элементы этих таблиц можно ссылаться как на любые другие данные объектного типа. Например, как на элементы справочников или документы.

