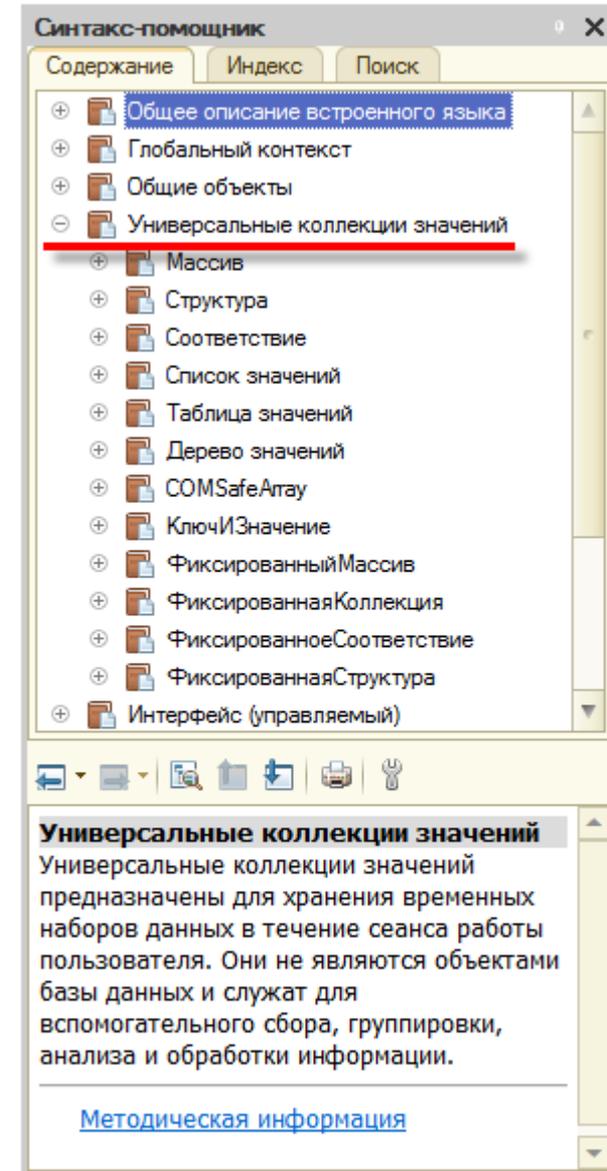


Типы данных, определенные на уровне платформы

- **Примитивные типы данных:** Строка, Число, Дата, Булево и др.
- **Универсальные коллекции значений:** Массив, Структура, СписокЗначений и др.
- **Интерфейсные типы** – типы, связанные с работой **форм** и их **элементов**.
- **Общие типы** - специфические типы: ТабличныйДокумент, ХранилищеЗначения, УникальныйИдентификатор и др.



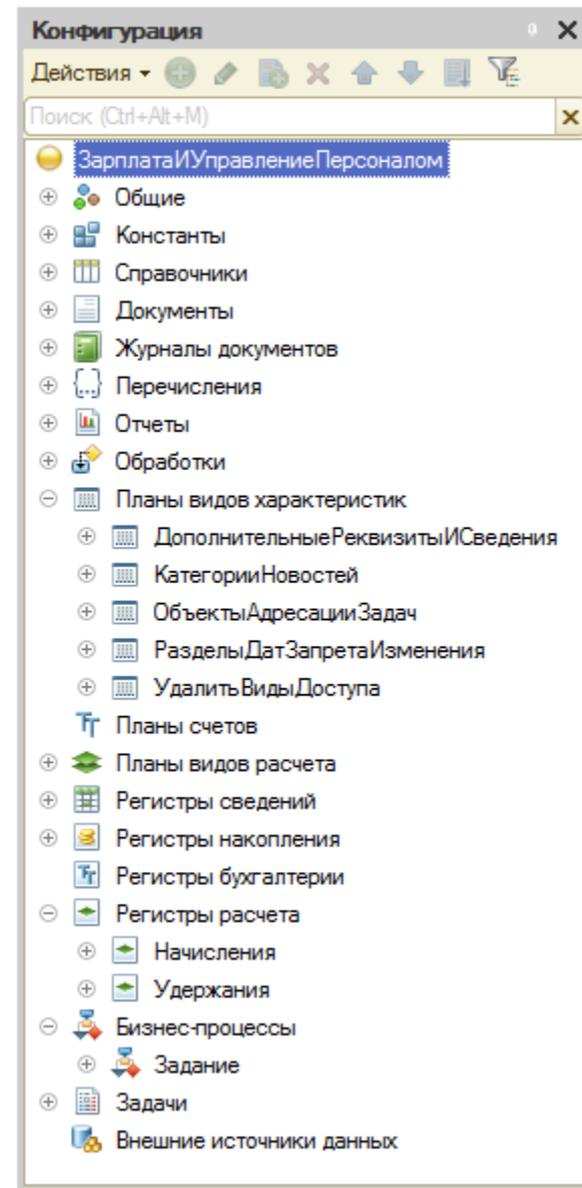
Типы данных, образуемые в прикладном решении

Появление новых типов данных в прикладном решении связано с использованием **прикладных объектов конфигурации**.

«прикладные»?!

На уровне платформы поддерживается несколько классов (шаблонов) прикладных объектов:

Справочники, Документы, Регистры сведений, Отчеты, Обработки и др.



Объекты конфигурации

<http://v8.1c.ru>, [Толковый словарь 1С:Предприятия 8](#) :

Объекты конфигурации - это составные элементы, "детали", из которых складывается любое прикладное решение.

Они представляют собой **проблемно-ориентированные объекты**, поддерживаемые на уровне технологической платформы.

Задача разработчика в том, чтобы **собрать из этих объектов, как из конструктора, необходимую структуру прикладного решения и затем описать специфические алгоритмы функционирования и взаимодействия этих объектов, отличающиеся от их типового поведения.**

Алгоритмы + структуры данных = программы.

Никлаус Вирт, 1976

Различные виды объектов конфигурации различным образом обеспечивают решение **одной или нескольких** задач из следующего списка:

- 1. ввод;**
- 2. хранение;**
- 3. обработка;**
- 4. отображение данных.**

Основные инструменты конфигурирования

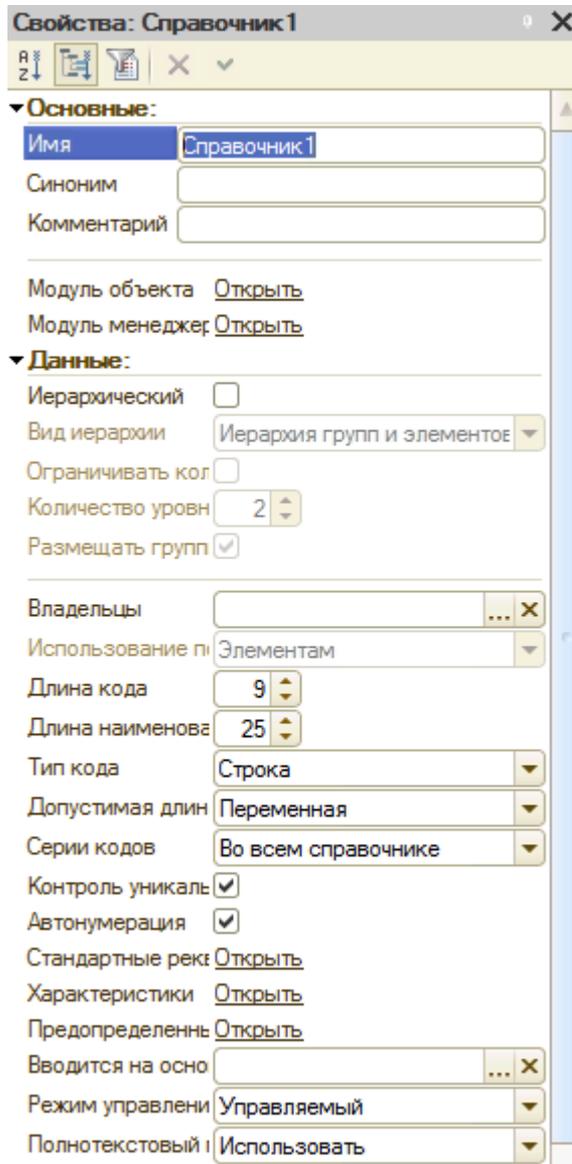
Окно редактирования объекта

The screenshot shows a software window titled 'Справочник Товары' (Goods Reference). On the left is a tree view with the following items: Основные (selected), Подсистемы, Функциональные опции, Иерархия, Владельцы, Данные, Нумерация, Формы, Команды, Макеты, Ввод на основании, Права, Обмен данными, and Прочее. The main area contains a form with the following fields: 'Имя:' (Name) with the value 'Товары'; 'Синоним:' (Synonym) with the value 'Товары'; 'Комментарий:' (Comment); 'Представление объекта:' (Object representation) with the value 'Товар'; 'Расширенное представление объекта:' (Extended object representation); 'Представление списка:' (List representation) with the value 'Товары'; 'Расширенное представление списка:' (Extended list representation); and 'Пояснение:' (Explanation). At the bottom of the window are buttons for 'Действия' (Actions), '<Назад' (Back), 'Далее>' (Next), 'Закреть' (Close), and 'Справка' (Help).

Появляется после создания нового объекта.

Вызывается двойным щелчком в дереве конфигурации.

Палитра свойств



Свойства: Справочник 1

▼ Основные:

Имя

Синоним

Комментарий

Модуль объекта [Открыть](#)

Модуль менеджер [Открыть](#)

▼ Данные:

Иерархический

Вид иерархии

Ограничивать кол

Количество уровн

Размещать групп

Владельцы

Использование п

Длина кода

Длина наименова

Тип кода

Допустимая длин

Серии кодов

Контроль уникаль

Автонумерация

Стандартные рек [Открыть](#)

Характеристики [Открыть](#)

Предопределенн [Открыть](#)

Вводится на осно

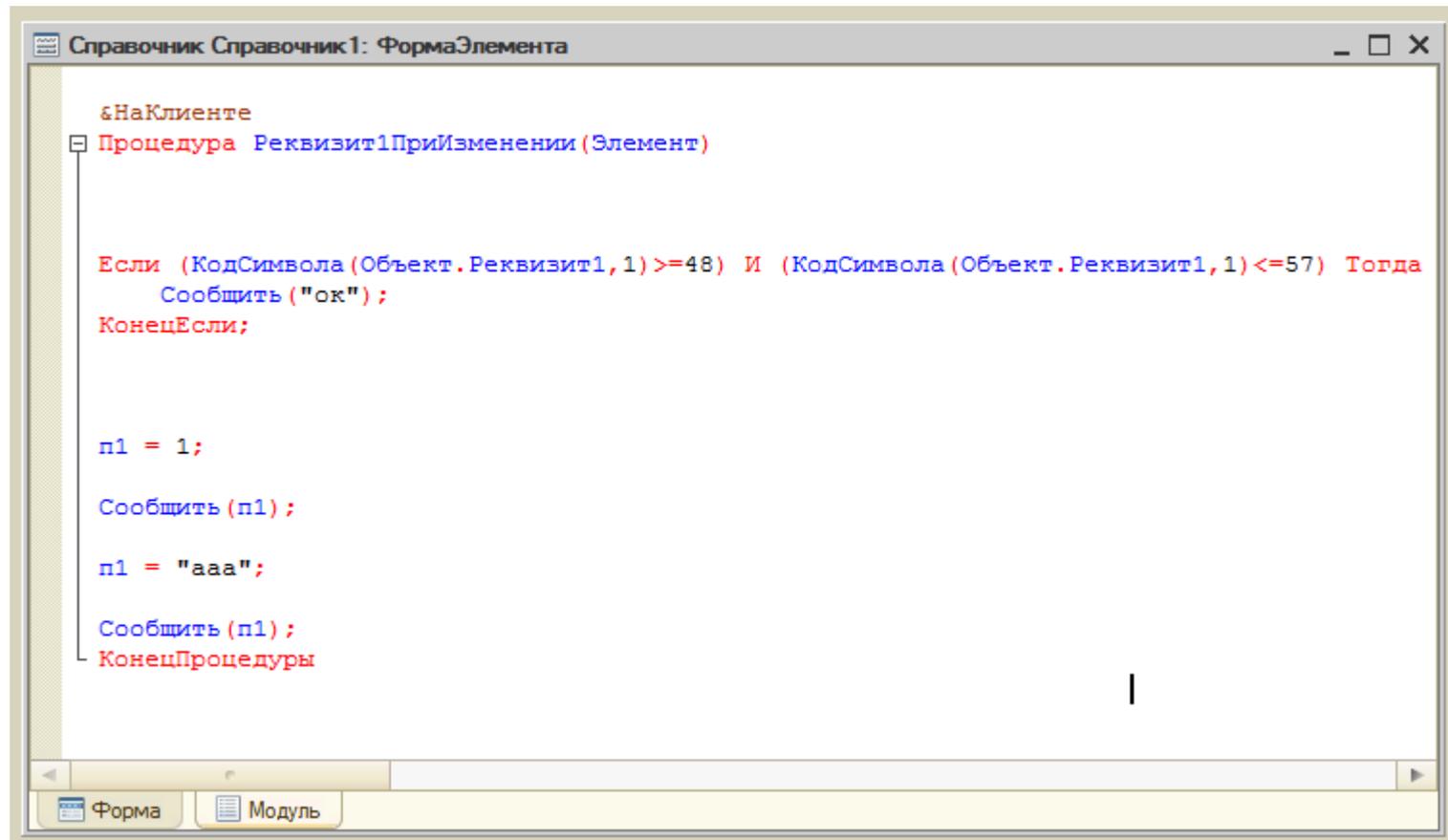
Режим управлени

Полнотекстовый

*Вызывается
соответствующим
пунктом
контекстного меню.*

Каждый объект имеет свойство «**Имя**» (строка **специального формата**), которое используется для программного обращения к объекту; и свойство «**Синоним**» (**произвольная** строка), которое отображается пользователю .

Текстовый редактор (редактор модуля)



```
Справочник Справочник1: ФормаЭлемента

&НаКлиенте
Процедура Реквизит1ПриИзменении(Элемент)

Если (КодСимвола(Объект.Реквизит1,1)>=48) И (КодСимвола(Объект.Реквизит1,1)<=57) Тогда
    Сообщить ("ок");
КонецЕсли;

p1 = 1;

Сообщить (p1);

p1 = "aaa";

Сообщить (p1);
КонецПроцедуры
```

Встроенный язык

позволяет разработчику описывать собственные алгоритмы функционирования прикладного решения.

Основные свойства:

✓ **Динамическая типизация** – определение типа переменной в момент присваивания значения.

«неопределенное значение»?!

✓ Правила доступа к свойствам и методам объектов **как в объектно-ориентированных языках** (без возможности объявлять новые типы).

✓ **Событийная ориентированность**: выполнение модулей происходит при возникновении определенных событий (например, открытие или закрытие форм, изменение содержимого элементов формы, создание или запись элементов справочника или документов, начало работы пользователя и т.п).

Виды программных модулей:

модуль управляемого приложения;
общие модули;
модули прикладных объектов;
модули менеджеров объектов;
модули форм и т.п.

Каждый модуль связан с остальной частью конфигурации, эта связь называется **контекстом выполнения модуля**.

Доступность объектов, методов и свойств встроенного языка зависит от контекста выполнения.

Объект может быть доступен

на сервере,

на любом клиенте,

в тонком клиенте,

в веб-клиенте,

в толстом клиенте,

во внешнем соединении.

Инструкции препроцессора и директивы компиляции

предназначены для того, чтобы **оставить** в скомпилированном модуле только то, что действительно должно присутствовать в **ТОМ** или **ином контексте**.

Директивы компиляции

действуют на структурные единицы программного кода – процедуры, функции, переменные.

Синтаксис:

`&<Директива >`

`<Конструкция языка>`

Русское имя	Английское имя
НаКлиенте	<code>AtClient</code>
НаСервере	<code>AtServer</code>
НаСервереБезКонтекста	<code>AtServerNoContext</code>
НаКлиентеНаСервереБезКонтекста	<code>AtClientAtServerNoContext</code>
НаКлиентеНаСервере	<code>AtClientAtServer</code>

Инструкции препроцессора

действуют на произвольный фрагмент программного кода.

Синтаксис:

```
#Если <Логическое выражение> Тогда  
  
#ИначеЕсли <Логическое выражение> Тогда  
  
...  
  
#Иначе  
  
#КонецЕсли
```

В логическом выражении используются:

Сервер
НаСервере
Клиент
НаКлиенте
ТонкийКлиент
ВебКлиент
ВнешнееСоединение
ТолстыйКлиентУправляемоеПриложение
ТолстыйКлиентОбычноеПриложение

Редактор формы

The screenshot shows a software interface for editing a form. The window title is "Справочник Товары: ФормаЭлемента".

Left Sidebar (Form Elements):

- Форма
- Командная панель
- Родитель
- Код
- Наименование
- Вид
- Артикул
- Поставщик
- ШтрихКод
- Страницы
- ГруппаКартинки

Central Table (Requisites):

Реквизит	Используй... всегда	Тип
Объект		(СправочникОбъект.Това...
Ссылка	<input checked="" type="checkbox"/>	СправочникСсылка.Товары
Код	<input type="checkbox"/>	Строка
Наименование	<input type="checkbox"/>	Строка
Родитель	<input type="checkbox"/>	СправочникСсылка.Товары
ЭтоГруппа	<input checked="" type="checkbox"/>	Булево
ПометкаУдаления	<input checked="" type="checkbox"/>	Булево
Предопределенный	<input checked="" type="checkbox"/>	Булево
ИмяПредопределе...	<input checked="" type="checkbox"/>	Строка
Артикул	<input checked="" type="checkbox"/>	Строка

Bottom Form (Form Fields):

- Записать и закрыть (highlighted)
- Записать
- Создать на основании (dropdown)
- Печать (dropdown)
- Группа: [text field]
- Код: [text field]
- Наименование: [text field]
- Вид: Товар Услуга
- Артикул: [text field]
- Поставщик: [text field]
- Штрихкод: [text field]
- Изображение | Характеристики | Описание (tabs)

Bottom Bar: Форма | Модуль

Любая форма представляет совокупность нескольких составляющих:

- ✓ **элементов** - объектов, определяющих визуальное представление формы и осуществляющих взаимодействие с пользователем,
- ✓ **командного интерфейса** - совокупности команд, отображаемых в форме;
- ✓ **реквизитов** - объектов, данные которых форма использует в своей работе.
- ✓ **команд** - действий, которые определены в данной конкретной форме,
- ✓ **параметров** - объектов, значения которых характеризуют саму форму, используются при ее создании и остаются постоянными в процессе "жизни" формы,
- ✓ **модуля** - программы на встроенном языке, отвечающей за работу с элементами и за обработку событий;

Система прав доступа

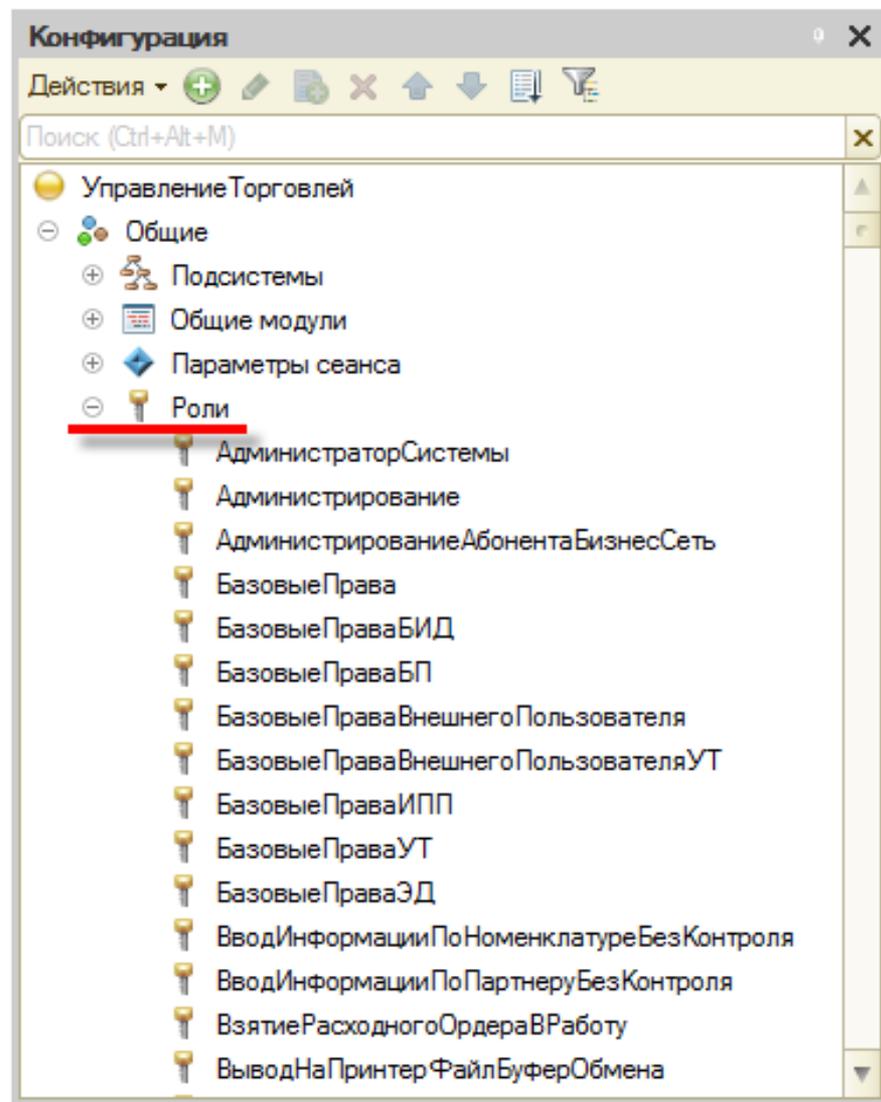
Роли - это общие объекты конфигурации.

Они предназначены для реализации ограничения прав доступа в прикладных решениях.

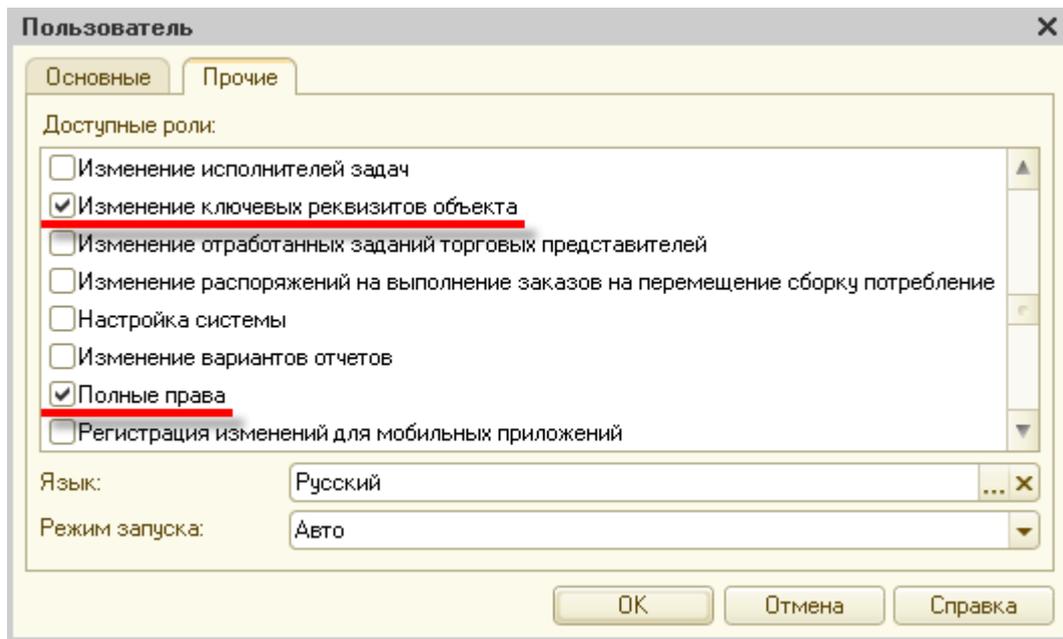
Роль в конфигурации может соответствовать

должностям или **видам**

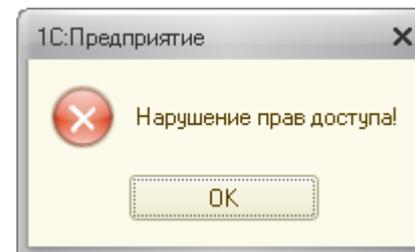
деятельности различных групп пользователей.



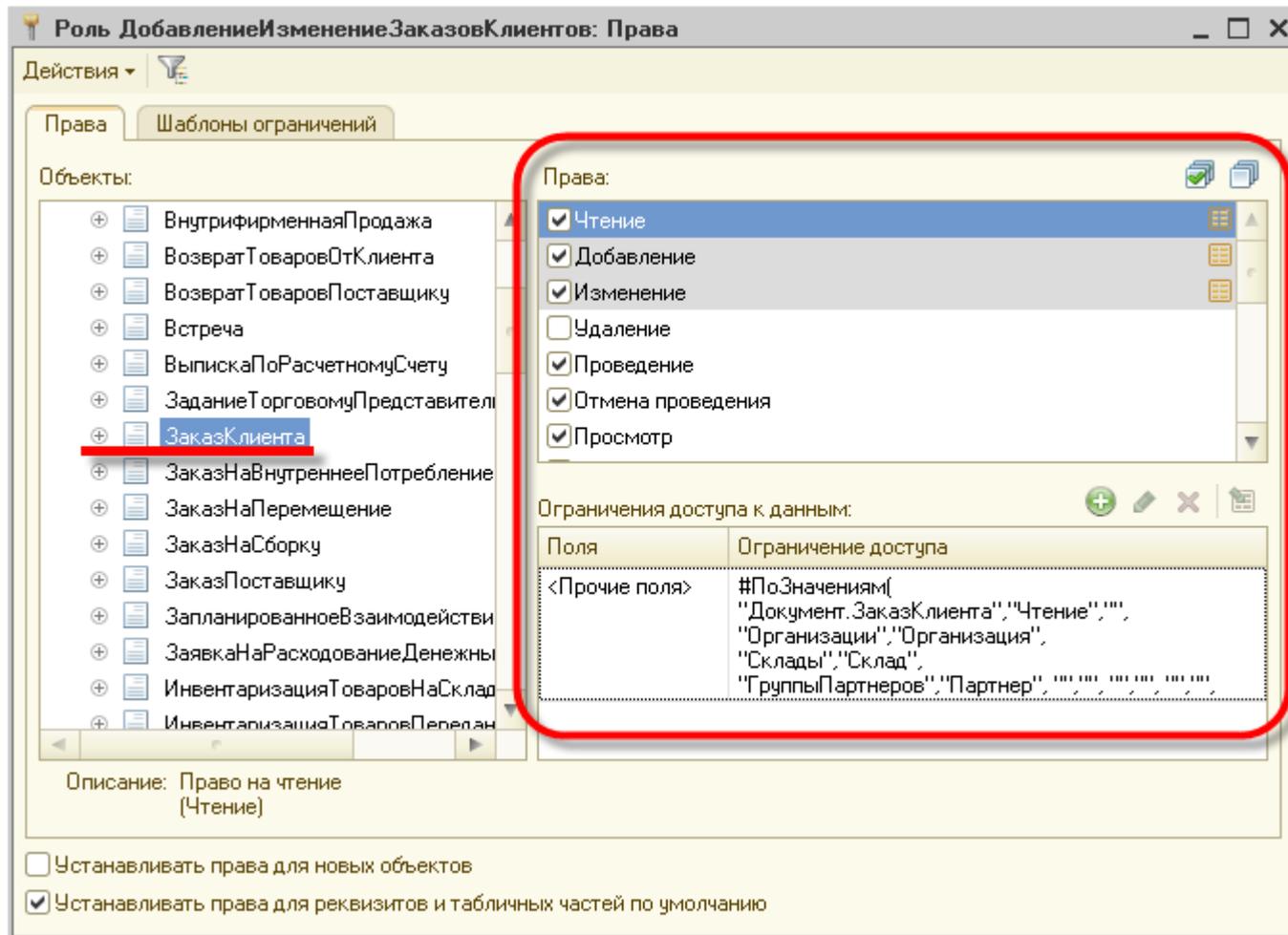
Каждому пользователю ставится в соответствие одна или несколько ролей.



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



Роль определяет, какие действия, над какими объектами метаданных может выполнять пользователь



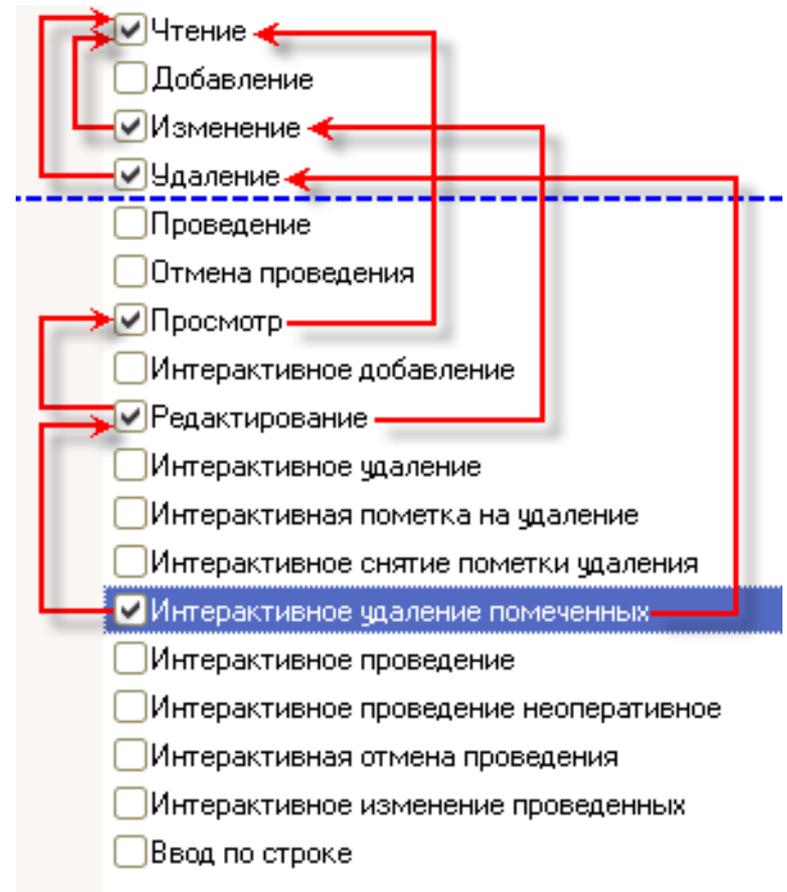
Метаданные - сведения о признаках и свойствах, характеризующих какие-либо сущности.

В 1С **метаданные** = объекты конфигурации.

Интерактивные и основные права

Основные права проверяются всегда, независимо от способа обращения к данным.

Интерактивные права описывают действия, которые могут быть выполнены пользователем интерактивно и проверяются только при выполнении интерактивных операций стандартными способами.



Механизм запросов

Используя запросы, разработчик может читать и обрабатывать данные, хранящиеся в информационной базе; изменение данных с помощью запросов невозможно.

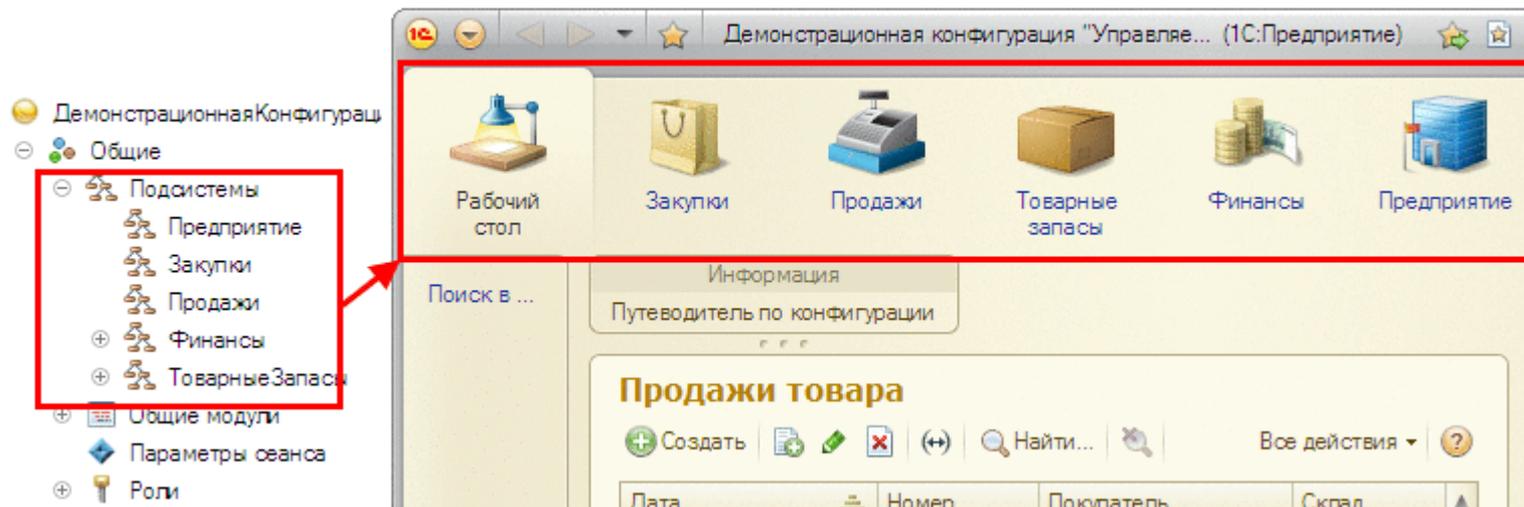
Язык запросов схож с SQL.

```
Запрос
ВЫБРАТЬ
    Контрагенты.Партнер.ОсновнойМенеджер.физическоеЛицо.Наименование
ИЗ
    Справочник.Контрагенты КАК Контрагенты
```

```
Запрос
ВЫБРАТЬ
    СвободныеОстатки.Свободно,
    Номенклатура.Наименование
ИЗ
    Справочник.Номенклатура КАК Номенклатура
    ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.СвободныеОстатки КАК СвободныеОстатки
    ПО СвободныеОстатки.Номенклатура = Номенклатура.Ссылка
```

Командный интерфейс

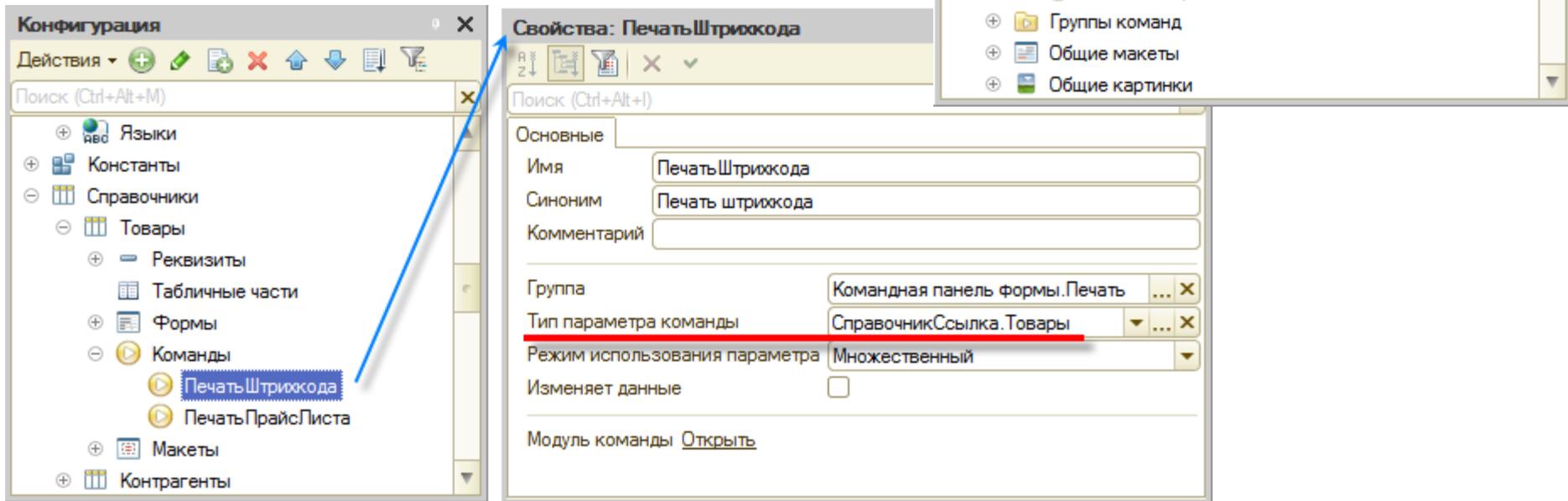
– это основное средство навигации пользователя по функциональности конфигурации. Он строится на основе подсистем.



Виды команд

Команды можно разделить

- на **независимые** глобальные команды,
- **параметризуемые** глобальные команды,
- локальные **команды формы**.



С другой стороны команды делятся

- на **стандартные** команды доступа к функциональности прикладных объектов (вызов списков справочников, документов, открытие отчетов, обработок и т. д.),
- команды, **созданные** в конфигурации.

Управление интерфейсом

Модель построения интерфейса позволяет автоматически адаптировать интерфейс для конкретного пользователя.



**Информационная модель
предметной области в 1С.**

**Проектирование структуры
метаданных.**

Модель предметной области в 1С

Константы

позволяют хранить одиночные значения, которые остаются постоянными длительное время.

Примеры:

- сведения об организации (ИНН, номера счетов и т.п.);
- настройки прикладного решения (адреса почтовых серверов, каталоги для сохранения и загрузки файлов и т.п.);
- параметры учета (валюта управленческого учета, валюта регламентированного учета и т.п.).

Модель предметной области в 1С

Перечисления

предназначены для описания наборов постоянных значений, не изменяемых пользователем в процессе работы.

Примеры:

- пол физического лица (мужской или женский);
- ставки НДС (18%, 10%, Без НДС);
- виды почтовых протоколов (POP3, SMTP, IMAP4).

Модель предметной области в 1С

Справочники

предназначены для хранения сведений о множестве однородных объектов перечень которых может изменяться в процессе работы.

Примеры:

- сотрудники;
- клиенты;
- товары.

Модель предметной области в 1С

Документы

отражают некоторые события предметной области (могут иметь названия печатных форм, но это необязательно).

Примеры:

- приказ о приеме на работу;
- продажа товара;
- расходная накладная.

Модель предметной области в 1С

Регистры

служат для накопления оперативных данных и быстрого получения сводной информации. Сведения из регистров используются для формирования отчетов.

Виды:

- Регистры **сведений**;
- Регистры **накоплений**;
- Регистры **бухгалтерии**;
- Регистры **расчета**.

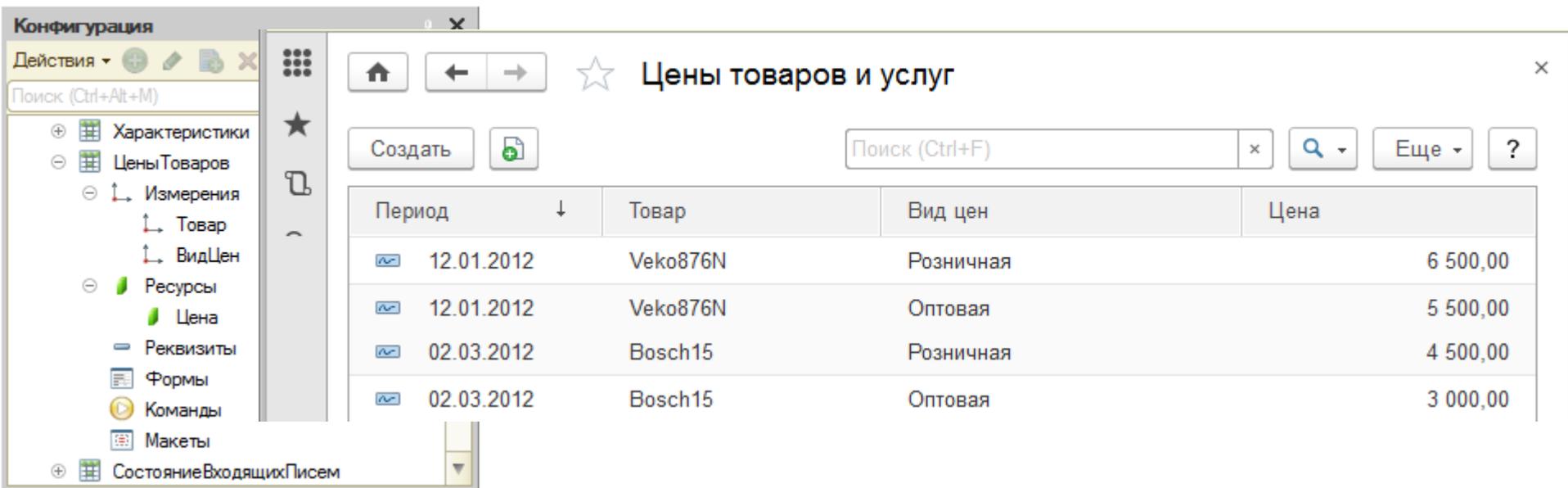
Модель предметной области в 1С

Регистры сведений

позволяют хранить в прикладном решении произвольные данные в разрезе нескольких измерений. Могут быть периодическими, что позволяет отслеживать историю изменений.

Примеры:

- цены товаров (в разрезе товаров и видов цен);
- должности сотрудника (на период).



Конфигурация

Действия

Поиск (Ctrl+Alt+M)

Характеристики

Цены Товаров

Измерения

Товар

ВидЦен

Ресурсы

Цена

Реквизиты

Формы

Команды

Макеты

СостояниеВходящихПисем

Цены товаров и услуг

Создать

Поиск (Ctrl+F)

Период	Товар	Вид цен	Цена
12.01.2012	Veko876N	Розничная	6 500,00
12.01.2012	Veko876N	Оптовая	5 500,00
02.03.2012	Bosch15	Розничная	4 500,00
02.03.2012	Bosch15	Оптовая	3 000,00

Модель предметной области в 1С

Регистры накопления

позволяют "накапливать" числовые данные в разрезе нескольких измерений. Измерения регистра описывают разрезы, в которых хранится информация, а в ресурсах регистра накапливаются нужные числовые данные.

Примеры:

- информация об остатках товаров в разрезе номенклатуры и склада;
- информация об объемах продаж в разрезе номенклатуры и подразделения.

Механизмы реализации учета

Проведение документов

- ✓ Проведение – это изменение состояния документа, подтверждающее, что **событие** которое отражает документ **произошло**.
- ✓ **В процессе** проведения происходит **изменение** связанной с данным документом **информации**, отражающей хозяйственную деятельность предприятия.
- ✓ **После** проведения **редактирование** документа **не допускается**.

Под изменением информации при проведении обычно понимают добавление записей в регистры, но возможны и другие действия.

Например, при обработке проведения документа "Приказ о приеме на работу" работает процедура-обработчик, которая в справочник «Сотрудники» добавит нового сотрудника.

Механизмы реализации учета

Движения документов

- **записи**, создающиеся **при проведении** документа **в регистрах**, для которых документ является «**регистратором**».

Создание движений определяется алгоритмом процедуры-обработчика события «**ОбработкаПроведения**».

Для облегчения написания таких процедур существует «**конструктор движений**».