

Информационная система

ГОСТ Р 53622-2009 Информационные технологии (ИТ).
Типовая информационно-вычислительная система (ИВС)
включает следующие компоненты:

- программы,
- данные,
- вычислительную технику.

ИВС составляют основу автоматизированных систем управления (**АСУ**).

АСУ

```
graph TD; A[АСУ] --> B[АСУТП – АСУ технологическим процессом]; A --> C[АСУП – АСУ предприятием];
```

АСУТП – АСУ

технологическим
процессом

*Управление
оборудованием.*

АСУП – АСУ

предприятием

***Управление
предприятием** в
широком смысле.*

Корпоративная информационная система (КИС)

- система, предназначенная для комплексной автоматизации всех видов финансово-хозяйственной деятельности предприятия.

ERP система (Enterprise Resource Planning) - система планирования ресурсов предприятия.

АСУП,

как правило, включают в себя различные подсистемы управления:

- ☐ продажами,
 - ☐ закупками,
 - ☐ финансами,
 - ☐ запасами,
 - ☐ складами,
 - ☐ персоналом,
 - ☐ отношениями с клиентами
- и т.п.

П-ОИС:
1. причем здесь управление,
2. и что есть «предмет»?

Любая АСУ в процессе своей работы
должна поддерживать выполнение
следующих функций:

- сбор, обработка и анализ информации о состоянии объекта управления (сигналов, сообщений, документов и т.п.);
- выработка управляющих воздействий (программ, планов и т.д.);
- передача управляющих воздействий на исполнение (сигналов, указаний, документов и т.п.);
- реализация и контроль выполнения управляющих воздействий.

Обзор системы «1С:Предприятие 8»

<http://v8.1c.ru/>

Система программ «1С:Предприятие 8» включает в себя **платформу** и **прикладные решения**, разработанные на ее основе (**конфигурации**), для автоматизации деятельности организаций и частных лиц.

Платформа представляет собой framework, в котором функционирует прикладное решение: является средой их исполнения и содержит инструментарий, необходимый для их **разработки** и **администрирования**.

Некоторые **типовые решения** на платформе "1С:Предприятие 8"

Комплексная автоматизация:

"1С:Управление производственным предприятием";

"1С:ERP Управление предприятием";

"1С:Управление торговлей";

....

Автоматизация отдельных видов деятельности:

"1С:Бухгалтерия";

"1С:Отчетность предпринимателя";

"1С:Зарплата и управление персоналом";

....

Отраслевые и специализированные решения на платформе "1С:Предприятие 8"

Примеры классификации по функциональным задачам.

- ❖ Управление взаимоотношениями с клиентами (CRM).
- ❖ Управленческий и финансовый учет (FRP).
- ❖ Управление продажами, логистикой и транспортом (SFM, WMS, TMS).
- ❖ Управление проектами и портфелями проектов (PMO, EPM).
- ❖

Примеры отраслевых решений:

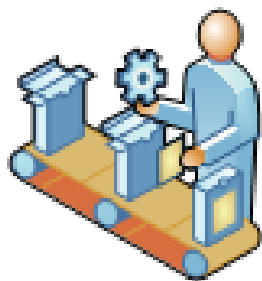
- ✓ 1С:Медицина. Стоматологическая клиника.
- ✓ 1С:Предприятие 8. Молокозавод.
- ✓ 1С:Предприятие 8. Рыбопереработка.
- ✓ 1С:Страховая компания 8 КОРП.



Разработчики
платформы



Разработчики типовых
прикладных решений



Разработчики прочих
прикладных решений



Конечные пользователи

Основная задача платформы заключается в повышении **уровня абстракции** при разработке и использовании (?) прикладных решений.

разработка «с нуля» **vs.** разработка «на платформе»

Это позволяет перейти от технических и низкоуровневых понятий к более содержательным и высокоуровневым.

Позволяет приблизить эти понятия к языку пользователей и специалистов в предметной области (?)

Предметно-ориентированное программирование

- парадигма программирования, в которой язык максимально приближен к терминологии предметной области.

Версия для обучения программированию

<http://v8.1c.ru/edu/>



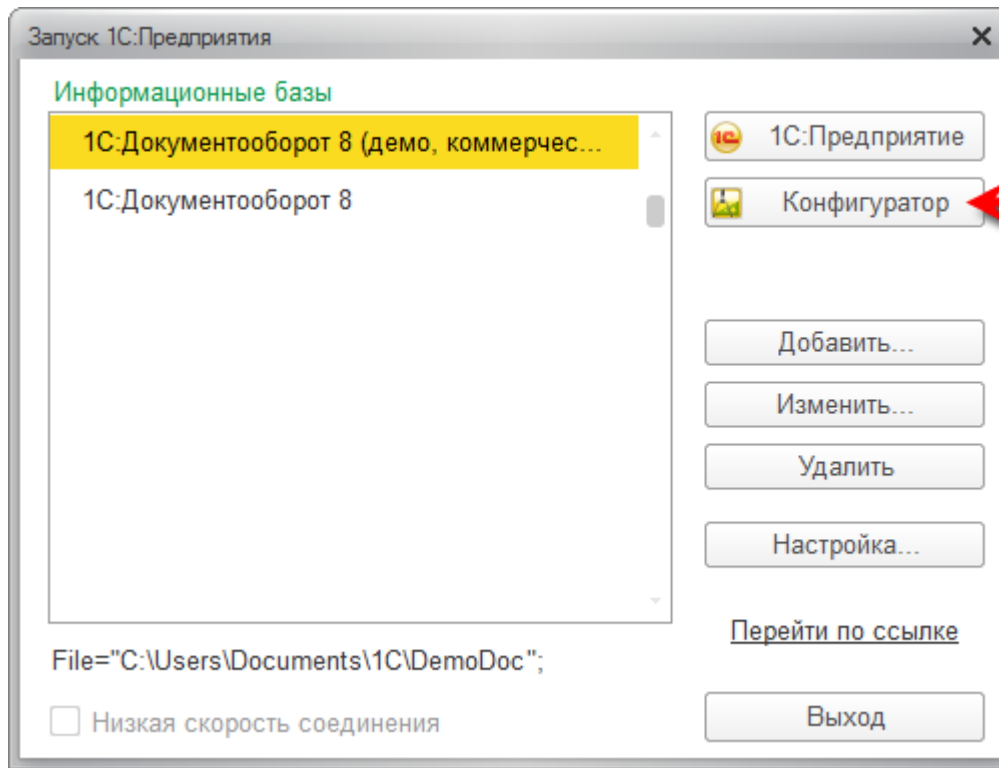
978 руб.

купить

скачать бесплатно

- ♦ **удобно** - не имеет аппаратной или программной защиты;
- ♦ **можно** - создать собственную конфигурацию;
- ♦ **можно** - изучить любую имеющуюся конфигурацию;
- ♦ **можно** - доработать любую имеющуюся конфигурацию, в том числе для реальных задач автоматизации предприятия;
- ♦ **нельзя** - использовать на предприятии для ведения реального учета пользователями.

Режимы работы системы



В режиме **Конфигуратора** разрабатываются прикладные решения и выполняется администрирование **информационных баз**.

Информационная база = конфигурация + данные.

В режиме **1С:Предприятие** осуществляется работа с прикладным решением.

Варианты работы системы

Платформа поддерживает два варианта работы.

1. файловый.

Рассчитан на работу небольшого количества пользователей в локальной сети. Все данные *информационной базы* располагаются в одном файле.

2. клиент-серверный.

Предназначен для использования в масштабе предприятия. Реализован на основе трехуровневой архитектуры. Данные *информационной базы* хранится в одной из поддерживаемых СУБД, а взаимодействие между клиентским приложением и СУБД осуществляет кластер серверов 1С:Предприятия 8.

И в том, и в другом варианте все прикладные решения работают полностью идентично.

Работа с базой данных:



Разработчик 1С:Предприятия не обращается к базе данных **напрямую**, он работает с платформой.

Подключение через Интернет

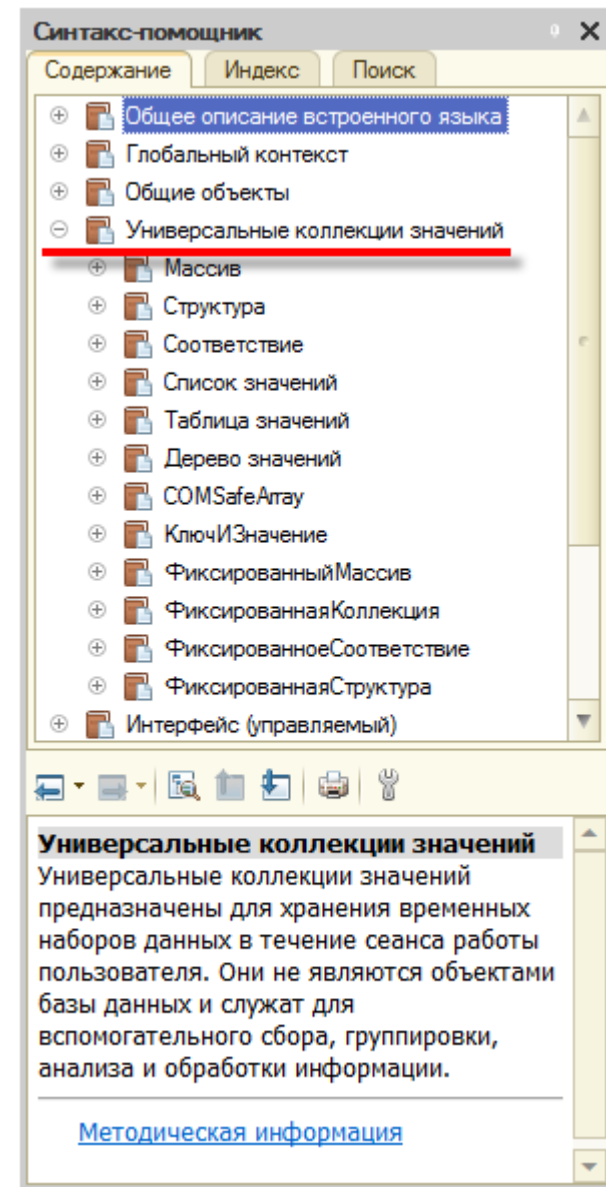
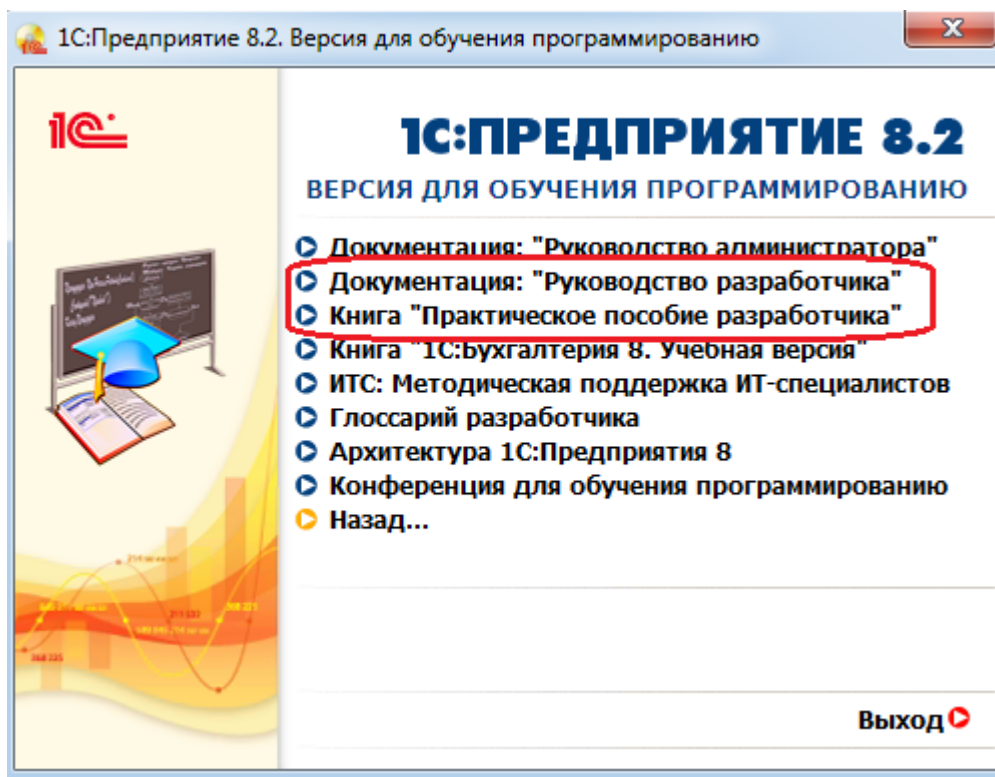


Тонкий клиент требует предварительной установки на компьютере пользователя.

Веб-клиент работает под управлением интернет-браузера и не требует установки, однако поддерживает не все возможности платформы

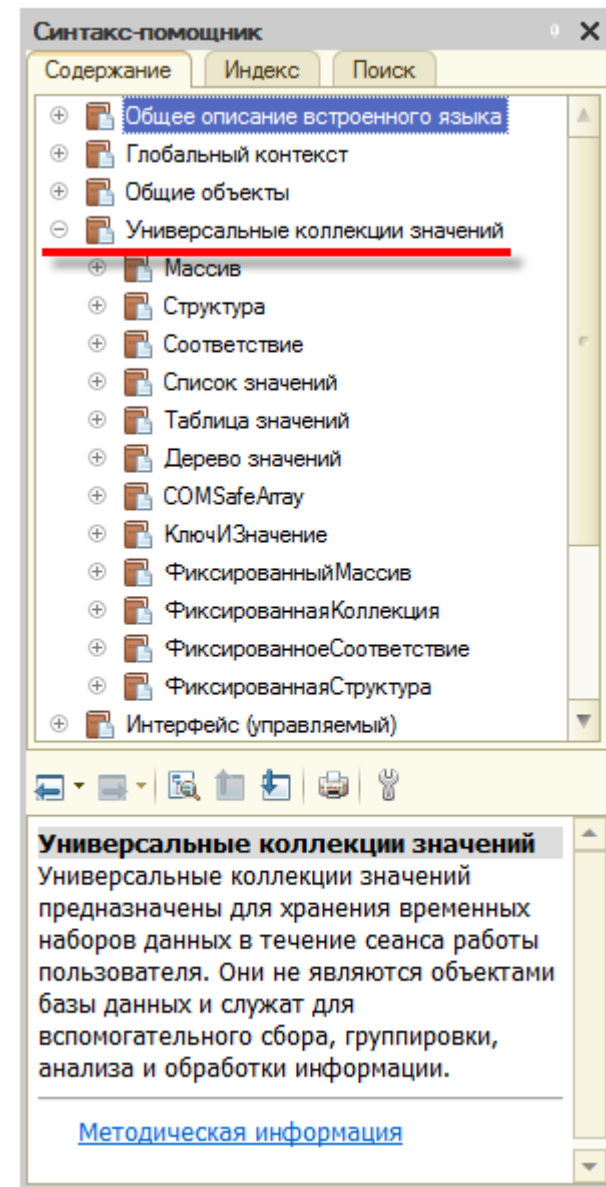
Учебно-методические материалы

1. «**Практическое пособие разработчика**» идет в комплекте с учебной версией.
2. «**Руководство разработчика**» идет в комплекте с учебной версией.
3. **Синтакс-помощник** – справочник доступный в режиме конфигуратора.



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.

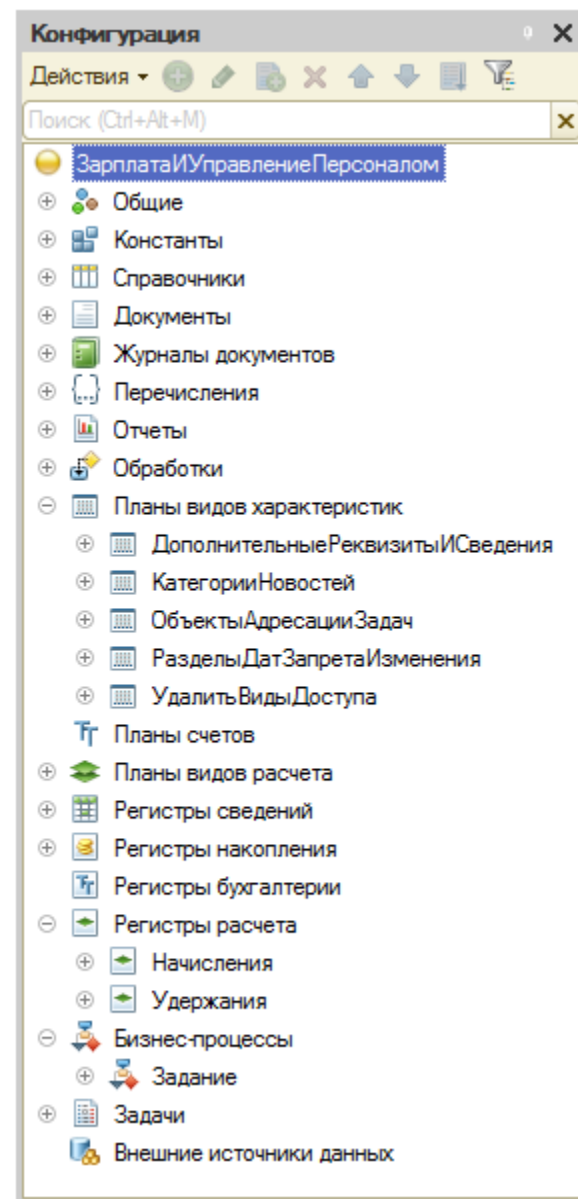


Типы данных, образуемые в прикладном решении

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

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

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



Для каждого класса прикладных объектов определена соответствующая ему базовая функциональность:

- ✓ типы таблиц базы данных; */повод поговорить о:/*
- ✓ типовые формы; */механизме запросов,/ /редакторе формы,/*
- ✓ типовые объекты языка; */встроенном языке,/*
- ✓ наборы прав и пр. */системе прав доступа./*

Конкретное имя типа, состав свойств и методов объекта конфигурации будут зависеть от разработчика.

Редактор формы

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;

Справочник Товары: ФормаЭлемента

+

✕

↕

↕

📄

Форма

📄 Командная панель

— Родитель

— Код

— Наименование

— Вид

— Артикул

— Поставщик

— ШтрихКод

📁 Страницы

📁 ГруппаКартинки

📁 ГруппаКартинкиСКодом

Элементы

Командный интерфейс

+

📄

✕

Реквизит

Используй...
всегда

Тип

— Объект

— Ссылка ☒ СправочникСсылка.Товары

— Код ☐ Строка

— Наименование ☐ Строка

— Родитель ☐ СправочникСсылка.Товары

— ЭтоГруппа ☒ Булево

— ПометкаУдаления ☒ Булево

— Предопределенный ☒ Булево

— ИмяПредопределе... ☒ Строка

— Артикул ☒ Строка

Реквизиты

Команды

Параметры

Записать и закрыть

Записать

Создать на основании ▾

Печать ▾

Группа:

Код:

Наименование:

Вид: ☐ Товар ☐ Услуга

Артикул:

Поставщик:

Штрихкод:

Изображение

Характеристики

Описание

Форма

Модуль

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

Встроенный язык

позволяет разработчику описывать собственные алгоритмы функционирования прикладного решения.

Большая часть прикладного решения создается разработчиком путем визуального конструирования - создания новых объектов конфигурации, задания их свойств, форм представления, взаимосвязей и пр.

Встроенный язык используется лишь для того, чтобы определить поведение объектов прикладного решения, отличное от типового, и создать собственные алгоритмы обработки данных.

Ключевые слова

Русское имя	Английское имя
Если	If
Тогда	Then
ИначеЕсли	ElsIf
Иначе	Else
КонецЕсли	EndIf
Для	For
Каждого	Each
Из	In
По	To
Пока	While
Цикл	Do
КонецЦикла	EndDo
Процедура	Procedure
Функция	Function
КонецПроцедуры	EndProcedure
КонецФункции	EndFunction
Перем	Var
Перейти	Goto
Возврат	Return
Продолжить	Continue
Прервать	Break

Русское имя	Английское имя
И	And
Или	Or
Не	Not
Попытка	Try
Исключение	Except
ВызватьИсключение	Raise
КонецПопытки	EndTry
Новый	New
Выполнить	Execute

Событийная ориентированность встроенного языка

Использование модулей на встроенном языке является событийно-зависимым, т.е. выполнение модулей происходит **при возникновении определенных событий** в процессе функционирования прикладного решения.

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	AtClient
НаСервере	AtServer
НаСервереБезКонтекста	AtServerNoContext
НаКлиентеНаСервереБезКонтекста	AtClientAtServerNoContext
НаКлиентеНаСервере	AtClientAtServer

Инструкции препроцессора

действуют на произвольный фрагмент программного кода.

Синтаксис:

#Если <Логическое выражение> Тогда

#ИначеЕсли <Логическое выражение> Тогда

...

#Иначе

#КонецЕсли

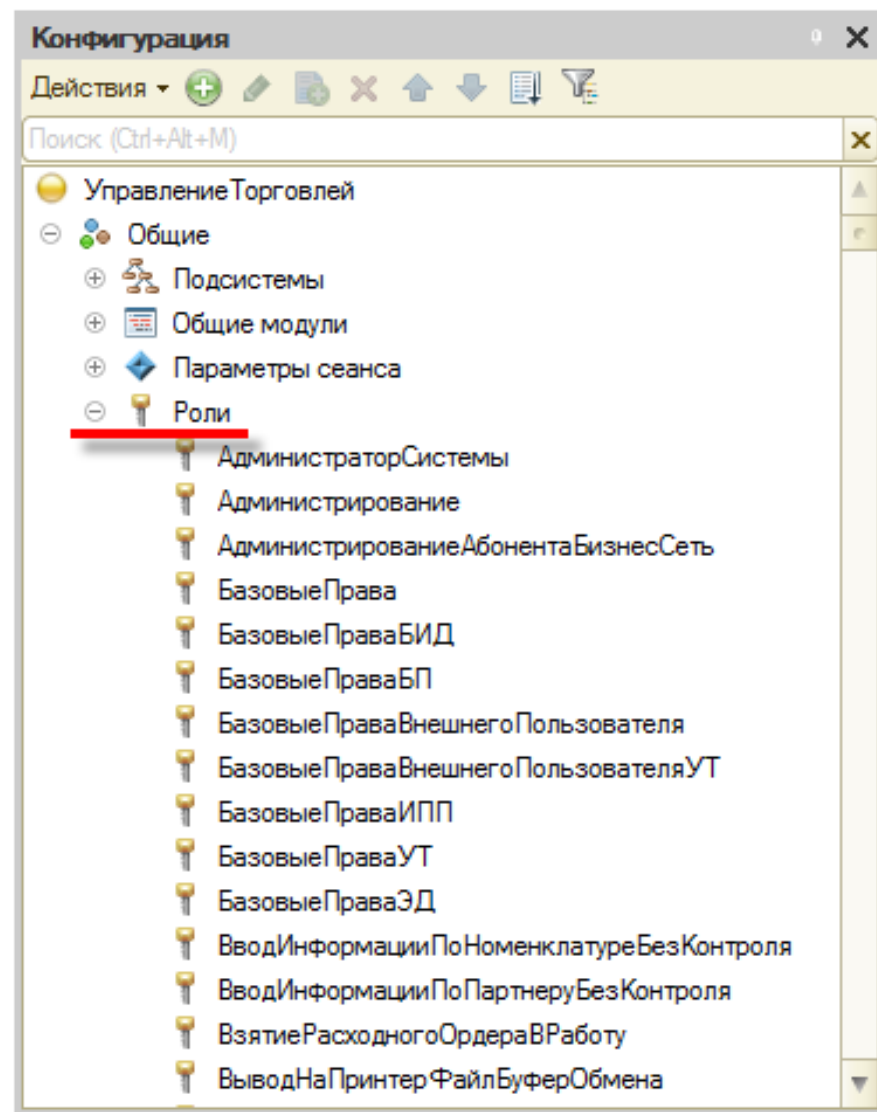
В логическом выражении используются:

Сервер
НаСервере
Клиент
НаКлиенте
ТонкийКлиент
ВебКлиент
ВнешнееСоединение
ТолстыйКлиентУправляемоеПриложение
ТолстыйКлиентОбычноеПриложение

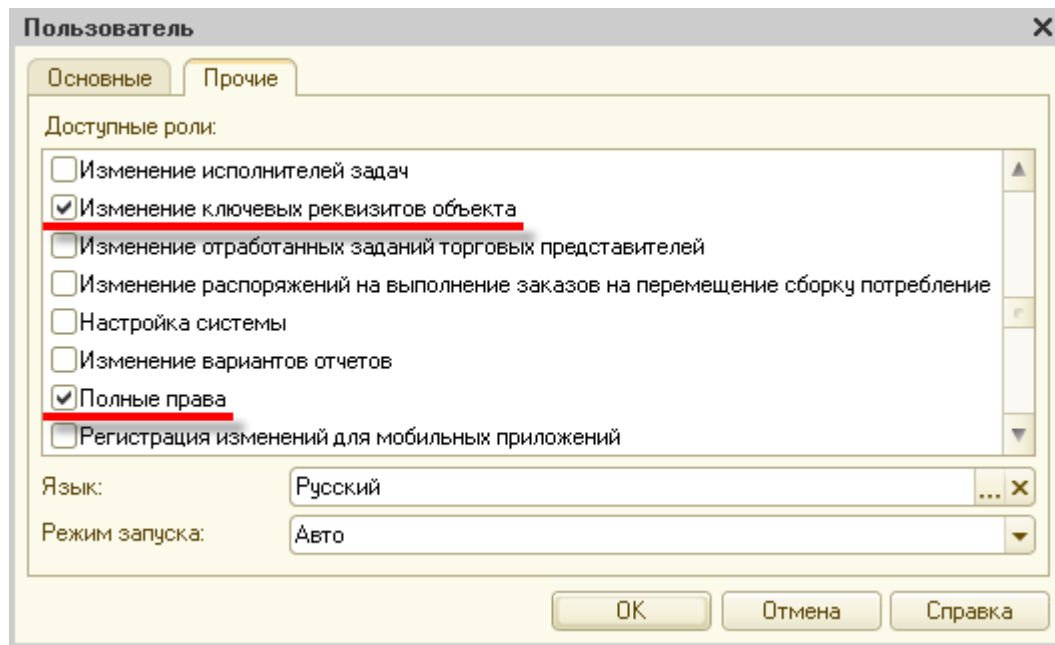
Система прав доступа

Роли - это общие объекты конфигурации.
Они предназначены для реализации ограничения прав доступа в прикладных решениях.

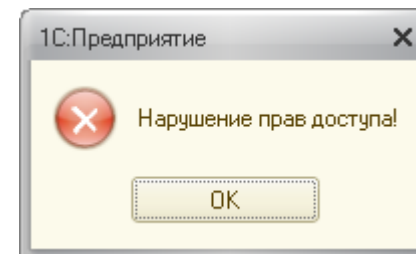
Роль в конфигурации может соответствовать **должностям** или **видам деятельности** различных групп пользователей.



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

Роль ДобавлениеИзменениеЗаказовКлиентов: Права

Действия ▾

Права | Шаблоны ограничений

Объекты:

- ВнутрифирменнаяПродажа
- ВозвратТоваровОтКлиента
- ВозвратТоваровПоставщику
- Встреча
- ВыпискаПоРасчетномуСчету
- ЗаданиеТорговомуПредставителю
- ЗаказКлиента**
- ЗаказНаВнутреннееПотребление
- ЗаказНаПеремещение
- ЗаказНаСборку
- ЗаказПоставщику
- ЗапланированноеВзаимодействие
- ЗаявкаНаРасходованиеДенежныхСредств
- ИнвентаризацияТоваровНаСкладе
- ИнвентаризацияТоваровПередаваемых

Права:

- ☒ Чтение
- ☒ Добавление
- ☒ Изменение
- ☐ Удаление
- ☒ Проведение
- ☒ Отмена проведения
- ☒ Просмотр

Ограничения доступа к данным:

Поля	Ограничение доступа
<Прочие поля>	#ПоЗначениям{"Документ.ЗаказКлиента","Чтение","","Организации","Организация","Склады","Склад","ГруппыПартнеров","Партнер",,,,,,,}

Описание: Право на чтение (Чтение)

☐ Устанавливать права для новых объектов

☒ Устанавливать права для реквизитов и табличных частей по умолчанию

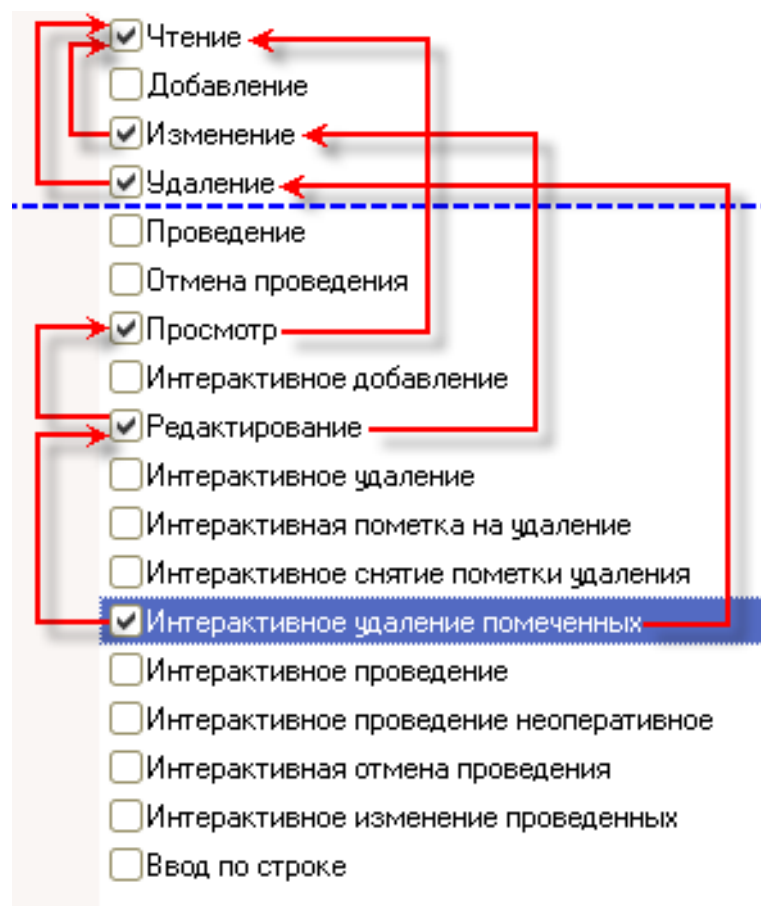
Метаданные - сведения о признаках и свойствах, характеризующих какие-либо сущности.

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.



Запросы

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

Язык запросов схож с SQL.

