

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КубГУ»)

Кафедра прикладной математики

КУРСОВАЯ РАБОТА

**АВТОМАТИЗАЦИЯ ДОКУМЕНТООБОРОТА
ПРОИЗВОДСТВЕННОГО ПРЕДПРИЯТИЯ**

Работу выполнил _____ Р.В. Задорожний _____ Р.В. Задорожний
(подпись, дата) (инициалы, фамилия)

Факультет компьютерных технологий и прикладной математики курс 3
Направление 09.03.03 Прикладная информатика

Научный руководитель,
Профессор,
Д-р. ф. - м.н. _____ М.Х. Уртен _____ М.Х. Уртен
(подпись, дата) (инициалы, фамилия)

Нормоконтролер,
К.ф. - м.н. _____ Г.В. Калаидина _____ Г.В. Калаидина
(подпись, дата) (инициалы, фамилия)

Краснодар 2018

СОДЕРЖАНИЕ

| | |
|---|----|
| ВВЕДЕНИЕ..... | 4 |
| 1 1С на различных платформах | 5 |
| 1.1 1С:Предприятие на Linux | 5 |
| 1.2 Особенности разработки приложений 1с для Linux..... | 7 |
| 1.3 1С:Предприятие на Mac OS | 8 |
| 1.4 1С:Предприятие на Windows | 11 |
| 1.5 Мобильные решения на платформе "1С:Предприятия 8" | 11 |
| 1.6 1С-Битрикс..... | 13 |
| 1.6.1 Программные продукты Битрикс | 15 |
| 2 Создание информационной базы. Конфигуратор..... | 20 |
| 2.1 Подсистемы..... | 21 |
| 2.2 Справочники | 23 |
| 2.3 Документы | 29 |
| 2.4 Регистры накопления..... | 32 |
| 2.5 Простой отчет | 34 |
| 2.6 Макеты. Редактирование макетов и форм..... | 36 |
| 2.7 Периодические регистры сведений..... | 39 |
| 2.8 Перечисления..... | 40 |
| 2.9 Проведение документа по нескольким регистрам..... | 42 |
| 2.10 Обратные регистры накопления..... | 43 |
| 2.11 Оптимизация проведения документа «Оказание услуги» | 51 |

| | |
|--|----|
| 2.12 План видов характеристик | 57 |
| 2.13 План видов расчета, регистр расчета | 65 |
| 2.14 Иллюстрация механизмов вытеснения и зависимости от базы | 68 |
| ЗАКЛЮЧЕНИЕ | 72 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ | 74 |

ВВЕДЕНИЕ

На данный момент 1С является одной из передовых российских компаний, специализирующихся на дистрибуции, поддержке и разработке компьютерных программ и баз данных делового и домашнего назначения. В связи с развитием информационных технологий и продвижении многоплатформенности приложений 1С создает программные продукты как для различных ОС персональных компьютеров, так и для ОС мобильных устройств.

Целью курсовой работы является автоматизация документооборота ООО «На все руки мастер», а именно создание новой конфигурации на платформе 1С:Предприятие.

Основные задачи курсовой работы – изучить на каких операционных системах работает 1С, что такое «1С-Битрикс» и создать конфигурацию «Мастерская» для ООО «На все руки мастер», которая содержит справочники, документы, отчеты, планы видов характеристик, план счетов, планы видов расчета, регистры сведений, регистры накопления, регистры бухгалтерии, регистры расчета.

Курсовая работа состоит из двух глав, введения, заключения и списка использованной информации.

В первой главе подробно описываются решения 1С для различных операционных систем и информация об «1С-Битрикс».

Во второй главе описывается поэтапное создание конфигурации для ООО «На все руки мастер» (справочники, документы, номенклатуры, склады и хранящихся на них ресурсы, отчеты, информация о производимой продукции).

В заключении приведены основные источники курсовой работы.

1 1С на различных платформах

Многоплатформенность – это способность системы работать под управлением различных операционных систем. Основные компоненты системы могут работать как под управлением операционной системы Windows, так и под управлением операционной системы Linux. Кроме этого клиентская часть 1С:Предприятия может быть запущена и на компьютерах с операционными системами OS X и Apple iOS.

1.1 1С:Предприятие на Linux

Компания 1С выпустила финальную версию платформы для автоматизации бухгалтерского учёта и автоматизации деятельности предприятия "1С:Предприятие" 8.3, в состав которой впервые включены клиентские приложения и инструменты администрирования для Linux.

Начиная с версии 8.3 "1С:Предприятие" содержит весь спектр необходимых компонентов, позволяющих организовать работу как серверных, так и клиентских систем только с использованием Linux, без необходимости задействования Windows.

Также допускается создание комбинированных систем, подразумевающих включение в состав кластера "1С:Предприятие" рабочих станций и серверов, использующих различные операционные системы.

Схема работы 1С на Linux представлена на рисунке 1.1.

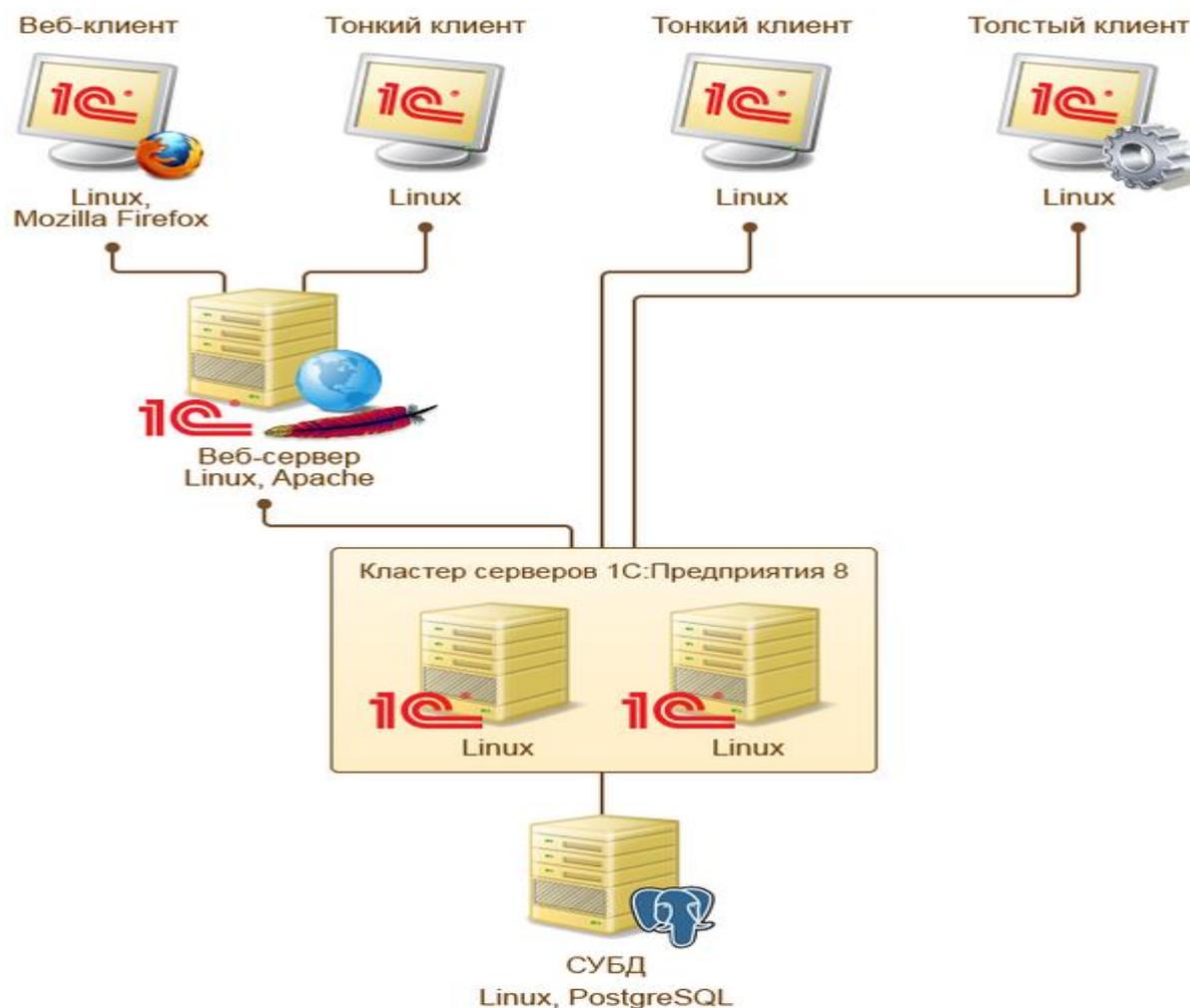


Рисунок 1.1 – Схема работы 1С на Linux

Обеспечение в ПО "1С:Предприятие" официальной полноценной поддержки Linux, не уступающей по своему уровню версии для Windows, является важной вехой в продвижении свободного ПО в отечественном корпоративном секторе, так как указанный продукт используется на большинстве российских предприятий.

Ранее компании имели возможность использовать клиентскую часть "1С:Предприятие" в Linux только путем запуска Windows-версии с привлечением дополнительных надстроек, основанных на Wine, таких как WINE@Etersoft. Тем не менее, следует отметить, что первый шаг в сторону многоплатформенности был сделан компанией 1С ещё в 2009 году, когда был выпущен поддерживающий Firefox многоплатформенный web-клиент и обеспечена поддержка СУБД PostgreSQL в серверной части.

В состав 1С:Предприятие" включены:

1) клиентские приложения для платформы Linux, полностью аналогичные тем, которые существуют для операционной системы Windows. В том числе, "тонкий клиент", обеспечивающий работу пользователей в режиме управляемого приложения; "толстый клиент", обеспечивающий работу в режиме обычного или управляемого приложения или в режиме конфигуратора. Оба клиентских приложения доступны для архитектур x86 и x86-64, и поддерживают файловый и клиент-серверный варианты работы.

2) инструменты для администрирования кластера с систем под управлением Linux. В состав включены Linux-версии сервера администрирования (для передачи управляющих команд одному кластеру серверов 1С:Предприятия) и управляющей утилиты, работающей в режиме командной строки и поддерживающей весь спектр команд администрирования.[1]

1.2 Особенности разработки приложений 1с для Linux

Для начала стоит определить на какой ОС запущена конфигурация, это делается примерно так, как представлено на рисунке 1.2.

```
Процедура ПередНачаломРаботыСистемы(Отказ)
Информация = Новый СистемнаяИнформация;
Если Информация.ТипПлатформы = ТипПлатформы.Linux_x86 Или Информация.ТипПлатформы = ТипПлатформы.Linux_x86_64 Тогда
//
Иначе
//
КонецЕсли;
КонецПроцедуры
```

Рисунок 1.2 – Определение ОС

Далее – отказаться (найти замену) от всех не кроссплатформенных сторонних программ и библиотек (Office, WinHttp и так далее).

Вместо объекта «Почта» следует использовать объект «ИнтернетПочта». А внешний вид элемента управления «ПолеHTMLДокумента» может отличаться от привычного.

Если где-либо в конфигурации используются WMF или EMF файлы, то их следует заменить на общепринятые растровые форматы (PNG или JPEG).

Особое внимание нужно обратить на работу с файловой системой. Нужно помнить, что в Linux имена файлов являются регистро-зависимыми. Кроме этого не следует вручную прописывать разделители имени файла и маску всех файлов («/» и «*.»»), вместо этого рекомендуется использовать функции «ПолучитьРазделительПути» и «ПолучитьМаскуВсехФайлов».

И, наконец, если для работы с именами файлов используете «Библиотеку стандартных подсистем», то рекомендуется использовать функции из общего модуля «ОбщегоНазначенияКлиентСервер».[2]

1.3 1С:Предприятие на Mac OS

В преддверии 2016 года 1С объявила об официальной поддержке продукции компании Apple. Теперь все версии платформы начиная с 8.3.7.1845 поддерживают работу с OS X. Однако, на момент выхода Mac OS версия платформы имела серьёзные недостатки.

1С:Предприятие. В статусе бета версии реализованы родные (native) клиентские приложения и конфигуратор, работающие под управлением операционной системы OS X версии 10.8 и выше.

Теперь пользователи персональных компьютеров, выпускаемых компанией Apple, могут не только работать с прикладными решениями 1С:Предприятия, но и вести полноценную разработку конфигураций.

Клиентские приложения и конфигуратор реализованы для 64-битной архитектуры. Благодаря тому, что это родные для OS X приложения, они имеют возможность учитывать специфические особенности устройств Apple. Например, уже в бета-версии оба приложения поддерживают работу с дисплеями высокого разрешения Retina.

Особенности бета версии

При запуске прикладных решений не выполняется автоматический подбор нужной версии платформы для работы с информационной базой.

Поэтому пока необходимо сразу же запускать клиентское приложение нужной версии.

1) Невозможно обновление клиентского приложения по протоколу HTTP.

2) Не поддерживается работа с внешними компонентами, созданными по технологии Native API.

3) Отсутствует возможность установки шаблонов конфигураций из комплектов поставки. Для создания информационных баз из шаблона необходимо предварительно вручную скопировать все файлы шаблона на устройство Apple.

4) Функциональность, реализация которой рассматривается после выпуска финальной версии.

5) Работа в файловом варианте по локальной сети. Пока в файловом варианте можно работать только с локальными информационными базами. То есть с базами, расположенными на том же компьютере, на котором запускается клиентское приложение.

6) Поддержка панели меню OS X.

7) Работа с внешними источниками данных на стороне клиента.

8) Возможность аутентификации пользователя средствами операционной системы.

Функциональные ограничения, накладываемые операционной системой OS X

OS X не поддерживает технологию COM, поэтому:

1) Невозможна работа с объектом COMОбъект.

2) Невозможен запуск 1С:Предприятия в режиме Automation-сервера.

3) Невозможна работа с внешними компонентами, созданными с использованием COM-технологии.

4) Отсутствует консоль администрирования кластера серверов 1С:Предприятия. Администрирование можно выполнять с помощью кроссплатформенных инструментов (сервер администрирования и утилита командной строки).

5) Невозможна работа с объектом «Почта». При этом функциональность объекта «ИнтернетПочта» поддерживается в полном объеме.

6) Не используются метафайлы Windows (WMF, EMF).

Список недостатков:

1) Отсутствие поддержки технологии COM
2) Отсутствие поддержки работы клиентских приложений с панелью меню OS X

3) Отсутствие поддержки работы с файловой информационной базой, расположенной на сетевых ресурсах

4) Отсутствие поддержки работы с внешними источниками данных на стороне клиента и конфигуратора

5) Отсутствие поддержки аутентификации пользователя средствами операционной системы

6) Наличие только тонкого и толстого клиента, конфигуратора и отсутствие сервера.

Недостатки 1С на MAC OS X и на Linux:

1) Возможные проблемы с внешними обработками, использующими приложения Microsoft Office (Excel и т.д.).

2) Отсутствие поддержки компонент защиты (актуально для некоторых отраслевых продуктов).

3) Поддержка только конфигураций с управляемым интерфейсом.[1]

1.4 1С:Предприятие на Windows

Схема работы 1С:Предприятие на базе операционной системы Windows представлена на рисунке 1.3.

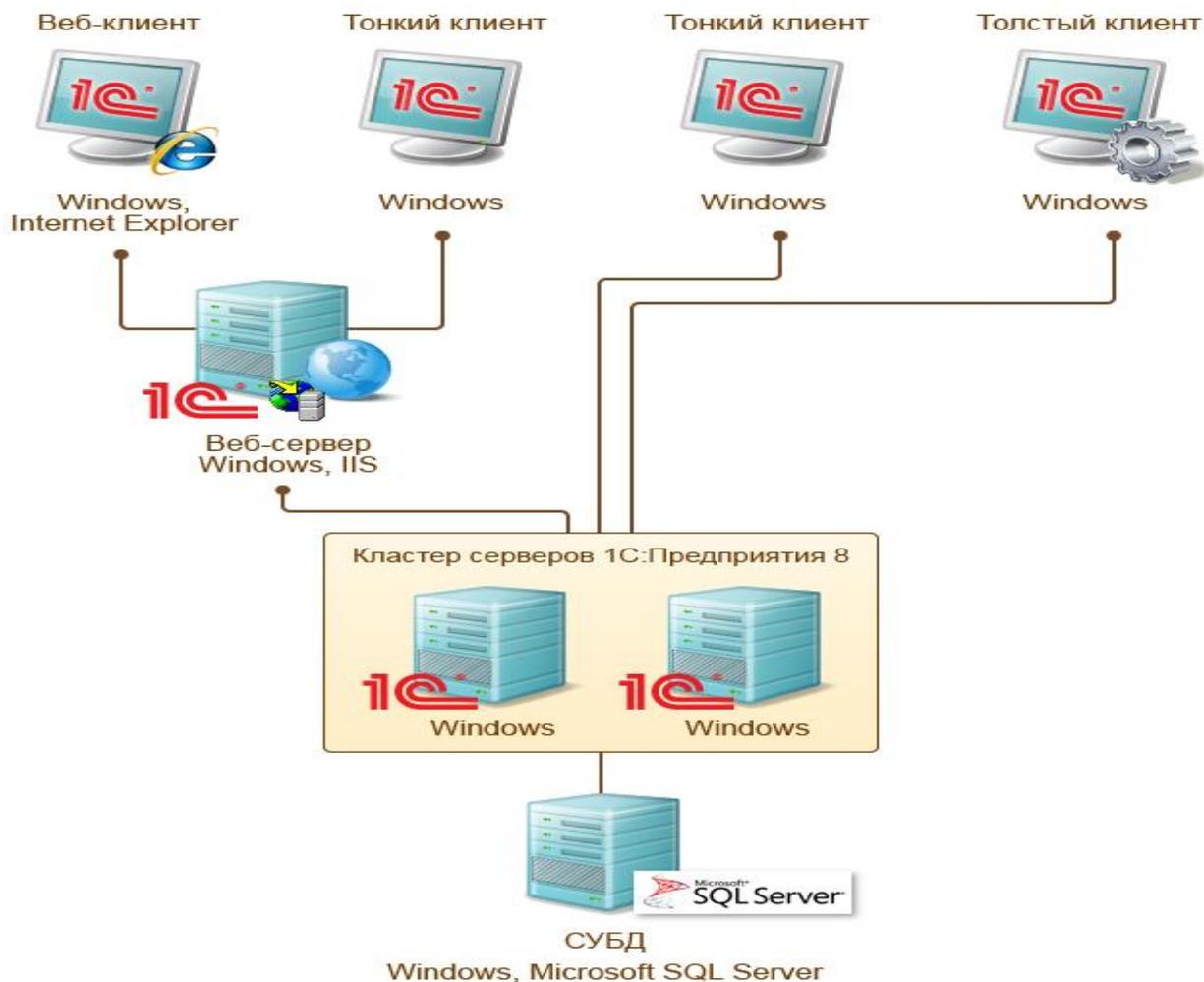


Рисунок 1.3 – Схема работы 1С:Предприятие на ОС Windows

1.5 Мобильные решения на платформе "1С:Предприятия 8"

Фирма "1С" выпустила ряд приложений, разработанных на мобильной платформе "1С:Предприятия", предназначенных для использования на смартфонах и планшетах в операционных системах iOS, Android и Windows:

- 1) "1С:Документооборот"
- 2) "1С:Управление нашей фирмой" (1С:УНФ)
- 3) "1С:Заказы"

- 4) "1С:Монитор ERP"
- 5) "1С:Мобильная касса"
- 6) Миниденьги
- 7) Несколько решений для пользователей сервисов 1С:
 - а) Клиент бухгалтерии 1cfresh
 - б) Клиент 1С:БухОбслуживания
- 8) "Конферометр" и несколько приложений на его основе для конкретных мероприятий,
- 9) Мобильный тренажер "1С:Профессионал".

Мобильная платформа «1С:Предприятия 8» — это название технологии, которая позволяет разрабатывать решения, работающие на смартфонах и планшетах Google Android или Apple iOS. Важным преимуществом платформы «1С» является то, что решения разрабатываются в одном варианте для обеих мобильных операционных систем и затем просто экспортируются для Android и для iOS без какой-либо дополнительной модификации и написания платформенно-зависимого программного кода.

На мобильное устройство устанавливается сразу и платформа «1С:Предприятие» и сама информационная база (конфигурация, в терминах «1С»). Информационная база на устройстве содержит вариант файловой базы данных (для хранения различных данных, с которыми работает или будет работать пользователь) и само приложение (программный байт-код, который исполняющийся на планшете или смартфоне).

На базе мобильной платформы «1С» могут быть созданы различные решения, но основным вариантом решений являются мобильные off-line клиенты для учетных систем. Такие решения предназначены для поддержки мобильных пользователей в удаленных рабочих местах. Обмен данными при этом производится посредством регулярной синхронизации с центральной базой данных. Обмен данными возможен посредством следующих механизмов:

- 1) Web-сервисы

- 2) Файловый обмен
- 3) Электронная почта
- 4) и другие механизмы

Альтернативным вариантом может являться работа в онлайн в режиме веб-клиента.

Разработанное мобильное приложение является отдельным, законченным программным продуктом и может обмениваться данными сразу с несколькими центральными базами данных. Например, в одном мобильном приложении можно реализовать доступ и работу с документами из бухгалтерской базы данных и задачами из системы документооборота предприятия.

Мобильные приложения, предназначенные для бизнеса, могут обмениваться данными с информационными базами типовых решений "1С:Предприятия", установленными локально или в облачном сервисе.

Мобильные решения доступны на сайтах Google play, App Store и Windows Store.

Наряду с готовым к работе мобильным приложением фирма "1С", как правило, выпускает также версию для разработчиков – типовое решение, на основе которого средствами мобильной платформы "1С:Предприятия" может быть создано новое прикладное решение.[3]

1.6 1С-Битрикс

Битрикс как программный продукт

Все программные продукты Битрикс созданы на базе Bitrix Framework.

С технической точки зрения программные продукты Битрикс (сайты и мобильные приложения) создаются на основе собственной платформы Bitrix Framework.

1) Сегодня эти программные продукты представлены следующими решениями:

2) 1С-Битрикс: Управление сайтом. Система для создания сайтов и интернет-магазинов.

3) 1С-Битрикс: Корпоративный портал. Платформа для создания мощных корпоративных порталов.

4) Мобильные приложения на основе Bitrix Framework (мобильные версии сайтов и т.д.)

5) Различные готовые отраслевые решения.

Кроме того, компания Битрикс поддерживает собственный магазин “App Store” под названием MarketPlace, где также можно купить или скачать бесплатно различные решения, например, шаблоны, дополнительные модули, надстройки для программных продуктов, готовые решения для подключения телефонии, интеграции с другими сервисами и т.д.

Разработкой программных продуктов на платформе Bitrix Framework занимается узкий круг программистов, состоящий из специалистов компании. Это коммерческий продукт, а потому такое ограничение можно считать преимуществом. Обновления программных продуктов предсказуемы и понятны, подход к работе практикуется комплексный, а новые функции обычно работают именно так, как заявлено.

Также компания очень ответственно подходит к работе своего магазина надстроек и приложений MarketPlace. Предложить разработку для этого магазина могут не только сотрудники компании, но и сторонние разработчики. Но любой предложенный продукт проходит строгую премодерацию, решение тестируется специалистами компании “Битрикс”, и только после их одобрения появляется в публичном доступе.

Из минусов подобного подхода необходимо отметить то, что количество расширений (модулей) намного меньше чем у продуктов с свободной лицензией, и они намного менее разнообразны. Оно и понятно — количество разработчиков не ограничено размерами компании.

1.6.1 Программные продукты Битрикс

Программные продукты Битрикс отличаются друг от друга преимущественно набором модулей, которые вошли в готовое решение, и делятся на категории по типу сайта, который может потребоваться в том или ином случае:

1) Решение «1С-Битрикс: Управление сайтом» - это программный продукт, который используют для разработки интернет-магазинов, сайтов-визиток и тому подобных проектов.

2) Решение «1С-Битрикс: Корпоративный портал» наилучшим образом подходит для крупных компаний и для любого бизнеса, где для работы нужен корпоративный портал.

3) Отраслевые решения – это специализированные решения на основе Bitrix Framework, в которых предусмотрены модули для организации работы в той или иной отрасли.

1С-Битрикс. Управление сайтом

«Управление сайтом» — это программный продукт, предназначенный для создания и управления интернет-магазинами, сайтами-визитками и т.д. Существует огромное количество редакций этого программного продукта, но все они предназначены для одной цели – создания сайта и дальнейшего его обслуживания (наполнения, редактирования и пр.).

Изначально «Битрикс» создавался как движок (CMS) для создания интернет-магазинов, а потому «Управление сайтом» имеет очень широкий перечень возможностей и богатый функционал. С другой стороны, всем богатством возможностей этого движка пользуются нечасто. Редко можно увидеть на битрикс-сайте несколько каталогов, практически не используются возможности маркетинга, так как для большинства интернет-магазинов не нужны все эти функции, но все же они имеются в числе возможностей программного продукта и при возникновении потребности их можно подключить в любой момент.

При создании интернет-магазина с использованием продукта «Управление сайтом» необходимо:

- 1) Установить на хостинг сам «движок»;
- 2) Установить дизайн, настроить информационные страницы;
- 3) Настроить каталог товаров и управление корзиной (оформление покупки);
- 4) При необходимости подключить платежные системы, обмен данными с учетной программой и т.п.

Большая часть функционала, который может потребоваться для реализации любого вида Интернет-торговли уже имеется в составе продукта «Управление сайтом», но также можно использовать собственные разработки или покупать решения в MarketPlace.

1С-Битрикс: Корпоративный портал

1С-Битрикс: Корпоративный портал – это некая площадка, где собирается информация, информационный центр компании. Это многофункциональный продукт, где можно объединить информацию по самым разным направлениям. Например, бухгалтерские данные компания ведет в 1С. Бухгалтерии, движение товаров по складам и продажи – в 1С. Торговля и склад, работу с потенциальными и реальными клиентами – в CRM-системе, техническую поддержку осуществляют в специализированном сервисе поддержки клиентов.

А корпоративный портал позволяет вести все эти направления в одном месте, собирать данные для управленческой отчетности, получать общую развернутую картину работы компании с разных точек зрения. В чем-то корпоративный портал Битрикс напоминает швейцарский нож: никакой специализации у этого инструмента нет, но с его помощью можно выполнять самые разные действия, так как в составе продукта есть инструменты для реализации работы с клиентами, с пользователями, и многое другое, пусть и на минимальном уровне.

1С-Битрикс: Корпоративный портал позиционируется также как продукт, который позволяет руководителю контролировать весь рабочий процесс компании, для чего используется разнообразная отчетность, также руководитель может выставлять функции, например, начала и завершения рабочего дня, ставить перед подразделениями задачи (проекты), устанавливать для них сроки и основные этапы реализации. Также с порталом может быть интегрирована CRM-система, телефония, бухгалтерские, учетные программы и т.д.

Существует два продукта от компании Битрикс, предназначенных для корпоративной работы:

1) Битрикс24 – SAAS-решение, где вы оплачиваете доступ к «облачному» сервису. Здесь вам не потребуется оплачивать собственный хостинг и услуги специалистов, все обновления сервиса будут вам доступны без каких-либо действий с вашей стороны. Но при этом вы сможете пользоваться только тем функционалом, который предлагает разработчик, а также решениями из MarketPlace. Собственные доработки в Битрикс24 невозможны.

2) Корпоративный портал – это программный продукт с открытым для покупателей кодом, который необходимо после покупки установить на собственный хостинг. Все настройки этого программного продукта вам нужно будет производить самостоятельно. Для получения обновлений вам понадобится их скачать и также своими силами установить. Но при этом у ваших специалистов есть доступ ко всем функциям программного продукта и любые доработки вам будут доступны.

Отраслевые решения

Специализированные отраслевые решения – это система «Управление сайтом» со всеми имеющимися в базовой версии функциями, дополненная предустановленными надстройками для организации работы в той или иной отрасли. Например, такие решения очень популярны для организации работы

медицинских учреждений, бюджетных организаций, образовательных учреждений и т.д.

Здесь уже имеются специальные анкеты для создания медицинской карты пациента или регистрации предпринимателя, плательщика налогов, будущего владельца недвижимости и т.д. и т.п. Реализованы варианты последовательности действий при записи на прием к специалисту, оплате государственной пошлины, счетов за услуги, а также другие модули, необходимые для создания удобного сайта для той или иной отрасли.

Композитный сайт

Композитный сайт – это еще одна технология, которую активно рекламируют разработчики 1С-Битрикс. Они позиционируют композитные сайты как возможность объединить высокую скорость загрузки со всеми функциями современного динамического сайта.

Как это работает:

- 1) Страницы сайта делятся на статическую и динамическую части.
- 2) Для быстрого отображения статической части активно применяется кэширование.
- 3) Динамическую часть система подгружает в фоновом режиме и также кэширует в браузере.

Активное использование кэширования позволяет снизить объем информации, которую сайт отправляет пользователю при каждом повторном обращении к сайту.

Т.е. при первом обращении к композитному сайту проходит обычный вариант загрузки страницы.

Из браузера пользователя к сайту на хостинге проходит обращение. Сайт формирует ответ, т.е. полноценную страницу вместе со всеми изображениями и другими видами контента. Пользователь дожидается загрузки всей информации на компьютер и видит полноценную страницу.

Композитные сайты после этого большую часть страницы (изображения, видео, другую статичную информацию) сохраняют в кэше

системы. И при повторном обращении учитывается наличие в кэше копии статичной части страницы, а потому формируется и передается только динамичная часть, остальное загружается из кэша. За счет этого скорость загрузки значительно возрастает.[4]

2 Создание информационной базы. Конфигуратор

Создаем новую пустую конфигурацию и открываем ее в режиме «Конфигуратор». Открытая конфигурация в «Конфигураторе» представлена на рисунке 2.1.

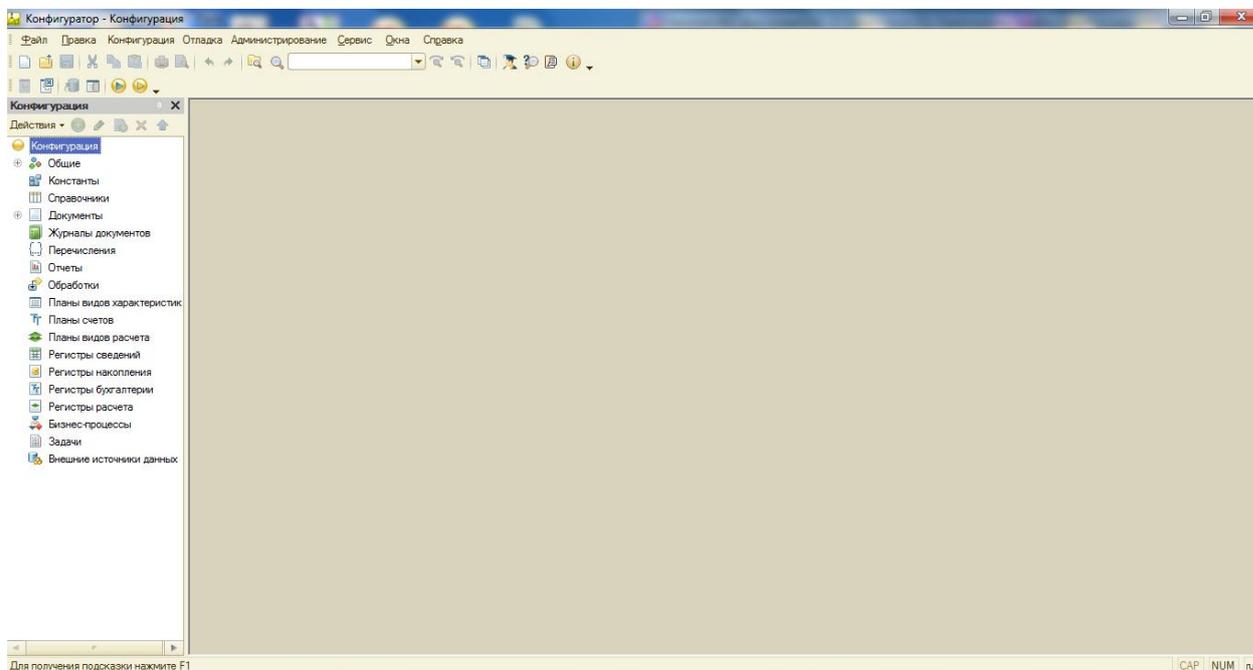


Рисунок 2.1 – Конфигурация в «Конфигураторе»

Откроем палитру свойств конфигурации, зададим имя конфигурации «ПособиеДляНачинающих» и «открепим» палитру свойств. Настройки палитры свойств представлены на рисунке 2.2.

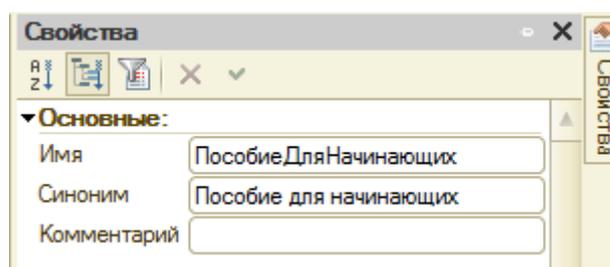


Рисунок 2.2 – Палитра свойств

Запустим отладку в режиме 1С:Предприятие. Запуск отладки представлен на рисунке 2.3.

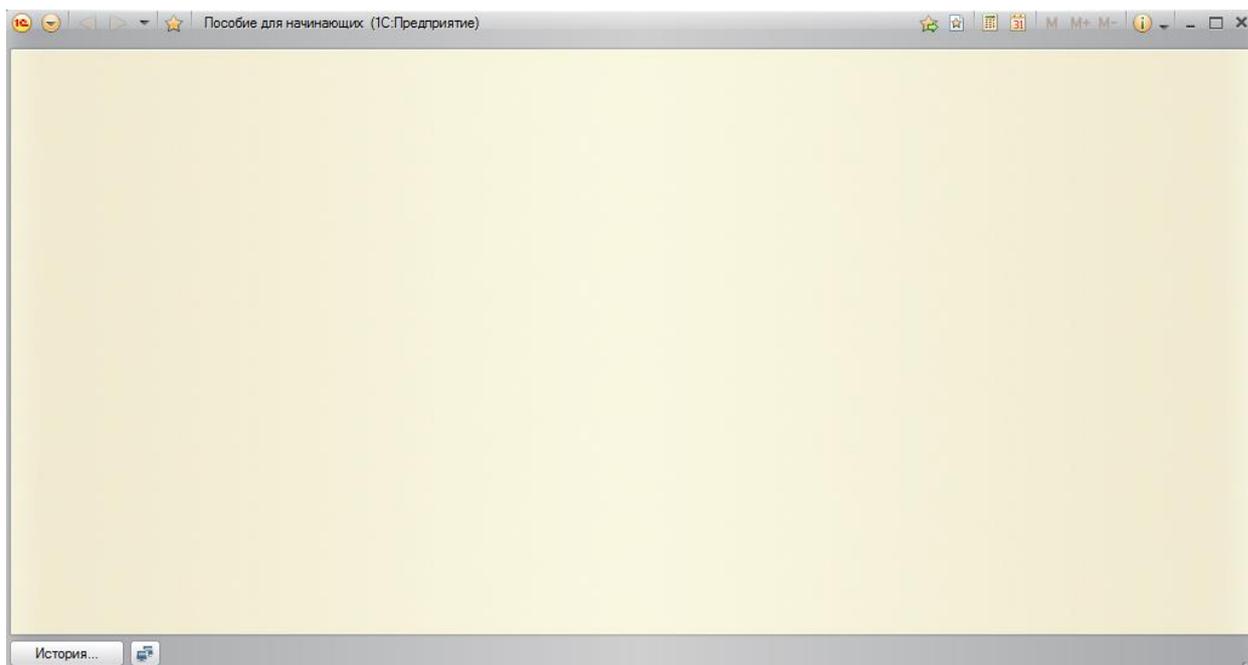


Рисунок 2.3 – Отладка в режиме 1С:Предприятие

2.1 Подсистемы

Добавим в режиме Конфигуратор подсистему Бухгалтерия. Добавление подсистемы представлено на рисунке 2.4.

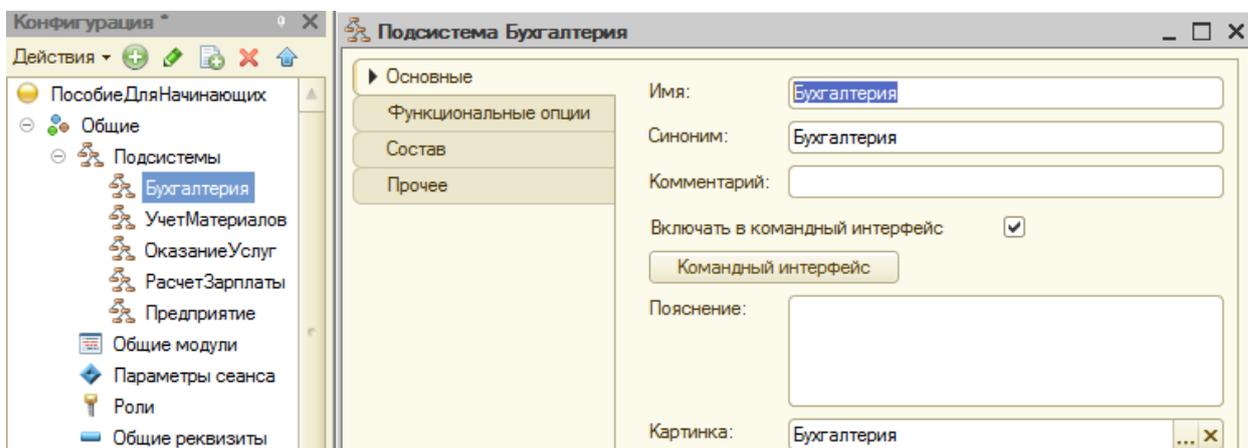


Рисунок 2.4 – Добавление подсистемы Бухгалтерия

Таким же образом добавим подсистемы УчетМатериалов, ОказаниеУслуг, РасчетЗарплаты и Предприятие. Добавление подсистем представлено на рисунке 2.5.

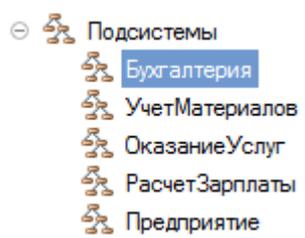


Рисунок 2.5 – Добавленные подсистемы

Панель разделов прикладного решения

Откроем конфигурацию в режиме отладки. Конфигурация представлена на рисунке 2.6.

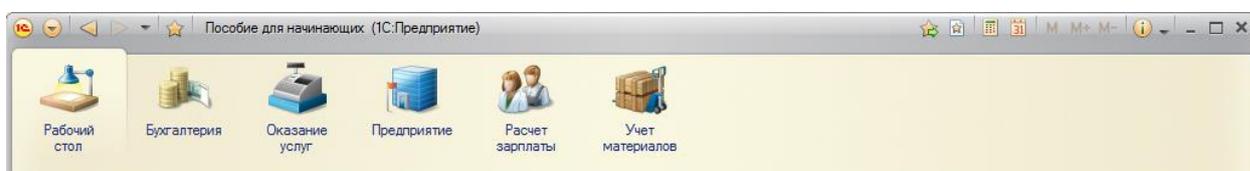


Рисунок 2.6 – Конфигурация с новыми подсистемами

Изменим порядок расположения разделов. Изменение порядка представлено на рисунке 2.7.



Рисунок 2.7 – Измененный порядок

Снова запустим конфигурацию в режиме отладки. Измененная конфигурация представлена на рисунке 2.8.

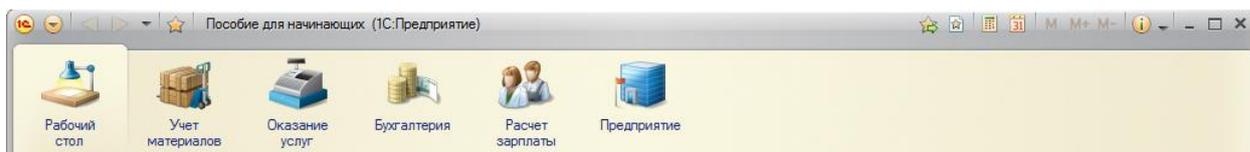


Рисунок 2.8 –Измененная конфигурация

2.2 Справочники

«Простой» справочник

Создадим новый объект конфигурации – справочник «Клиенты».

Укажем основные свойства справочника для правильного представления его в конфигурации. Заданные свойства представлены на рисунке 2.9.

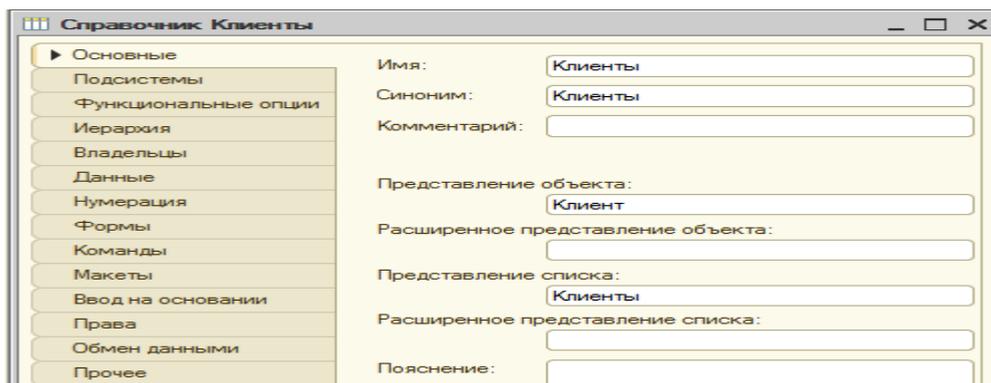


Рисунок 2.9 – Основные свойства объекта

Определим в каких подсистемах будет отображаться данный справочник. Задание принадлежности объекта к подсистемам представлено на рисунке 2.10.

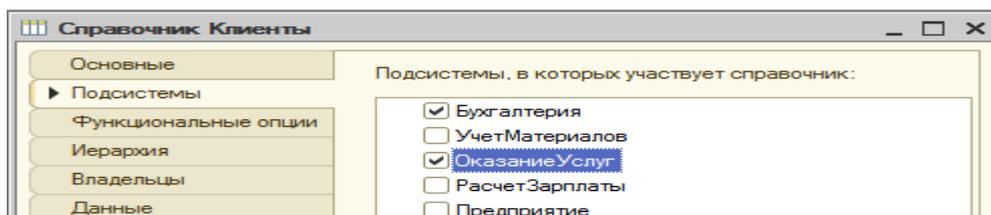


Рисунок 2.10 – Представление объекта в подсистемах

Укажем длину кода и наименования справочника. Заданные числа представлены на рисунке 2.11.

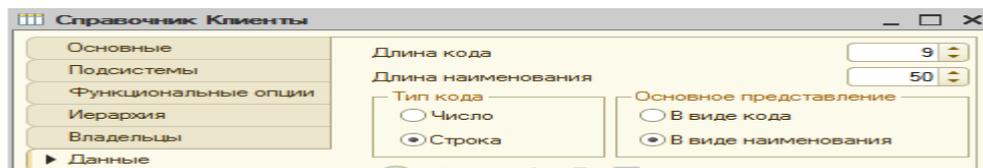


Рисунок 2.11 – Длина кода и наименования справочника

Команда добавления нового клиента.

Сделаем доступной в панели действия ОказаниеУслуг стандартную команду для создания новых клиентов. Включение доступности команды создания новых клиентов представлено на рисунке 2.12.

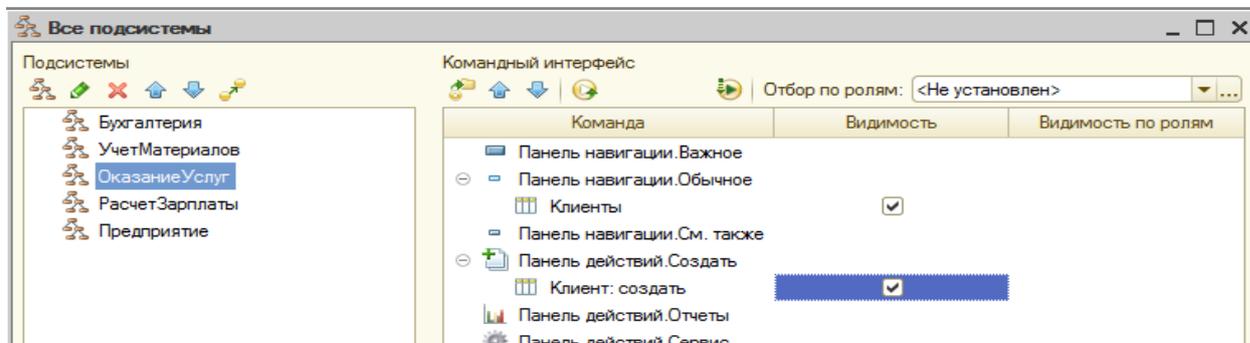


Рисунок 2.12 – Включение команды создания новых клиентов

В режиме «1С:Предприятие» внесём в список «Клиенты» новые наименования клиентов. Внесение новых данных представлено на рисунке 2.13.



Рисунок 2.13 – Новые данные в справочнике «Клиенты»

Справочник с табличной частью

Создадим новый справочник «Сотрудники». Заданные параметры представлены на рисунках 2.14, 2.15, 2.16.

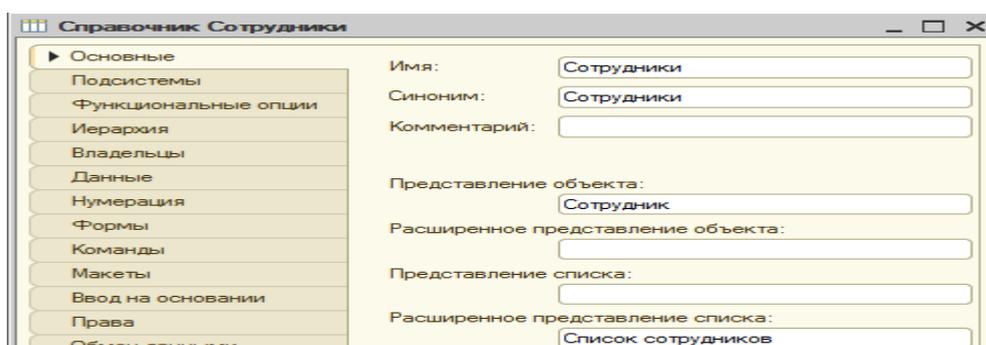


Рисунок 2.14 – Задание параметров нового справочника

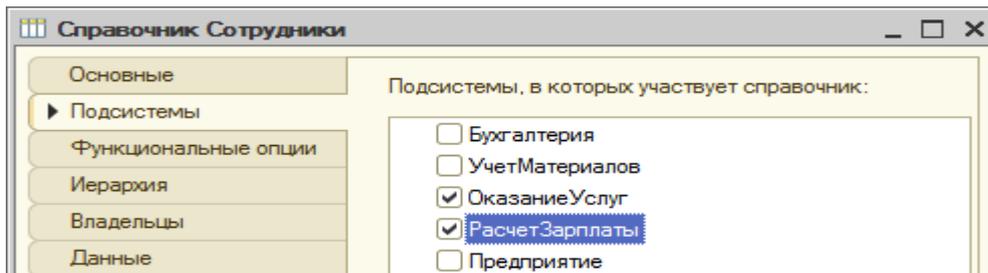


Рисунок 2.15 – Задание параметров нового справочника

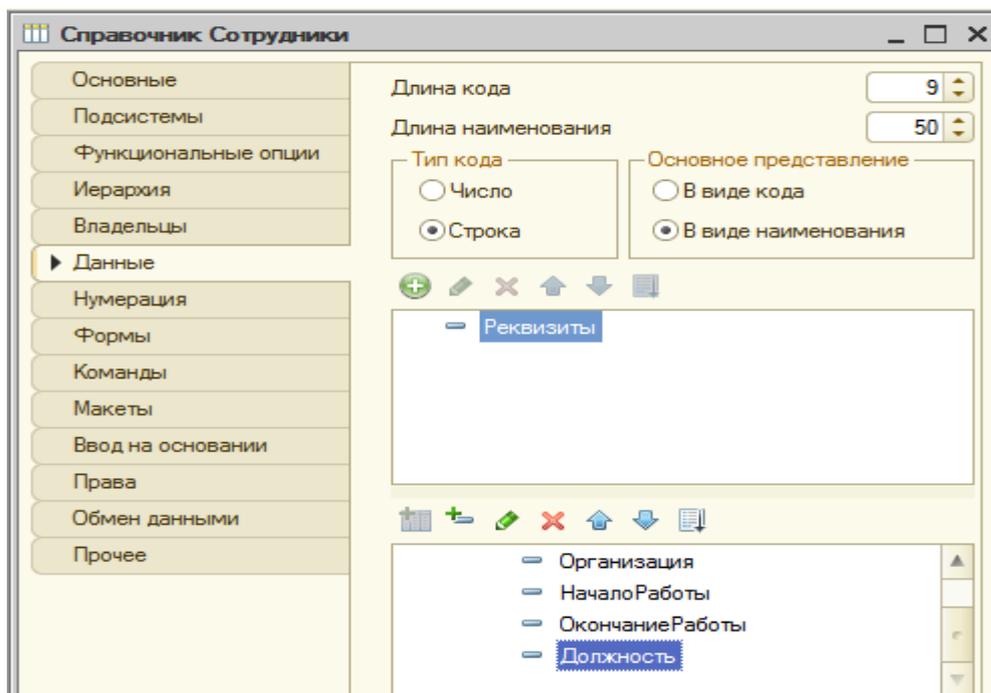


Рисунок 2.16 – Задание параметров нового справочника

В режиме «1С:Предприятие» заполним справочник «Сотрудники». Заполненный справочник представлен на рисунке 2.17.

| Список сотрудников | |
|----------------------------|----------|
| Наименование | Код |
| Гусakov Николай Дмитриевич | 00000001 |
| Деловой Иван Сергеевич | 00000002 |
| Симонов Валерий Михайлович | 00000003 |

Рисунок 2.17 – заполненный справочник «Сотрудники»

Сделаем видимой в панели действий подсистемы РасчетЗарплаты команду добавления новых сотрудников.

Иерархический справочник

Создадим справочник «Номенклатура». Заданный при создании параметры представлены на рисунках 2.18-2.21.

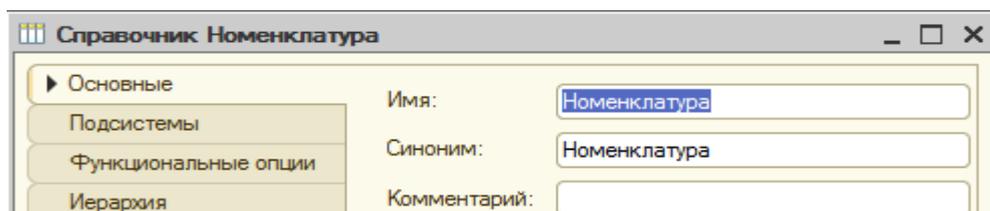


Рисунок 2.18 – Заданные параметры справочника «Номенклатура»

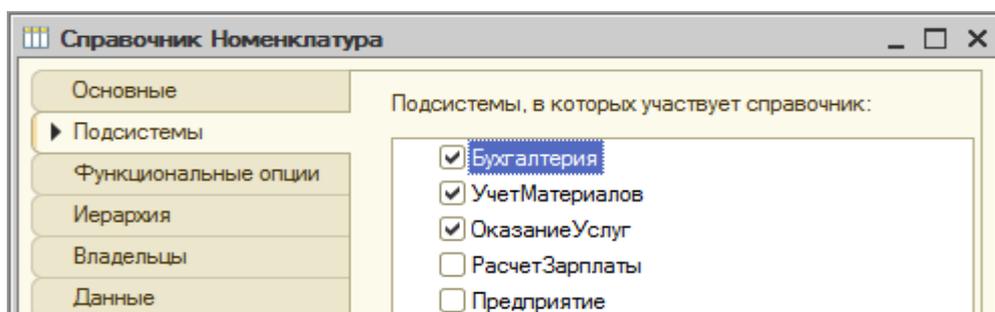


Рисунок 2.19 – Заданные параметры справочника «Номенклатура»

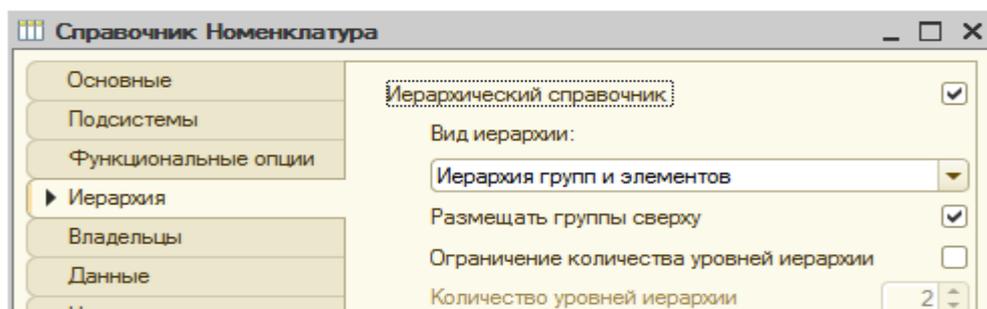


Рисунок 2.20 – Заданные параметры справочника «Номенклатура»

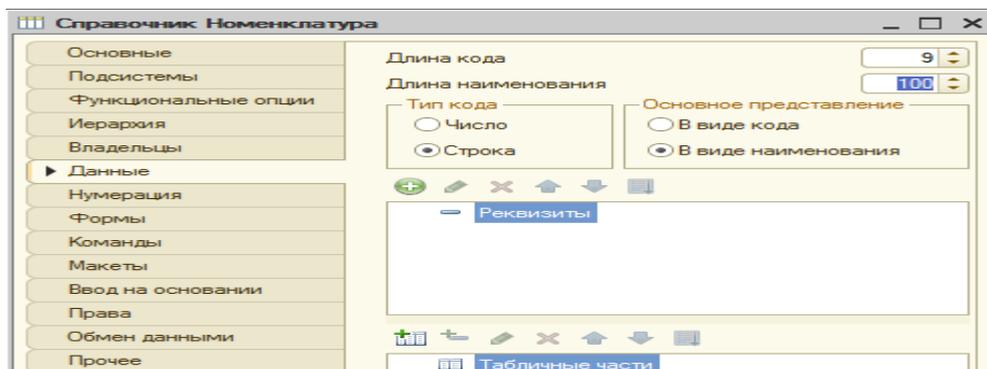


Рисунок 2.21 – Заданные параметры справочника «Номенклатура»

Сделаем видимой в панели действий разделов УчётМатериалов и ОказаниеУслуг команду создания новых элементов списка номенклатуры.

Заполним справочник «Номенклатура». Заполненный справочник представлен на рисунке 2.22.

| Наименование | Код |
|---------------------------------|-----------|
| Материалы | 000000001 |
| Кабель электрический | 000000007 |
| Строчный трансформатор GoldStar | 000000004 |
| Строчный трансформатор Samsung | 000000003 |
| Транзистор Phillips 2N2369 | 000000005 |
| Шланг резиновый | 000000006 |

| Наименование | Код |
|----------------------------------|-----------|
| Услуги | 000000002 |
| Диагностика | 000000008 |
| Подключение воды | 000000011 |
| Подключение электричества | 000000012 |
| Ремонт импортного телевизора | 000000010 |
| Ремонт отечественного телевизора | 000000009 |

Рисунок 2.22 – Заполненный справочник «Номенклатура»

Разнесем услуги по двум смысловым группам: услуги по ремонту телевизоров и услуги по установке стиральных машин, а материалы на радиодетали и прочее. Результат представлен на рисунке 2.23.

| | |
|----------------------------------|-----------|
| Номенклатура | |
| Материалы | 000000001 |
| Прочее | 000000016 |
| Кабель электрический | 000000007 |
| Шланг резиновый | 000000006 |
| Радиодетали | 000000015 |
| Строчный трансформатор GoldStar | 000000004 |
| Строчный трансформатор Samsung | 000000003 |
| Транзистор Phillips 2N2369 | 000000005 |
| Услуги | 000000002 |
| Стиральные машины | 000000013 |
| Подключение воды | 000000011 |
| Подключение электричества | 000000012 |
| Телевизоры | 000000014 |
| Диагностика | 000000008 |
| Ремонт импортного телевизора | 000000010 |
| Ремонт отечественного телевизора | 000000009 |

Рисунок 2.23 – разделенные по группам элементы «Номенклатуры»

Справочник с predetermined elements

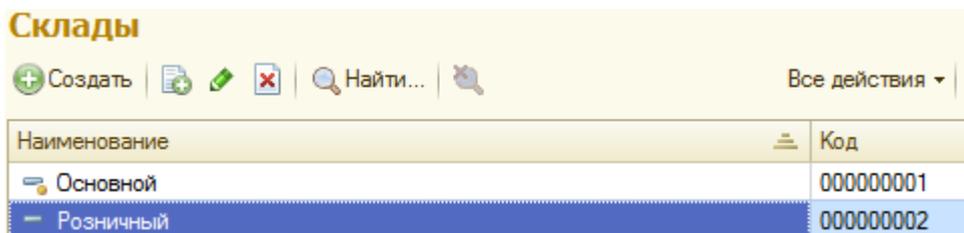
Create a reference «Warehouses». The specified parameters of the reference «Warehouses» are shown in Figure 2.24 and 2.25.

Figure 2.24 – Parameters of the reference «Warehouses»

Figure 2.25 – Parameters of the reference «Warehouses»

В панели действий раздела УчётМатериалов сделаем доступной команду создания новых складов.

Создадим новый элемент «Розничный» в справочнике «Склады». Справочник «Склады» представлен на рисунке 2.26.



| Наименование | Код |
|--------------|-----------|
| Основной | 000000001 |
| Розничный | 000000002 |

Рисунок 2.26 – справочник «Склады»

2.3 Документы

Документ «Приходная накладная»

Создадим документ «Приходная накладная». Заданные параметры документа «Приходная накладная» представлены на рисунке 2.27.

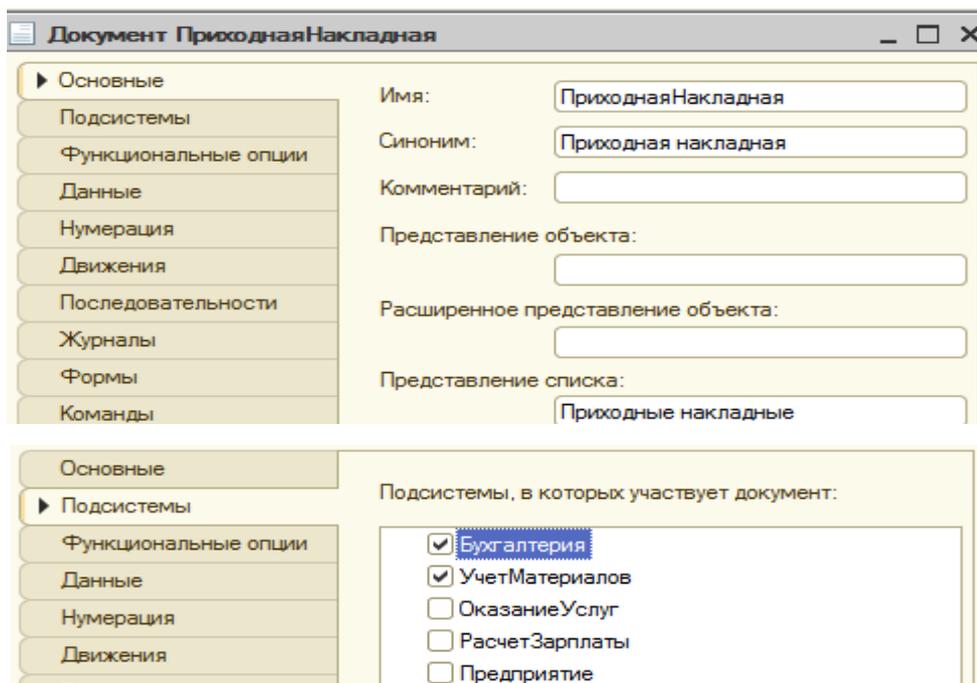


Рисунок 2.27 – Параметры документа «Приходная накладная»

Создадим реквизит документа с именем «Склад». Параметры созданного реквизита представлены на рисунке 2.28.

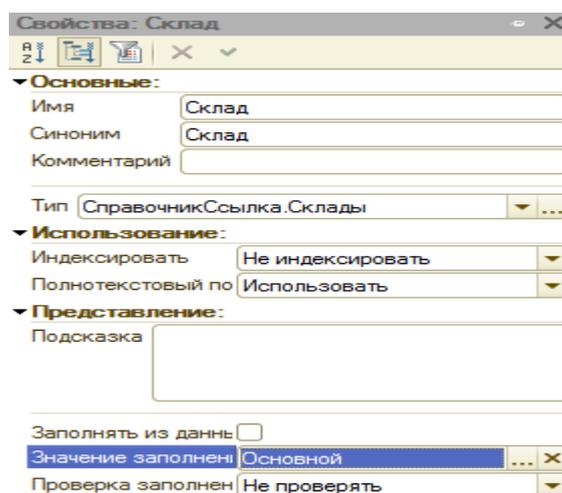


Рисунок 2.28 – Параметры реквизита «Склад»

Добавим в документ табличную часть с именем «Материалы». Параметры табличной части представлены на рисунке 2.29.

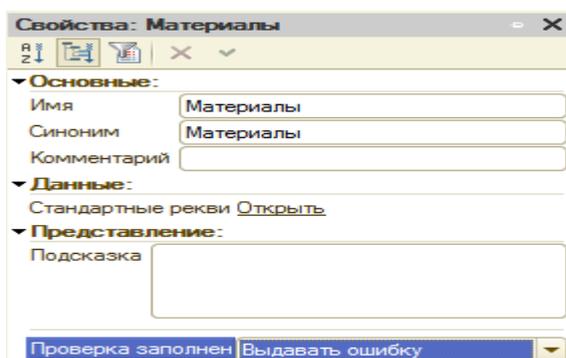


Рисунок 2.29 – Параметры табличной части документа

Создадим реквизиты табличной части «Материалы». Реквизиты представлены на рисунке 2.30.

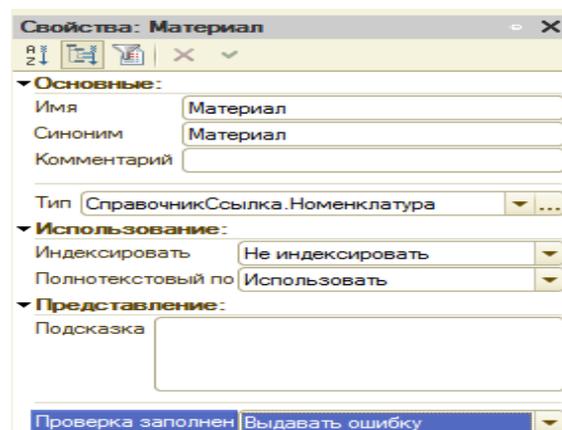
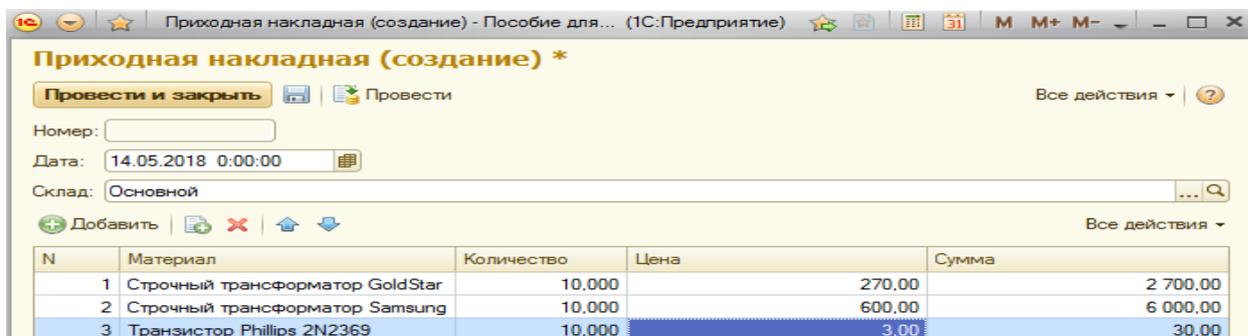


Рисунок 2.30 – Реквизиты табличной части «Материалы»

Отредактируем командный интерфейс так, чтобы в подсистеме «Учет материалов» была доступна команда создания новых документов.

Создадим несколько «Приходных накладных». Созданные накладные представлены на рисунке 2.31.



| N | Материал | Количество | Цена | Сумма |
|---|---------------------------------|------------|--------|----------|
| 1 | Строчный трансформатор GoldStar | 10,000 | 270,00 | 2 700,00 |
| 2 | Строчный трансформатор Samsung | 10,000 | 600,00 | 6 000,00 |
| 3 | Транзистор Phillips 2N2369 | 10,000 | 3,00 | 30,00 |

Рисунок 2.31 – Приходные накладные

Сделаем автоматический пересчет суммы в строках документа. Автоматический пересчет представлен на рисунке 2.32.

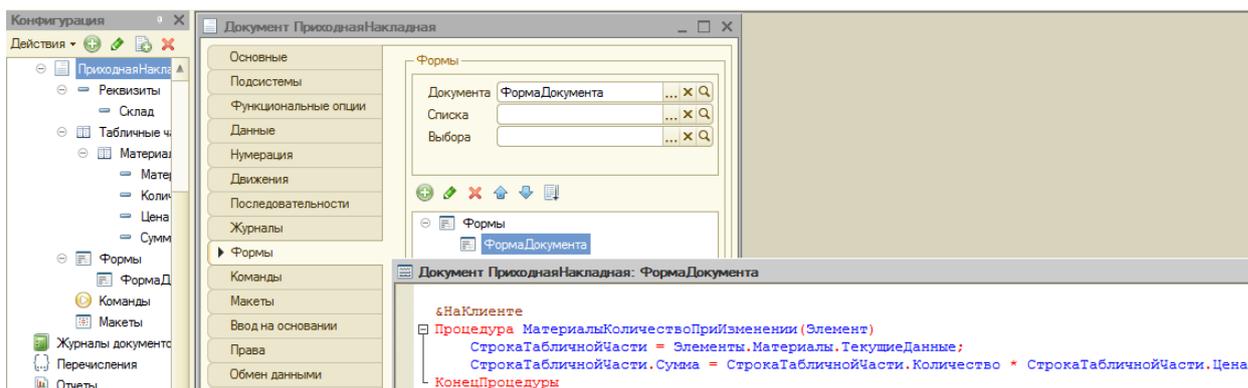


Рисунок 2.32 – Автоматический пересчет суммы

Одна процедура для обработки нескольких событий.

В «Общих модулях» создадим модуль «РаботаСДокументами» и пропишем в нём код, который представлен на рисунке 2.33.

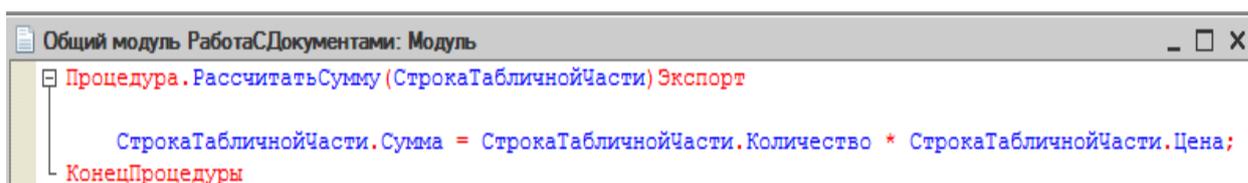


Рисунок 2.33 – Код процедуры «Рассчитать сумму»

Теперь в модуле нашей формы изменим текст обработчика. Новый текст обработчика представлен на рисунке 2.34.

```
Документ ПриходнаяНакладная: ФормаДокумента
    &НаКлиенте
    Процедура МатериальКоличествоПриИзменении (Элемент)
        СтрокаТабличнойЧасти = Элементы.Материалы.ТекущиеДанные;
        РаботаСДокументами.РассчитатьСумму (СтрокаТабличнойЧасти);
    КонечПроцедуры
```

Рисунок 2.34 – Текст обработчика

Документ «Оказание услуги»

Аналогично созданию документа «Приходная накладная» создадим документ «Оказание услуги».

Заполним новый документ «Оказание услуги». Заполненный документ представлен на рисунке 2.35.

| N | Номенклатура | Количество | Цена | Сумма |
|---|----------------------------|------------|------|-------|
| 1 | Транзистор Phillips 2N2369 | 1,000 | 3,00 | 3,00 |

Рисунок 2.35 – Заполненный документ «Оказание услуги»

2.4 Регистры накопления

Добавим регистр накопления «ОстаткиМатериалов». Заданные параметры представлены на рисунке 2.36.

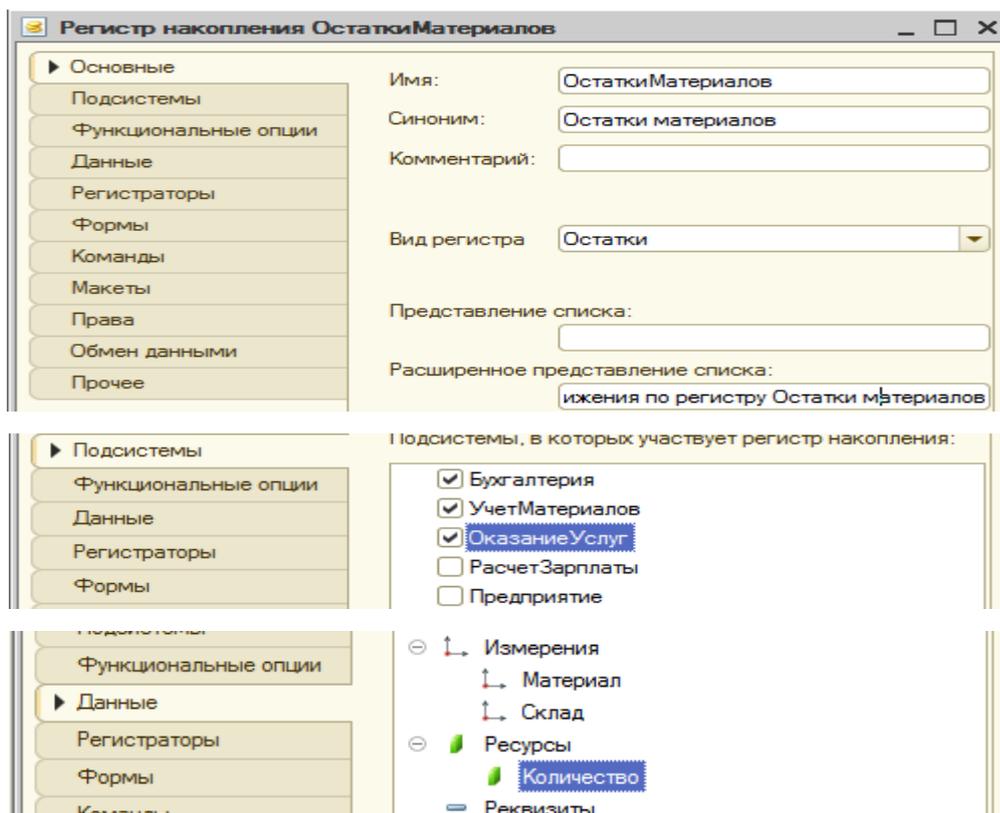


Рисунок 2.36 – Параметры регистра накопления «ОстаткиМатериалов»

Настроим движение документа. Заданные параметры представлены на рисунке 2.37.

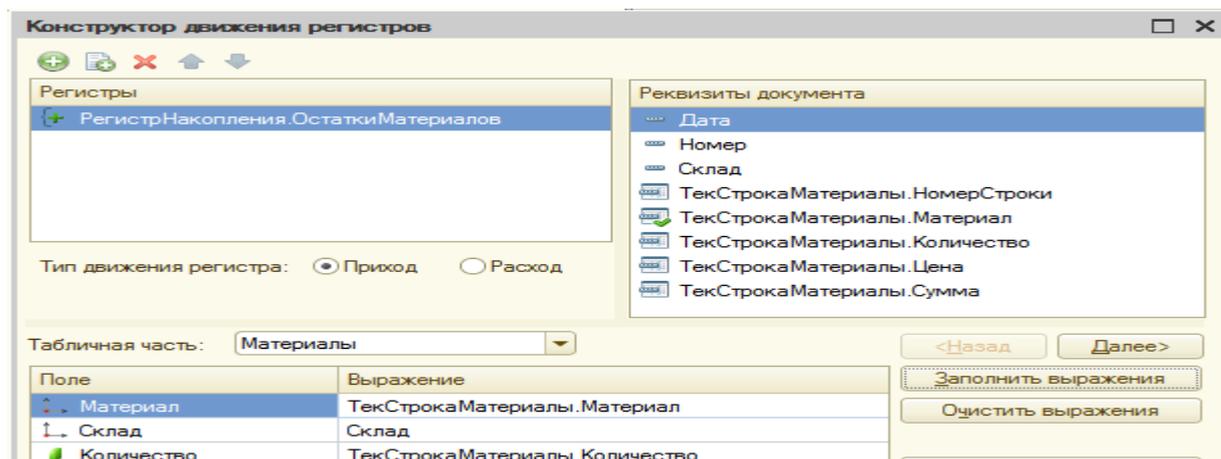


Рисунок 2.37 – Параметры движения документа

В итоге конструктор сформировал текст в модуле документа «ПриходнаяНакладная» представленный на рисунке 2.38.

```

□ Процедура ОбработкаПроведения(Отказ, Режим)
  //{{_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
  // Данный фрагмент построен конструктором.
  // При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!

  // регистр ОстаткиМатериалов Приход
  Движения.ОстаткиМатериалов.Записывать = Истина;
  Для Каждого ТекСтрокаМатериалы Из Материалы Цикл
    Движение = Движения.ОстаткиМатериалов.Добавить ();
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Материал = ТекСтрокаМатериалы.Материал;
    Движение.Склад = Склад;
    Движение.Количество = ТекСтрокаМатериалы.Количество;
  КонечЦикла;

  //}}_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
  КонечПроцедуры

```

Рисунок 2.38 – Текст, сформированный конструктором

Отредактируем командный интерфейс, чтобы в подсистемах «Бухгалтерия», «Оказание услуг» и «Учет материалов» была доступна ссылка для просмотра записей регистра накоплений.

Сделаем возможность вызова из формы документа список регистра, в котором показаны движения, произведенные только этим документом. Настройки представлены на рисунке 2.39.

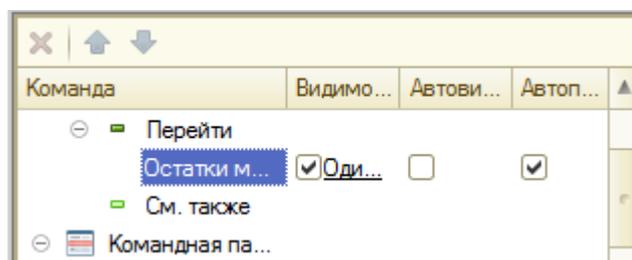


Рисунок 2.39 – Настройки документа

Аналогичным образом создадим движение документа «Оказание услуги».

2.5 Простой отчет

Создадим отчет, который будет нам показывать приход, расход и остатки материалов.

Зададим схему компоновки данных. Компанова данных представлена на рисунках 2.40 – 2.43.

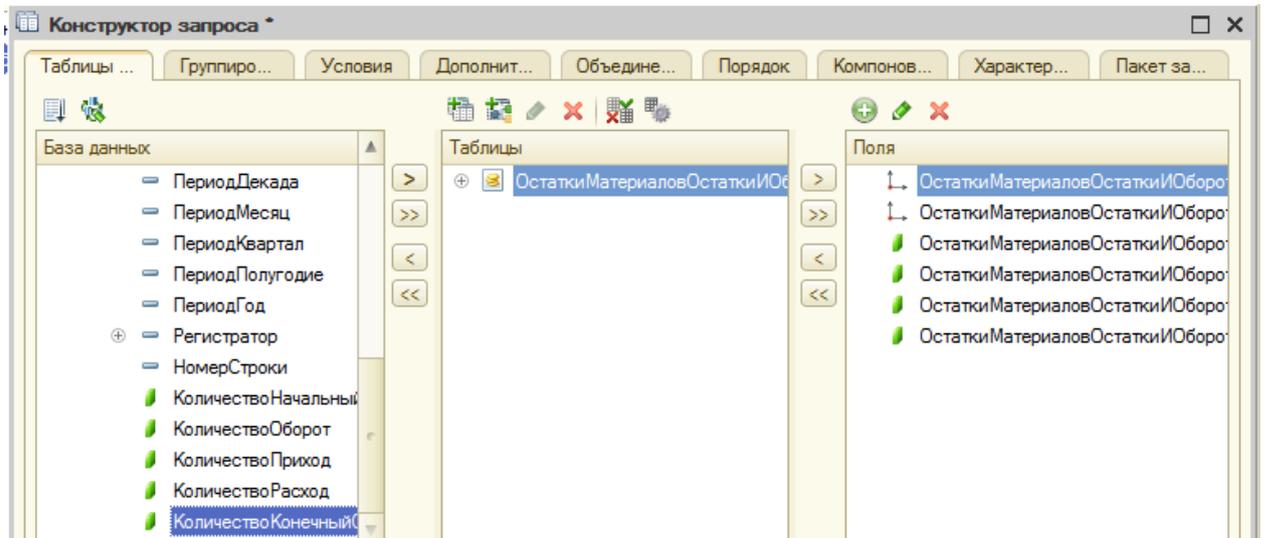


Рисунок 2.40 – Задание схемы компоновки данных

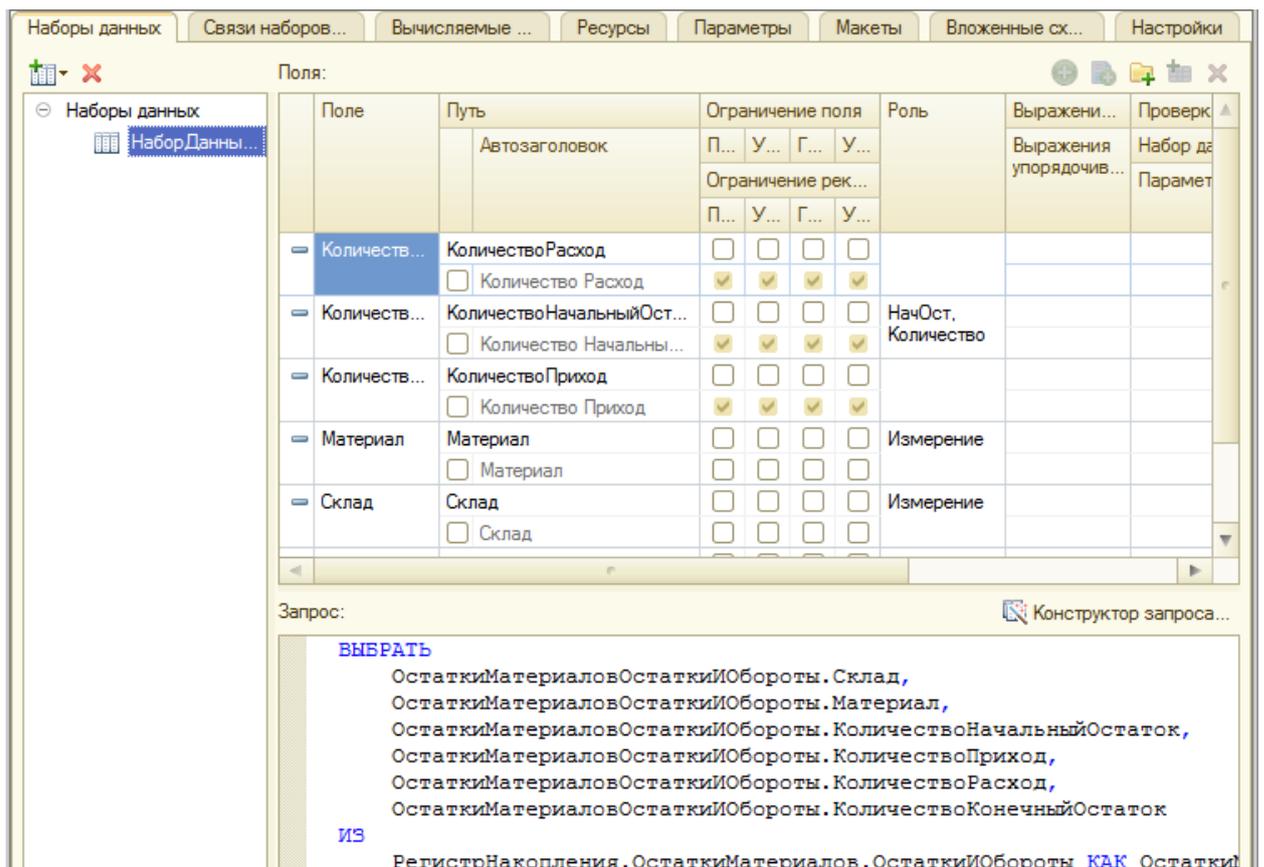


Рисунок 2.41 – Задание схемы компоновки данных

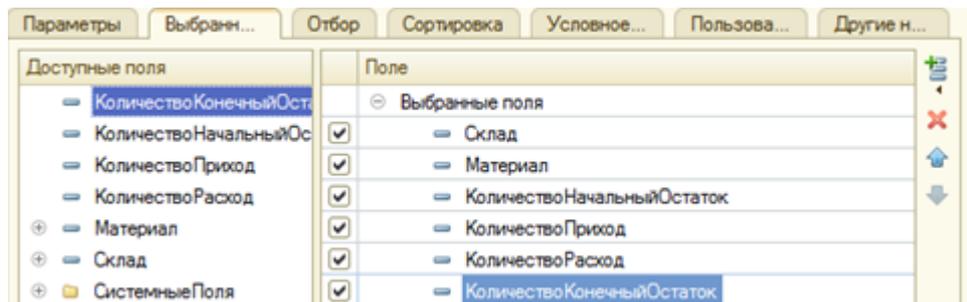


Рисунок 2.42 – Задание схемы компоновки данных

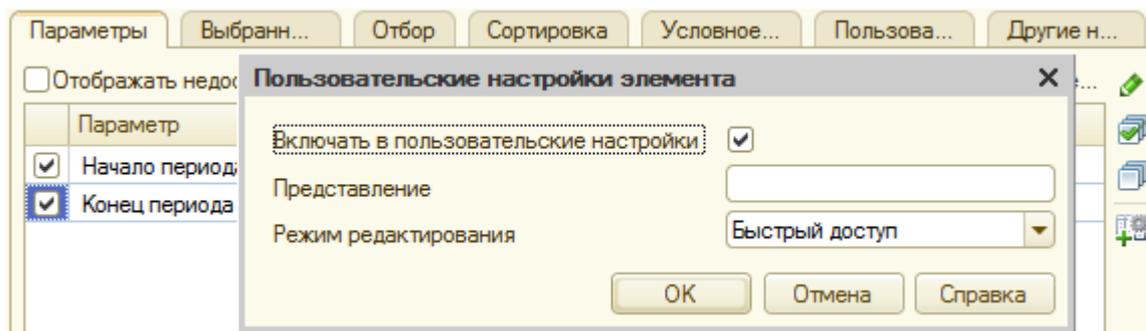


Рисунок 2.43 – Задание схемы компоновки данных

Зададим подсистемы в которых участвует отчет: «Бухгалтерия», «УчётМатериалов», «ОказаниеУслуг».

2.6 Макеты. Редактирование макетов и форм

Макет печатной формы

Создадим макет печатной формы документа «Оказание услуги». Задание параметров конструктора печати представлено на рисунках 2.44, 2.45.

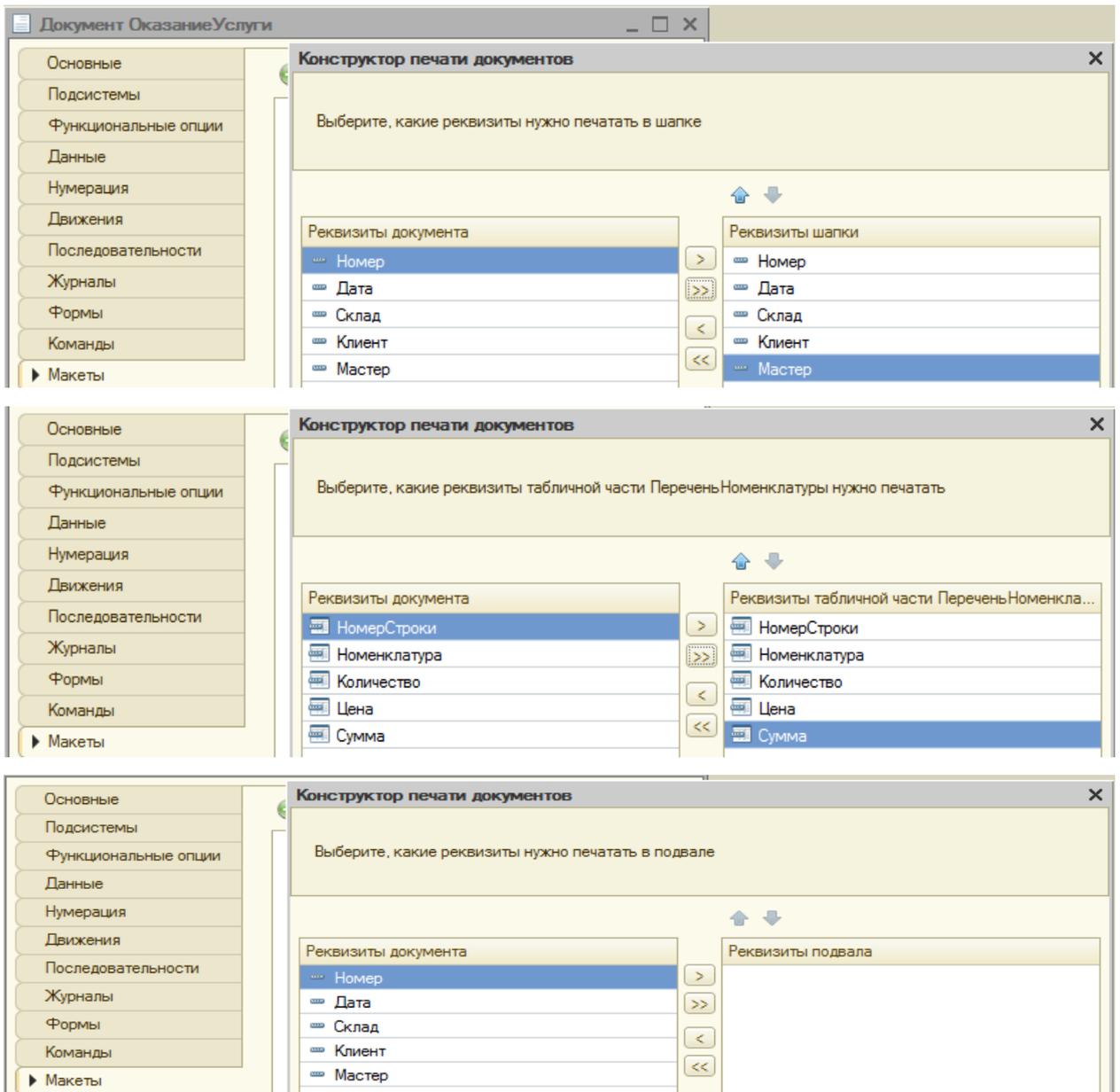


Рисунок 2.44 – Задание параметров конструктора печати

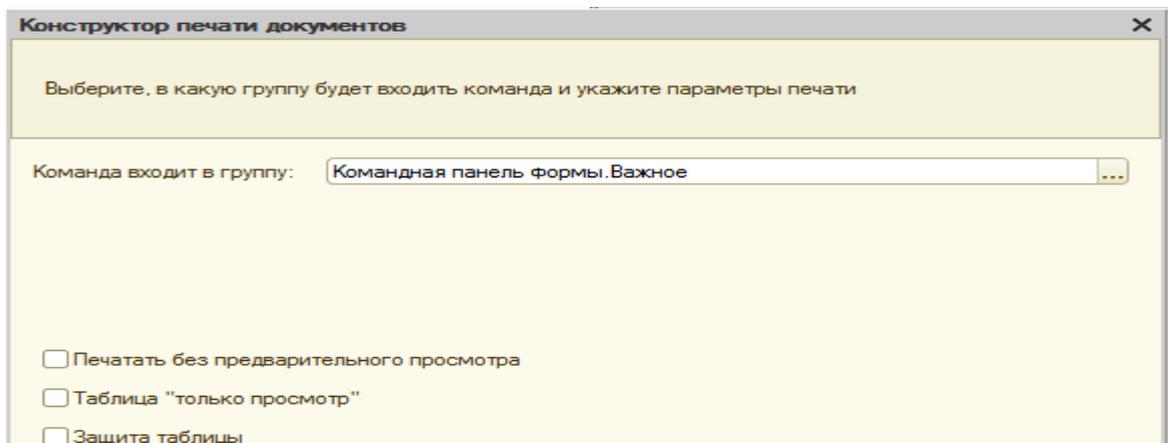


Рисунок 2.45 – Задание параметров конструктора печати

Редактирование макета

Добавим итоговую сумму в печатную форму документа. Добавление представлено на рисунке 2.46.

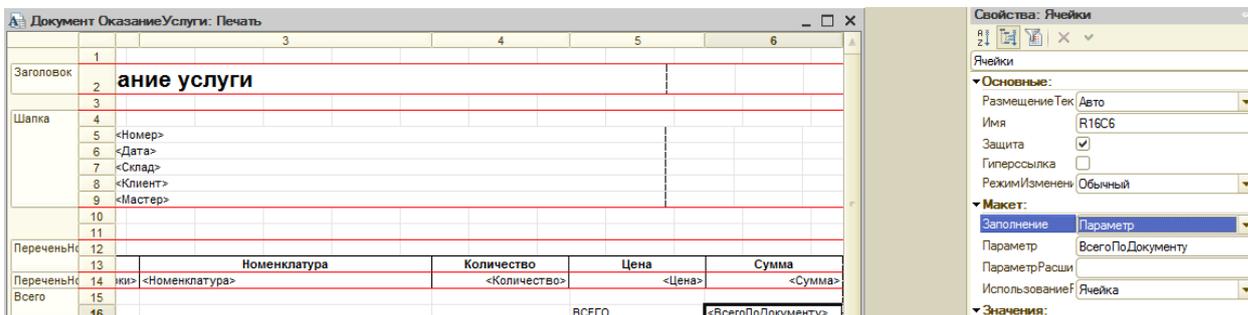


Рисунок 2.46 – Добавление итоговой суммы в печатную форму документа

Добавим строки, выделенные знаком «>>>>» в тело процедуры «Печать» в модуле менеджера документа «Оказание услуги». Измененная процедура представлена на рисунке 2.46.

```
ОбластьЗаголовок = Макет.ПолучитьОбласть ("Заголовок");
Шапка = Макет.ПолучитьОбласть ("Шапка");
ОбластьПереченьНоменклатурыШапка = Макет.ПолучитьОбласть ("ПереченьНоменклатурыШапка");
ОбластьПереченьНоменклатуры = Макет.ПолучитьОбласть ("ПереченьНоменклатуры");
ОбластьИтог = Макет.ПолучитьОбласть ("Всего");
>>> ТабДок.Очистить ();

ВставлятьРазделительСтраниц = Ложь;
Пока Выборка.Следующий () Цикл
    Если ВставлятьРазделительСтраниц Тогда
        ТабДок.ВывестиГоризонтальныйРазделительСтраниц ();
    КонецЕсли;

    ТабДок.Вывести (ОбластьЗаголовок);

    Шапка.Параметры.Заполнить (Выборка);
    ТабДок.Вывести (Шапка, Выборка.Уровень ());

    ТабДок.Вывести (ОбластьПереченьНоменклатурыШапка);
    ВыборкаПереченьНоменклатуры = Выборка.ПереченьНоменклатуры.Выбрать ();
    СуммаИтог = 0;
    Пока ВыборкаПереченьНоменклатуры.Следующий () Цикл
        ОбластьПереченьНоменклатуры.Параметры.Заполнить (ВыборкаПереченьНоменклатуры);
        ТабДок.Вывести (ОбластьПереченьНоменклатуры, ВыборкаПереченьНоменклатуры.Уровень ());
    >>> СуммаИтог = СуммаИтог + ВыборкаПереченьНоменклатуры.Сумма;
    КонецЦикла;

>>> ОбластьИтог.Параметры.ВсегоПоДокументу = СуммаИтог;
>>> ТабДок.Вывести (ОбластьИтог);
```

Рисунок 2.47 – Процедура «Печать»

Редактирование формы

Сделаем возможность просмотра суммы документа не печатая его. Для этого добавим «подвал» у документа и отредактируем свойства полей «Цена», и «Сумма». Результат представлен на рисунке 2.48.

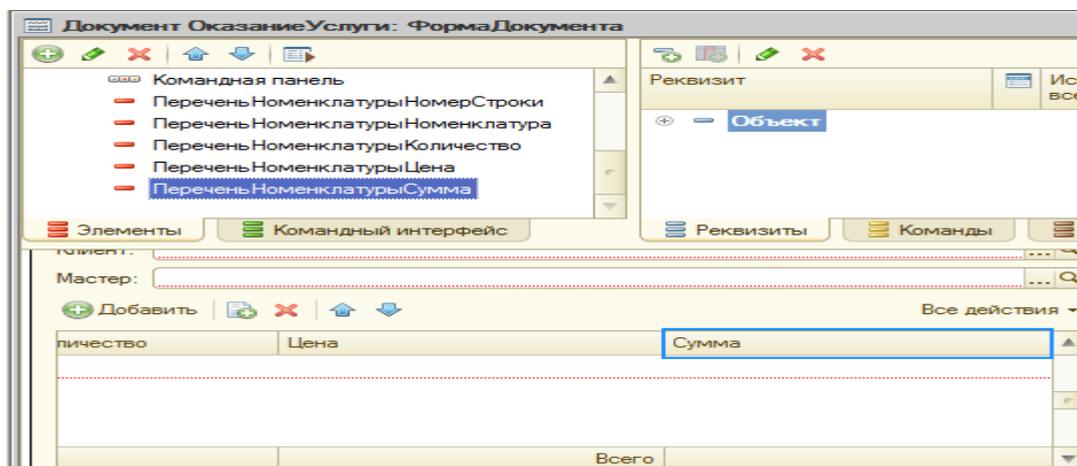


Рисунок 2.48 – Измененная форма документа

2.7 Периодические регистры сведений

Создадим периодический регистр сведений, который будет хранить развернутые во времени розничные цены материалов и стоимости услуг. Заданные при создании свойства представлены на рисунке 2.49.

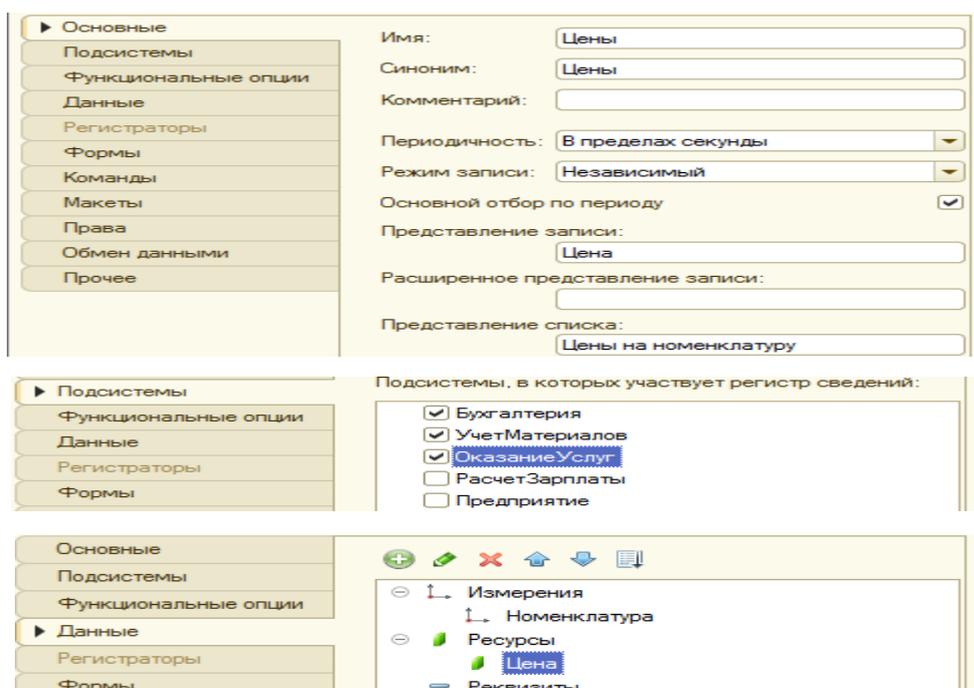


Рисунок 2.49 – Свойства периодического регистра сведений

Автоматическая подстановка цены в документ при выборе номенклатуры

Создадим общий модуль «РаботаСоСправочниками». Создание представлено на рисунке 2.50.

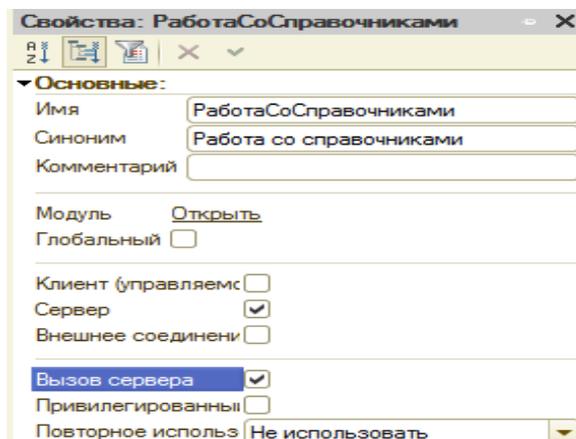


Рисунок 2.50 – Общий модуль «РаботаСоСправочниками»

В модуле «РаботаСоСправочниками» прописываем следующую функцию «РозничнаяЦена()» представленную на рисунке 2.50.

```
□ функция РозничнаяЦена(АктуальнаяДата, ЭлементНоменклатуры) Экспорт
    //Создать вспомогательный объект Отбор
    Отбор = Новый Структура("Номенклатура", ЭлементНоменклатуры);

    //Получить актуальные значения ресурсов регистра
    ЗначенияРесурсов = РегистрыСведений.Цены.ПолучитьПоследнее(АктуальнаяДата, Отбор);
    Возврат ЗначенияРесурсов.Цена;
Конецфункции
```

Рисунок 2.51 – Функция «РозничнаяЦена()»

2.8 Перечисления

Создадим новый объект конфигурации Перечисление с именем «ВидНоменклатуры». Свойства нового объекта представлены на рисунке 2.52.

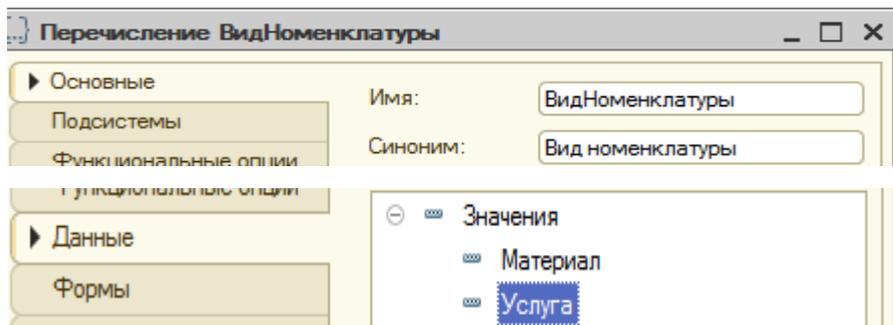


Рисунок 2.52 – Свойства объекта «ВидНоменклатуры»

Привязка номенклатуры к значениям перечисления «ВидНоменклатуры»

В справочнике «Номенклатура» добавим реквизит «ВидНоменклатуры». Свойства реквизита «ВидНоменклатуры» представлены на рисунке 2.53.

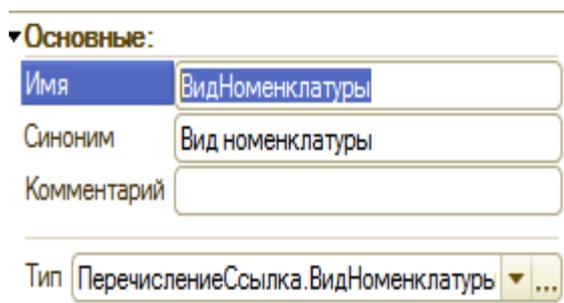


Рисунок 2.53 – Реквизит «ВидНоменклатуры»

Регистрация только той номенклатуры, которая является материалом
 Добавим в процедуру «ОбработкаПроведения» строки отмеченные знаком «>>>» на рисунке 2.54.

```

    Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл
    >>> Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры = Перечисления.ВидНоменклатуры.Материал Тогда
        Движение = Движения.ОстаткиМатериалов.Добавить ();
        Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
        Движение.Склад = Склад;
        Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;
    >>>| КонецЕсли;
        КонецЦикла;
    
```

Рисунок 2.54 – Измененная процедура «Обработка проведения»

2.9 Проведение документа по нескольким регистрам

По аналогии с регистром «ОстаткиМатериалов» создадим новый регистр накопления «СтоимостьМатериалов».

Проведение приходной накладной по двум регистрам

Изменим процедуру проведения. Отметим что документ будет создавать движение и по регистру «СтоимостьМатериалов». Измененная процедура «ОбработкаПроведения» представлена на рисунке 2.55.

```
Процедура ОбработкаПроведения(Отказ, Режим)
  //{{_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
  // Данный фрагмент построен конструктором.
  // При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!!

  // регистр ОстаткиМатериалов Приход
  Движения.ОстаткиМатериалов.Записывать = Истина;
  Движения.СтоимостьМатериалов.Записывать = Истина;
  Для Каждого ТекСтрокаМатериалы Из Материалы Цикл
    Движение = Движения.ОстаткиМатериалов.Добавить ();
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Материал = ТекСтрокаМатериалы.Материал;
    Движение.Склад = Склад;
    Движение.Количество = ТекСтрокаМатериалы.Количество;

  //регистр СтоимостьМатериалов Приход
  Движение = Движения.СтоимостьМатериалов.Добавить ();
  Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
  Движение.Период = Дата;
  Движение.Материал = ТекСтрокаМатериалы.Материал;
  Движение.Стоимость = ТекСтрокаМатериалы.Сумма;
  КонецЦикла;

  //}}_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
КонецПроцедуры
```

Рисунок 2.55 – Процедура «ОбработкаПроведения»

Отредактируем командный интерфейс документа, чтобы иметь возможность переходить к списку записей регистра «СтоимостьМатериалов».

Проведение документа «Оказание услуги» по двум регистрам.

Сначала добавим в табличную часть документа новый реквизит «Стоимость». Затем изменим процедуру проведения документа. Измененная процедура проведения документа представлена на рисунке 2.56.

```

Процедура ОбработкаПроведения(Отказ, Режим)
//{{_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!!

// регистр ОстаткиМатериалов Расход
Движения.ОстаткиМатериалов.Записывать = Истина;
Движения.СтоимостьМатериалов.Записывать = Истина;
Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл
    Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры = Перечисления.ВидНоменклатуры.Материал Тогда
        Движение = Движения.ОстаткиМатериалов.Добавить ();
        Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
        Движение.Склад = Склад;
        Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;

        //регистр СтоимостьМатериалов Расход
        Движение = Движения.ОстаткиМатериалов.Добавить ();
        Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
        Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Количество * ТекСтрокаПереченьНоменклатуры.Стоимость;
    КонецЕсли;
КонецЦикла;

//}}_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
КонецПроцедуры

```

Рисунок 2.56 – Процедура проведения документа

Отредактируем командный интерфейс документа, чтобы иметь возможность переходить к списку записей регистра «СтоимостьМатериалов».

2.10 Оборотные регистры накопления

Создадим регистр накопления «Продажи», указав его тип как «Оборотный».

Отредактируем командный интерфейс документа, чтобы в подсистемах «Бухгалтерия», «Оказание услуг» и «Учет материалов» была доступна ссылка для просмотра оборотного регистра накопления.

Проведение документа «Оказание услуги» по трем регистрам

Укажем, что документ «Оказание услуги» будет создавать движение и по регистру «Продажи».

Изменим процедуру проведения документа. Измененная процедура представлена на рисунке 2.57.

```

Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры = Перечисления.ВидНоменклатуры.Материал Тогда
    Движение = Движения.ОстаткиМатериалов.Добавить ();
    Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
    Движение.Период = Дата;
    Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
    Движение.Склад = Склад;
    Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;

//регистр СтоимостьМатериалов Расход
Движение = Движения.СтоимостьМатериалов.Добавить ();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Количество * ТекСтрокаПереченьНоменклатуры.Стоимость;

КонецЕсли;

//регистр Продажи
Движение = Движения.Продажи.Добавить ();
Движение.Период = Дата;
Движение.Номенклатура = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Клиент = Клиент;
Движение.Мастер = Мастер;
Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;
Движение.Выручка = ТекСтрокаПереченьНоменклатуры.Сумма;
Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Стоимость * ТекСтрокаПереченьНоменклатуры.Количество;
endif;

```

Рисунок 2.57 – Процедура проведения

Отредактируем командный интерфейс документа, чтобы иметь возможность переходить к списку записей регистра «Продажи».

Создадим отчет «Реестр документов оказание услуги». После задания параметров в конструкторе запроса получим запрос, представленный на рисунке 2.58.

```

ВЫБРАТЬ
    ОказаниеУслуги.Склад,
    ОказаниеУслуги.Мастер,
    ОказаниеУслуги.Клиент,
    ОказаниеУслуги.Ссылка КАК Документ
ИЗ
    Документ.ОказаниеУслуги КАК ОказаниеУслуги

УПОРЯДОЧИТЬ ПО
    Документ

```

Рисунок 2.58 – Запрос, сформированный конструктором запросов

Зададим настройки, определяющие, как будет выводиться информация в отчет. Настройки представлены на рисунке 2.59.

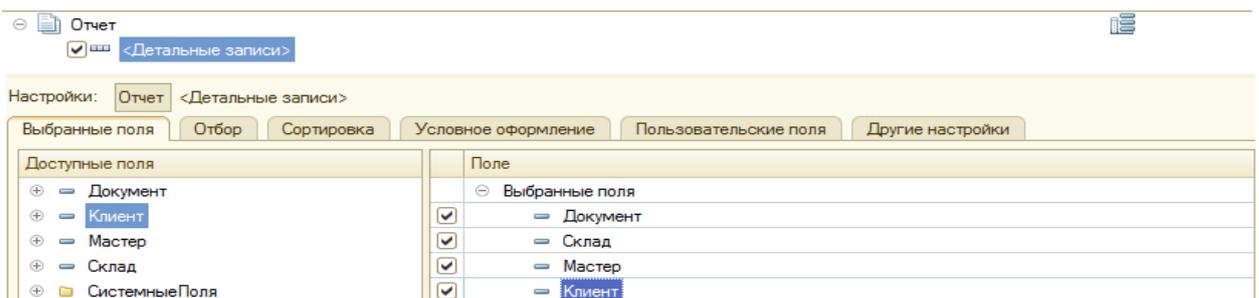


Рисунок 2.59 – Настройки отображения информации

Отметим на закладке «Подсистемы» подсистему «ОказаниеУслуг» для того, чтобы ссылка на отчет попала на панель действий подсистемы.

Выбор данных из двух таблиц

Создадим отчет «Рейтинг услуг», который будет отражать список услуг, принесших наибольшую прибыль в указанном периоде. Параметры создания отчета представлены на рисунках 2.60-2.68.

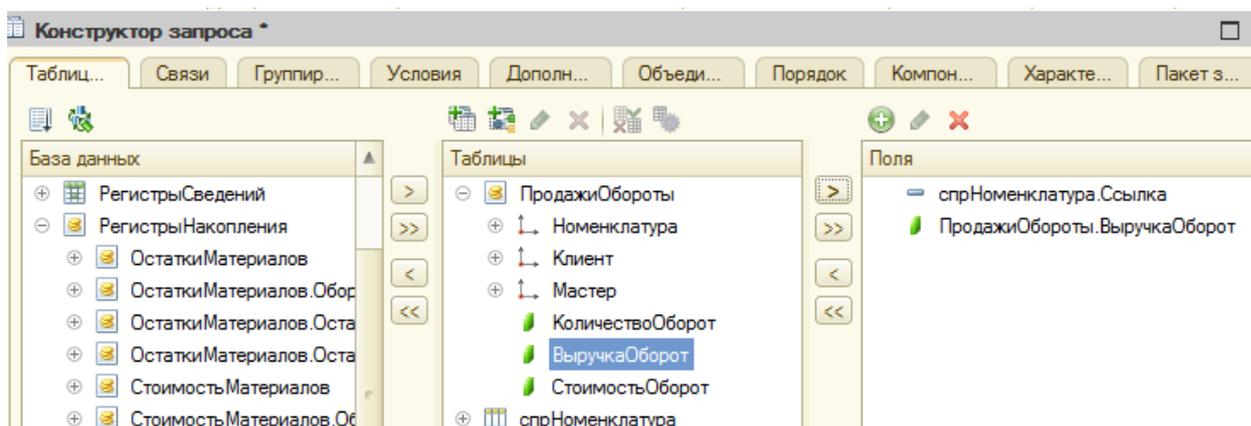


Рисунок 2.60 – Параметры отчета

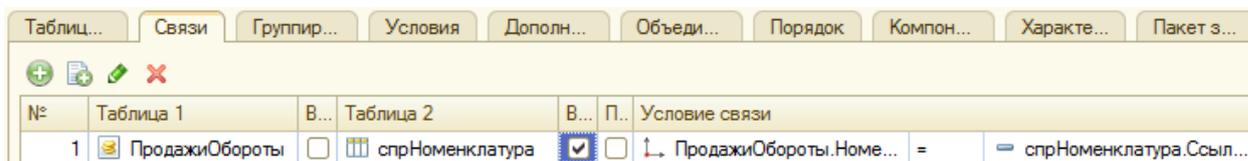


Рисунок 2.61 – Параметры отчета

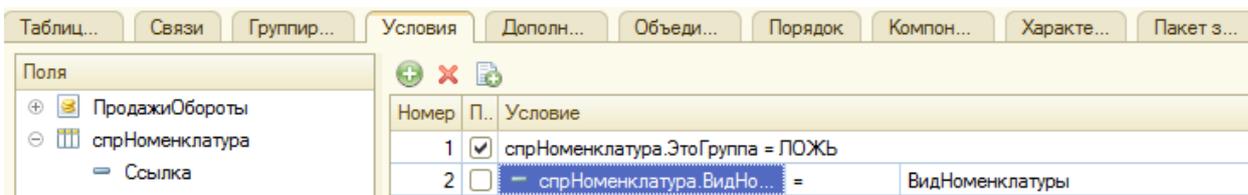


Рисунок 2.62 – Параметры отчета

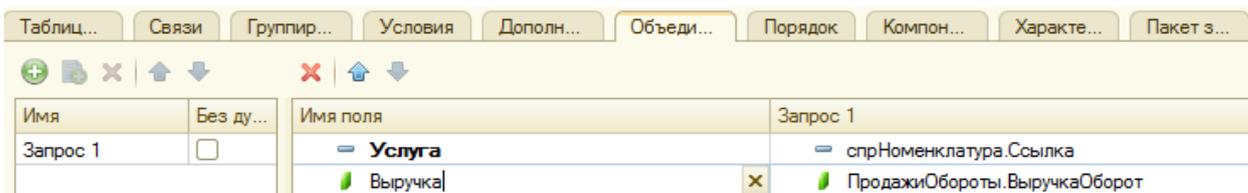


Рисунок 2.63 – Параметры отчета

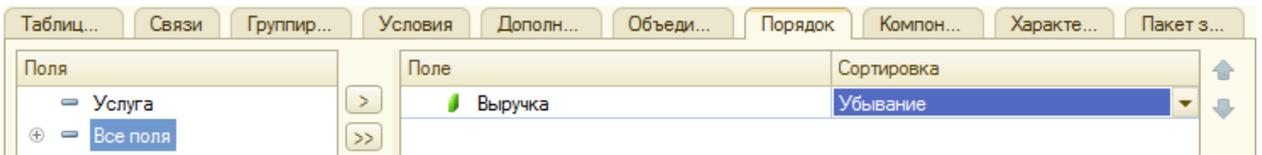


Рисунок 2.64 – Параметры отчета

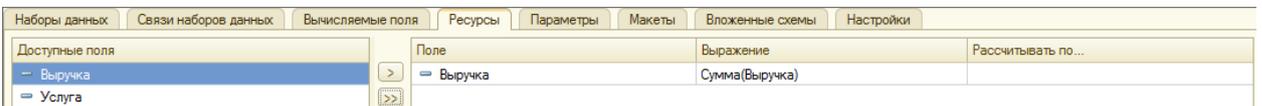


Рисунок 2.65 – Параметры отчета

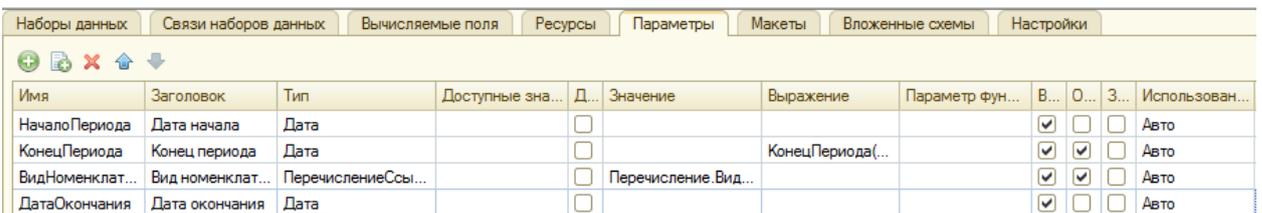


Рисунок 2.66 – Параметры отчета

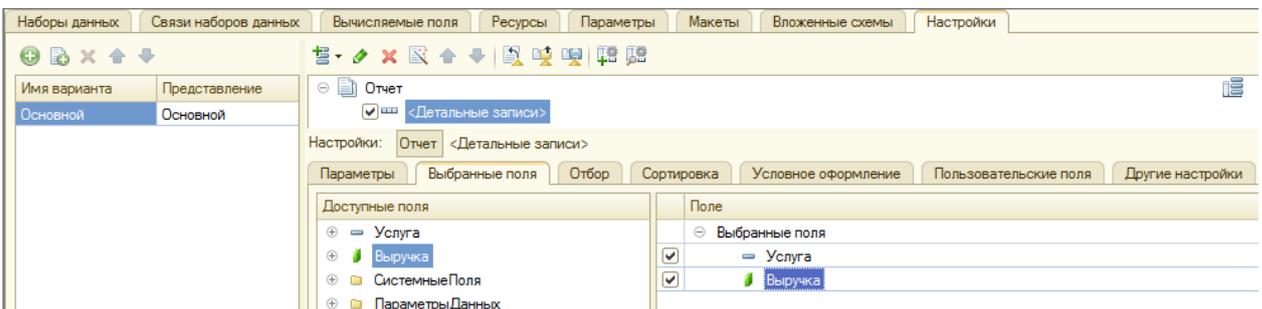


Рисунок 2.67 – Параметры отчета

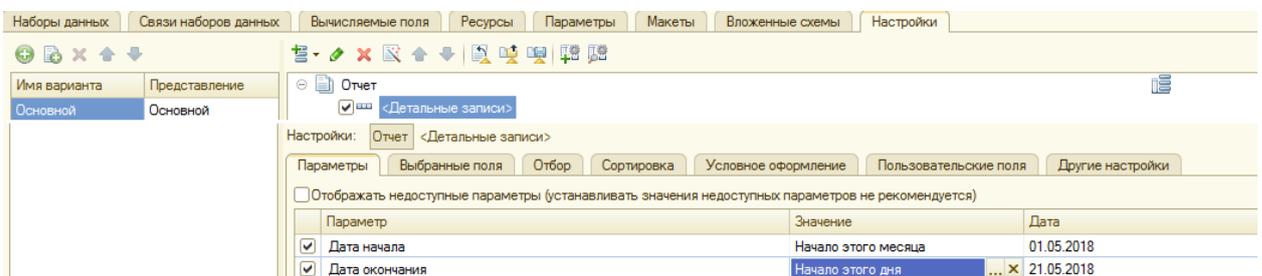


Рисунок 2.68 – Параметры отчета

Условное форматирование

Зададим параметры условного форматирования. Заданные параметры представлены на рисунке 2.69.

| Оформление | Условие | Оформляемые поля | Представление |
|---|----------------------|------------------|---------------------|
| <input checked="" type="checkbox"/> Цвет текста | Выручка Меньше "700" | Услуга, Выручка | Непопулярная услуга |

Рисунок 2.69 – Параметры условного форматирования

Пользовательские настройки

Предоставим пользователю часть полных настроек, которые задает разработчик. Предоставление представлено на рисунке 2.70.

| | Представление | Режим редактирования |
|---|----------------------|----------------------|
| <input type="checkbox"/> Выбранные поля | <input type="text"/> | Быстрый доступ |
| <input checked="" type="checkbox"/> Отбор | <input type="text"/> | Обычный |
| <input type="checkbox"/> Порядок | <input type="text"/> | Быстрый доступ |
| <input checked="" type="checkbox"/> Условное оформление | <input type="text"/> | Обычный |
| <input type="checkbox"/> Группировки | <input type="text"/> | Быстрый доступ |

OK Отмена Справка

Рисунок 2.70 – Состав пользовательских настроек отчета

Отбор

Создадим настройку отбора в отчете. Параметры представлены на рисунке 2.71.

| Доступные поля | Левое значение | Вид сравнения | Правое значение | Режим отобр... | Представление |
|---|---|---------------|--------------------|----------------|---------------|
| <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Номенклатура <input type="checkbox"/> Услуга | Отбор | | | | |
| | <input checked="" type="checkbox"/> Услуга... | Равно | Справочник.Номе... | Быстрый дос... | |

Рисунок 2.71 – Параметры отбора

Вывод данных по всем дням выбранного периода

Создадим отчет «ВыручкаМастеров» по подобию прошлого отчета, но укажем периодичность в один день. Задание периодичности представлено на рисунках 2.72 и 2.73.

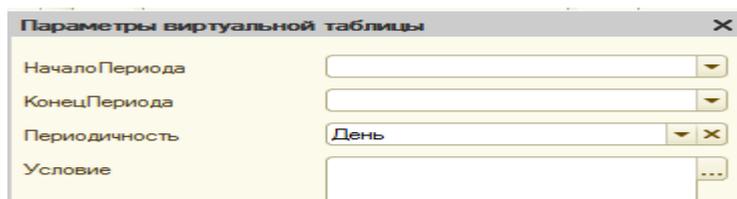


Рисунок 2.72 – Задание периодичности

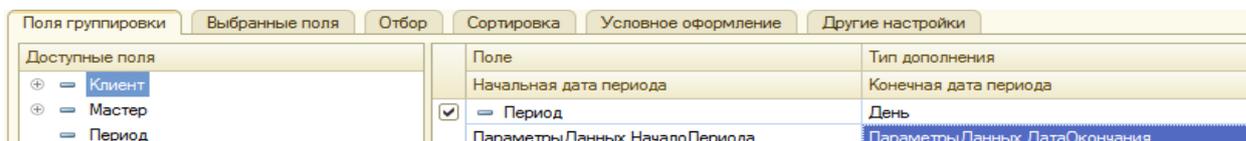


Рисунок 2.73 – Задание периодичности

Диаграмма

Создадим отображение отчета в виде диаграммы. Заданные параметры при создании диаграммы представлены на рисунках 2.74, 2.75 и 2.76.

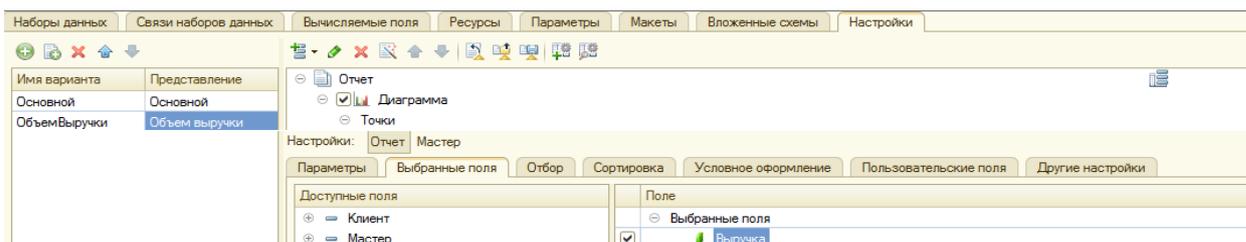


Рисунок 2.74 – Параметры диаграммы

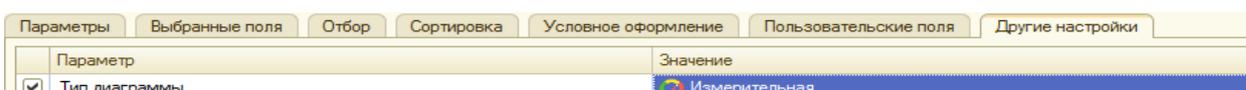


Рисунок 2.75 – Параметры диаграммы

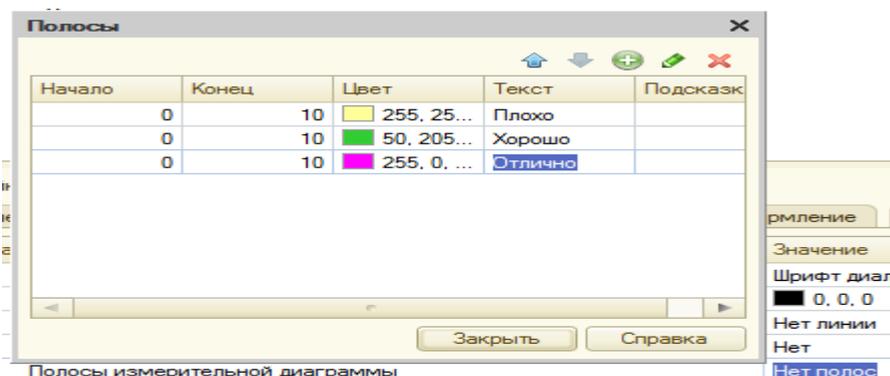


Рисунок 2.76 – Параметры диаграммы

Создадим отчет «Рейтинг материалов», который будет отражать список материалов, которые наиболее часто используются для выполнения работ указанном периоде. Параметры создания отчета представлены на рисунках 2.77-2.85.

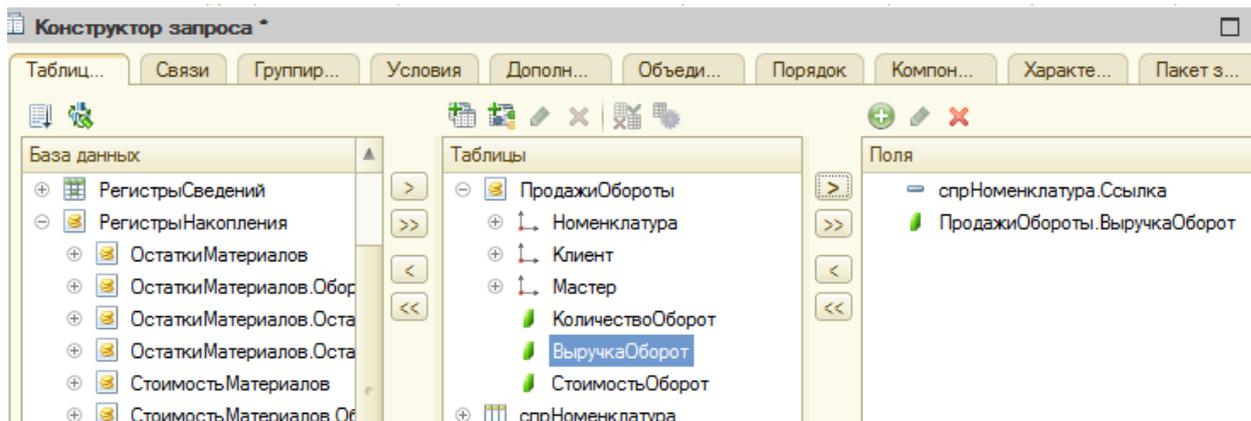


Рисунок 2.77 – Параметры отчета

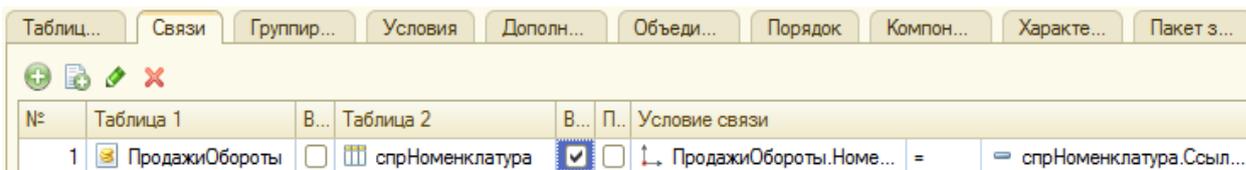


Рисунок 2.78 – Параметры отчета

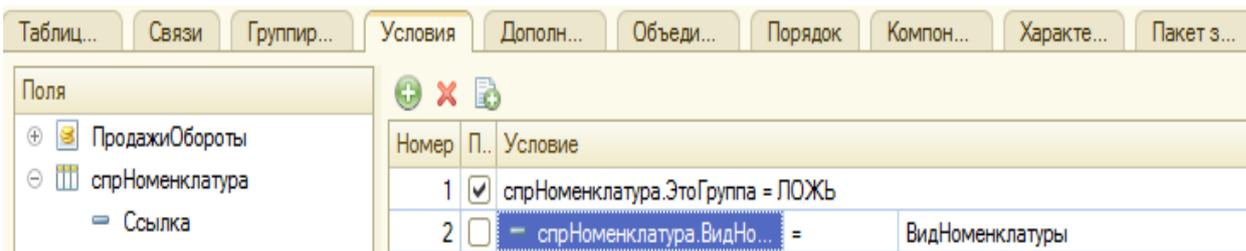


Рисунок 2.79 – Параметры отчета

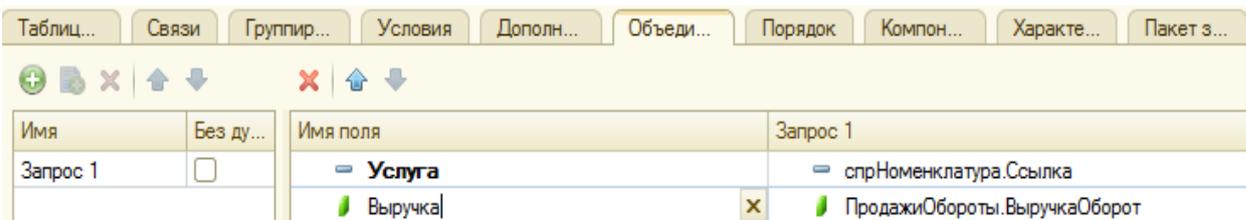


Рисунок 2.80 – Параметры отчета

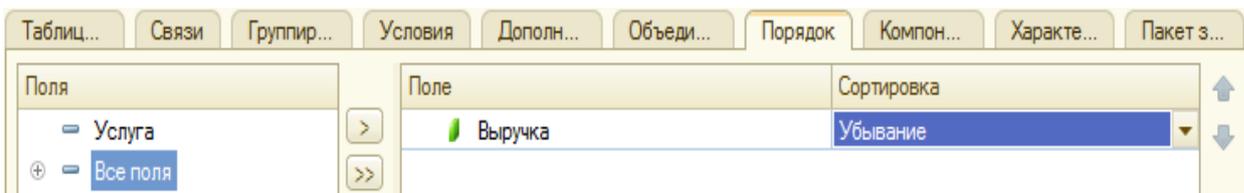


Рисунок 2.81 – Параметры отчета

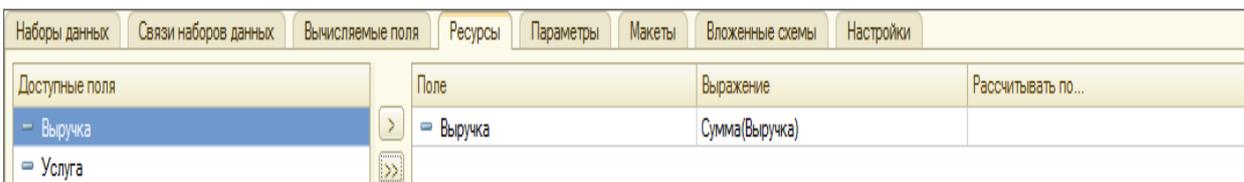


Рисунок 2.82 – Параметры отчета

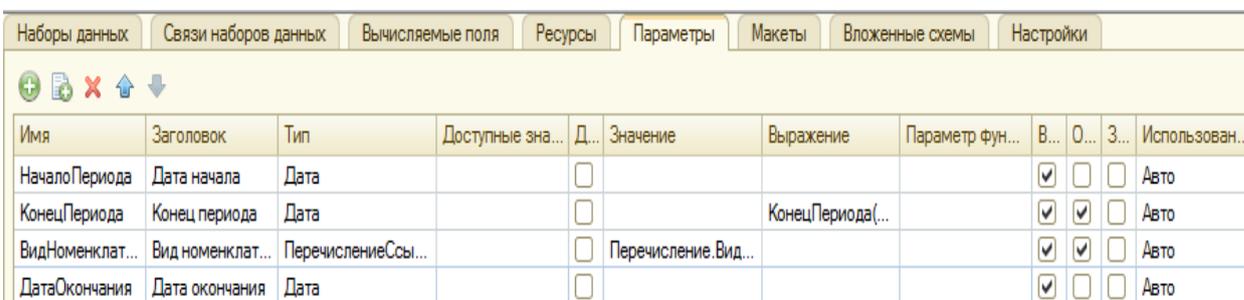


Рисунок 2.83 – Параметры отчета

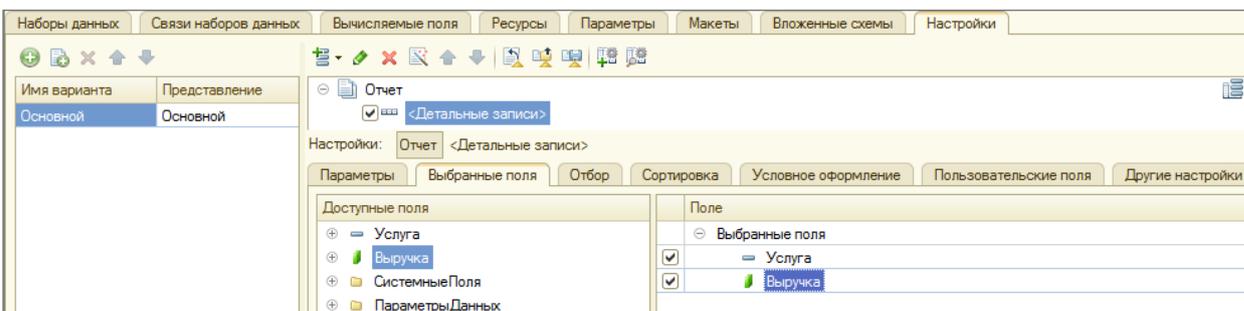


Рисунок 2.84 – Параметры отчета

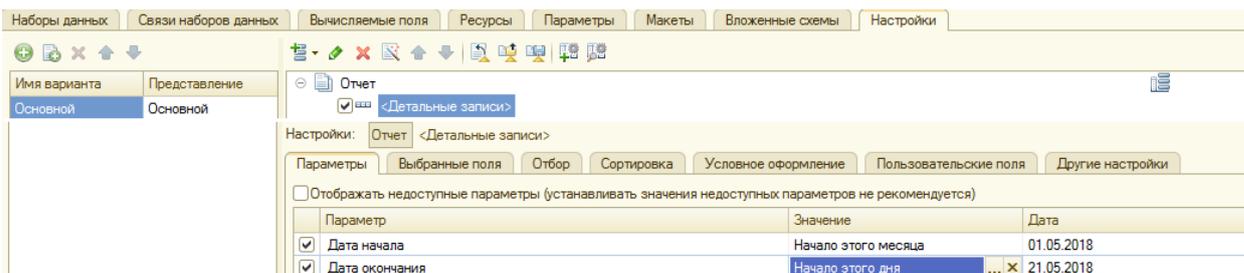


Рисунок 2.85 – Параметры отчета

Получение актуальных значений из периодического регистра сведений
Создадим отчет «Перечень услуг», содержащий информацию о том, какие услуги и по какой цене оказывает ООО «На все руки мастер».

Зададим параметры виртуальной таблицы. Задание параметров представлено на рисунке 2.86.

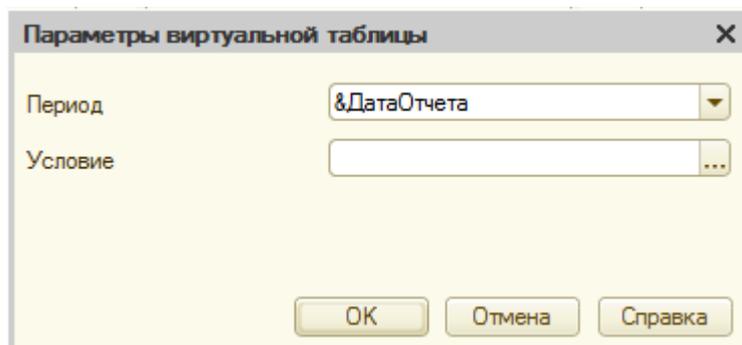


Рисунок 2.86 – Параметры виртуальной таблицы

Использование вычисляемого поля в отчете

Создадим отчет «Рейтинг клиентов».

Добавим вычисляемое поле. Параметры вычисляемого поля представлены на рисунке 2.87.

| Наборы данных | | Связи наборов данных | | Вычисляемые поля | | Ресурсы | | Параметры | | Максимум | |
|---------------|---------------------|----------------------|--------------------------|--------------------------|--------------------------|--------------------------|--|-----------|--|----------|--|
| + - X | | | | | | | | | | | |
| Путь к данным | Выражение | Заголовок | Ограничение доступности | | | | | | | | |
| | | | Поле | Усло... | Группа | Упор... | | | | | |
| Доход | Выручка - Стоимость | Доход | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | | |

Рисунок 87 – Параметры вычисляемого поля

Вывод данных в таблицу

По подобию прошлых отчетов создадим новый отчет «Универсальный».

2.11 Оптимизация проведения документа «Оказание услуги»

Повышение скорости проводимости

Перейдем в конструктор запроса с обработкой результата и укажем необходимые параметры. Параметры представлены на рисунках 2.88 и 2.89.

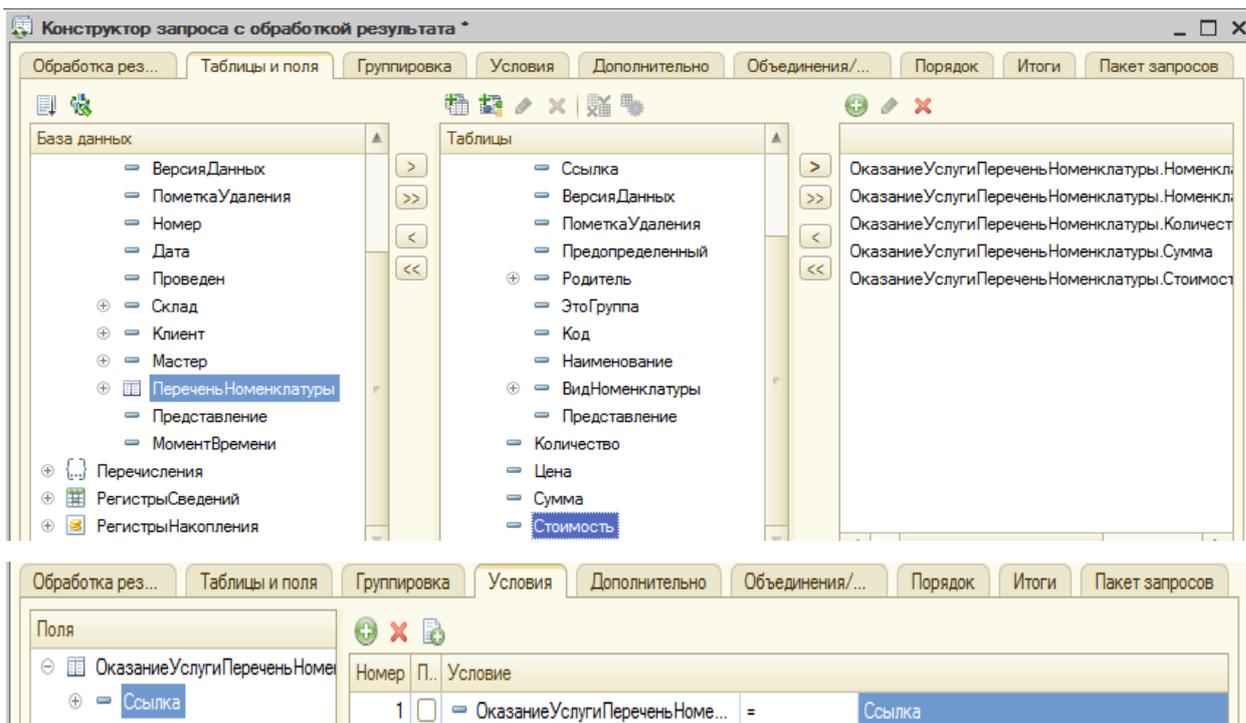


Рисунок 2.88 – Параметры запроса с обработкой результата

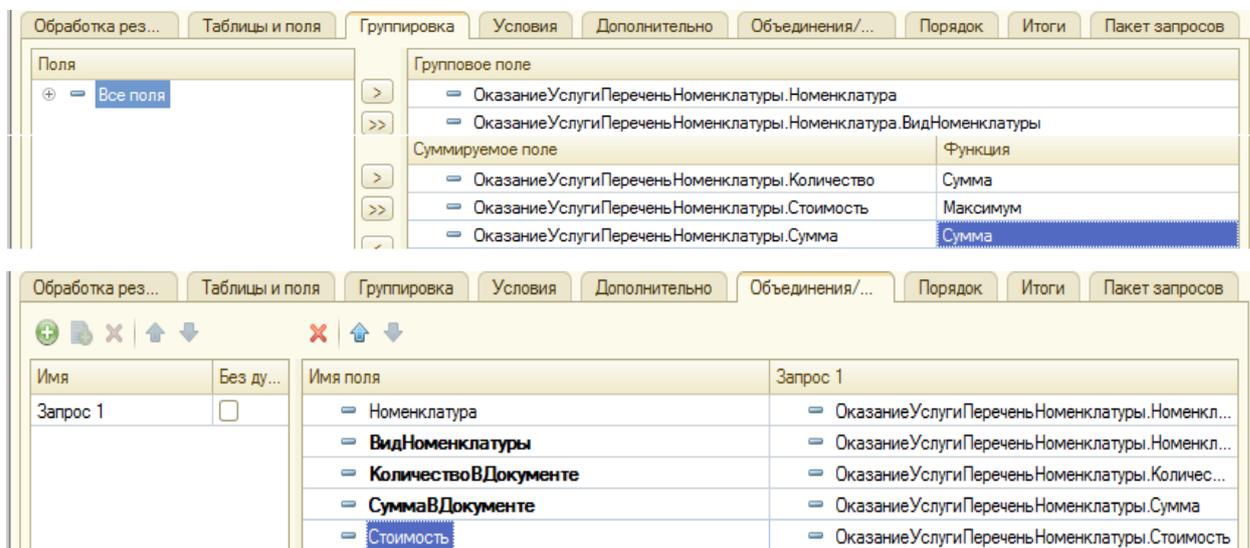


Рисунок 2.89 – Параметры запроса с обработкой результата

В результате процедура проведения примет вид, представленный на рисунке 2.90.

```

] Процедура ОбработкаПроведения(Отказ, Режим)
//{{{ КОНСТРУКТОР ДВИЖЕНИЙ РЕГИСТРОВ
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора, внесенные вручную изменения будут утеряны

// регистр ОстаткиМатериалов Расход
Движения.ОстаткиМатериалов.Записывать = Истина;
Движения.СтоимостьМатериалов.Записывать = Истина;
Движения.Продажи.Записывать = Истина;
//{{{ КОНСТРУКТОР ЗАПРОСА С ОБРАБОТКОЙ РЕЗУЛЬТАТА
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора, внесенные вручную изменения будут утеряны

Запрос = Новый Запрос;
Запрос.Текст =
    "ВЫБРАТЬ
    | ОказаниеУслугиПереченьНоменклатуры.Номенклатура,
    | ОказаниеУслугиПереченьНоменклатуры.Номенклатура.ВидНоменклатуры КАК ВидНоменклатуры,
    | СУММА (ОказаниеУслугиПереченьНоменклатуры.Количество) КАК КоличествоВДокументе,
    | СУММА (ОказаниеУслугиПереченьНоменклатуры.Сумма) КАК СуммаВДокументе,
    | МАКСИМУМ (ОказаниеУслугиПереченьНоменклатуры.Стоимость) КАК Стоимость
    | ИЗ
    | Документ.ОказаниеУслуги.ПереченьНоменклатуры КАК ОказаниеУслугиПереченьНоменклатуры
    | ГДЕ
    | ОказаниеУслугиПереченьНоменклатуры.Ссылка = &Ссылка
    | СГРУППИРОВАТЬ ПО
    | ОказаниеУслугиПереченьНоменклатуры.Номенклатура,
    | ОказаниеУслугиПереченьНоменклатуры.Номенклатура.ВидНоменклатуры";

Запрос.УстановитьПараметр ("Ссылка", Ссылка);

Результат = Запрос.Выполнить ();

ВыборкаДетальныеЗаписи = Результат.Выбрать ();

Пока ВыборкаДетальныеЗаписи.Следующий () Цикл
    Если ВыборкаДетальныеЗаписи.Номенклатура.ВидНоменклатуры = Перечисления.ВидНоменклатуры.Расход
    Тогда
        Движение = Движения.ОстаткиМатериалов.Добавить ();
        Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Материал = ВыборкаДетальныеЗаписи.Номенклатура;
        Движение.Склад = Склад;
        Движение.Количество = ВыборкаДетальныеЗаписи.КоличествоВДокументе;

        //регистр СтоимостьМатериалов Расход
        Движение = Движения.СтоимостьМатериалов.Добавить ();
        Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Материал = ВыборкаДетальныеЗаписи.Номенклатура;
        Движение.Стоимость = ВыборкаДетальныеЗаписи.КоличествоВДокументе * ВыборкаДетальныеЗаписи.Стоимость;
    Иначе
        Движение = Движения.Продажи.Добавить ();
        Движение.Период = Дата;
        Движение.Номенклатура = ВыборкаДетальныеЗаписи.Номенклатура;
        Движение.Клиент = Клиент;
        Движение.Мастер = Мастер;
        Движение.Количество = ВыборкаДетальныеЗаписи.КоличествоВДокументе;
        Движение.Выручка = ВыборкаДетальныеЗаписи.СуммаВДокументе;
        Движение.Стоимость = ВыборкаДетальныеЗаписи.Стоимость * ВыборкаДетальныеЗаписи.Количество;
    КонецЕсли;
КонецЦикла;
- Процедура

```

Рисунок 2.90 – Процедура проведения

Автоматический расчет стоимости

Сначала удалим реквизит табличной части «Стоимость» документа «ОказаниеУслуги» и соответствующее поле из таблицы «ПереченьНоменклатуры».

Создадим менеджер временных таблиц и укажем, что этот запрос будет использовать менеджер временных таблиц. Создание менеджера временных таблиц представлено на рисунке 2.91.

```

//Создать менеджер временных таблиц.
МенеджерВТ = Новый МенеджерВременныхТаблиц;

Запрос = Новый Запрос;
//Укажем, какой менеджер временных таблиц использует этот запрос
Запрос.МенеджерВременныхТаблиц = МенеджерВТ;

```

Рисунок 2.91 – Создание менеджера временных таблиц

В конструкторе запроса изменим запрос. Изменение представлено на рисунке 2.92.

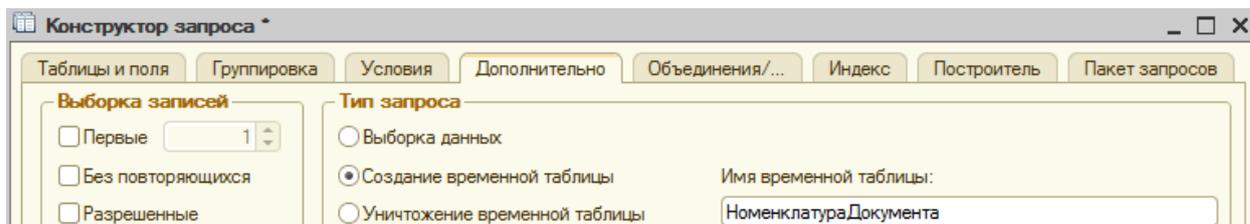


Рисунок 2.92 – Настройки запроса

Создадим новый запрос и сконструируем его в «Конструкторе запроса». Далее соединим виртуальную таблицу левым соединением с таблицей остатков. Параметры запроса представлены на рисунке 2.93.

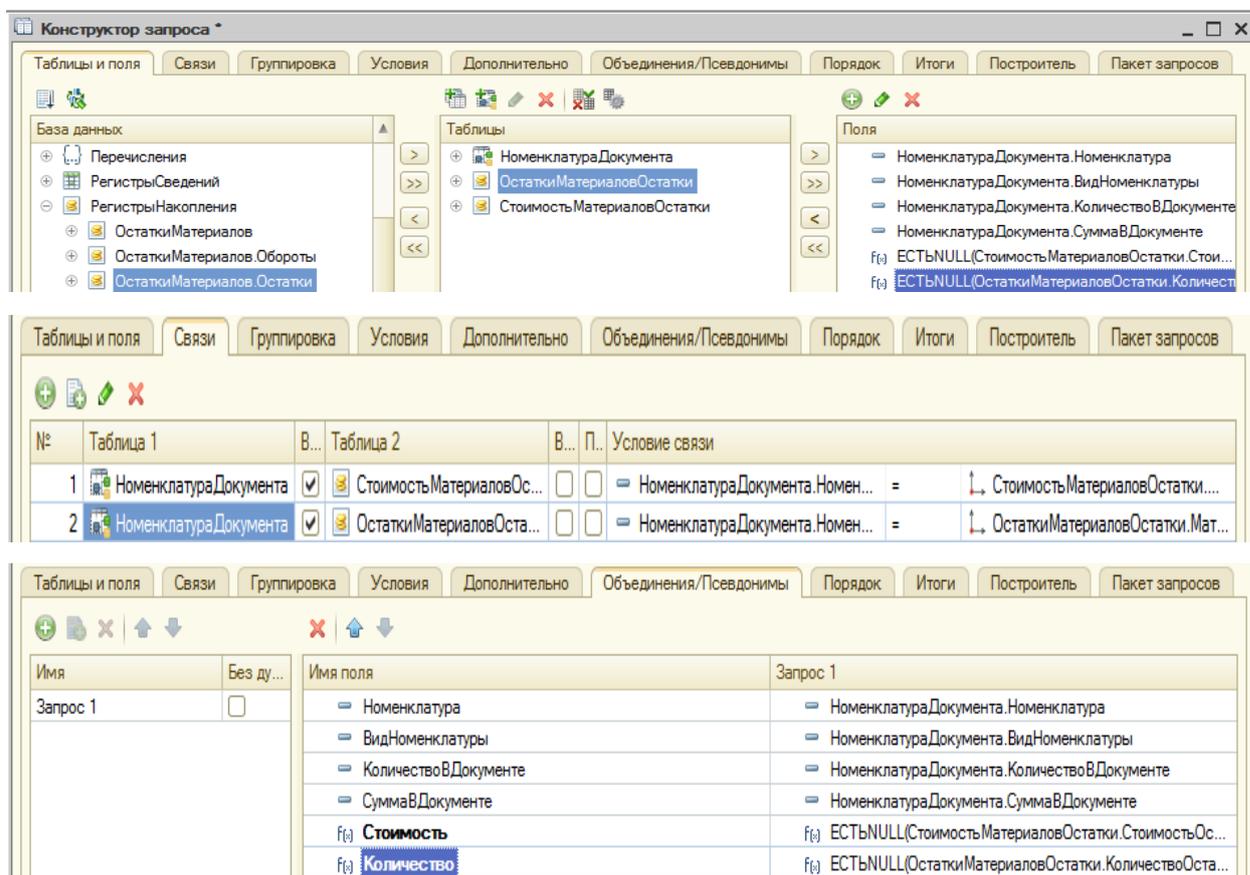


Рисунок 2.93 – Параметры запроса

Если у нас есть номенклатура, но она еще не поступала в нашу фирму, то в запросе лучше сразу избавиться от нее. Избавление представлено на рисунке 2.94.

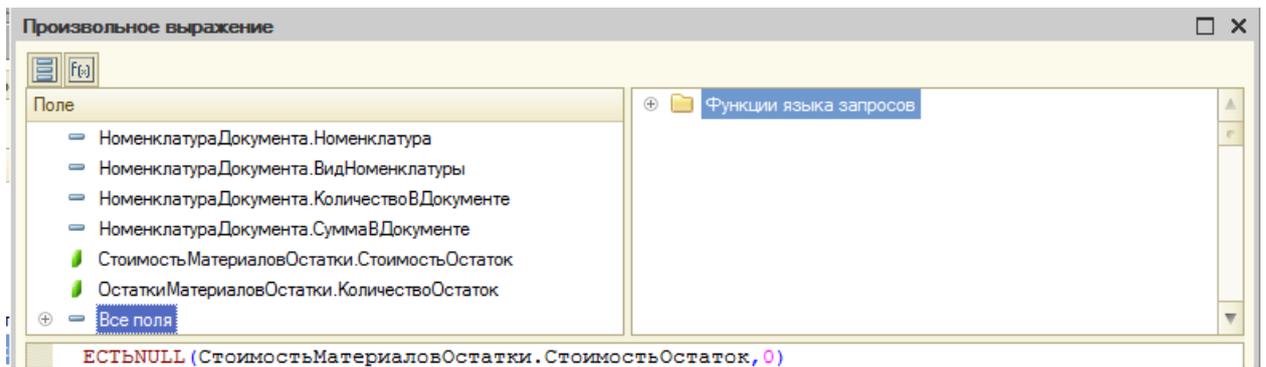


Рисунок 2.94 – Избавление от пустых значений

Полученный текст запроса представлен на рисунке 2.95.

```

ВЫБРАТЬ
    НоменклатураДокумента.Номенклатура,
    НоменклатураДокумента.ВидНоменклатуры,
    НоменклатураДокумента.КоличествоВДокументе,
    НоменклатураДокумента.СуммаВДокументе,
    ЕСТЬNULL (СтоимостьМатериаловОстатки.СтоимостьОстаток, 0) КАК Стоимость,
    ЕСТЬNULL (ОстаткиМатериаловОстатки.КоличествоОстаток, 0) КАК Количество
ИЗ
    НоменклатураДокумента КАК НоменклатураДокумента
        ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.СтоимостьМатериалов.Остатки (
            '
            Материал В
            (ВЫБРАТЬ
                НоменклатураДокумента.Номенклатура
            ИЗ
                НоменклатураДокумента) ) КАК СтоимостьМатериаловОстатки
        ПО НоменклатураДокумента.Номенклатура = СтоимостьМатериаловОстатки.Материал
        ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.ОстаткиМатериалов.Остатки (
            '
            Материал В
            (ВЫБРАТЬ
                НоменклатураДокумента.Номенклатура
            ИЗ
                НоменклатураДокумента) ) КАК ОстаткиМатериаловОстатки
        ПО НоменклатураДокумента.Номенклатура = ОстаткиМатериаловОстатки.Материал

```

Рисунок 2.95 – Текст запроса

Добавим изменения в текст процедуры. Изменения представлены на рисунках 2.96 и 2.97.

```

>>> Движения.СтоимостьМатериалов.Записать ();
>>> Движения.ОстаткиМатериалов.Записать ();

>>> | Результат = Запрос2.Выполнить ();

```

Рисунок 2.96 – Изменения в тексте процедуры

```

>>> Если ВыборкаДетальныеЗаписи.Количество = 0 Тогда
        СтоимостьМатериала = 0;
Иначе
        СтоимостьМатериала = ВыборкаДетальныеЗаписи.Стоимость/ВыборкаДетальныеЗаписи.Количество;
>>> КонецЕсли;

```

Рисунок 2.97 – Изменения в тексте процедуры

Контроль остатков

Создадим «Запрос» с параметрами, представленными на рисунке 2.98.

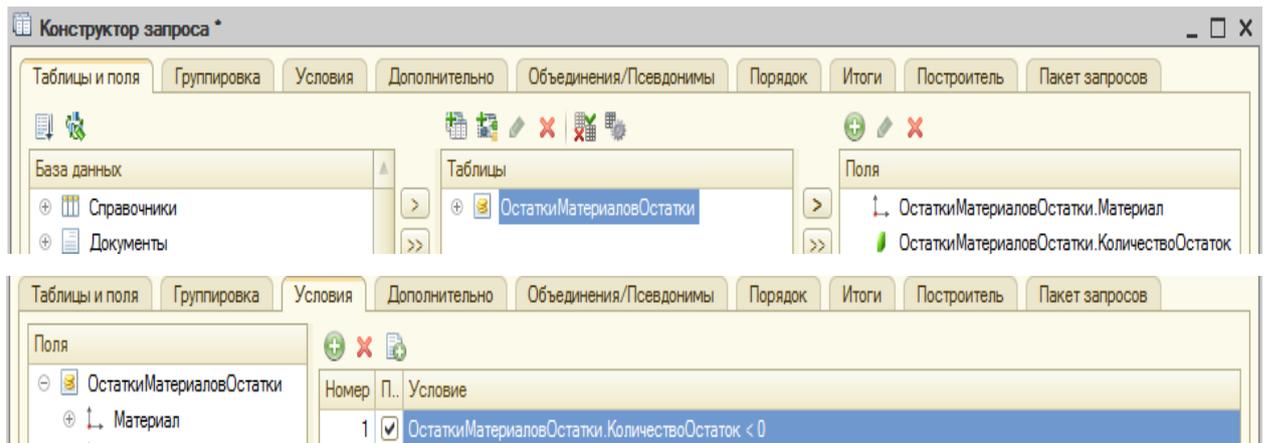


Рисунок 2.98 – Параметры «Запрос»

Добавим в текст процедуры строки для проверки отрицательных остатков. Добавленные строки представлены на рисунке 2.99.

```

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
    Сообщение = Новый СообщениеПользователю();
    Сообщение.Текст = "Не хватает" + Строка(ВыборкаДетальныеЗаписи.КоличествоОстаток) + "единиц материала" + ВыборкаДетальныеЗап
    Сообщение.Сообщить();
    Отказ = Истина;
КонецЦикла;

```

Рисунок 2.99 – Добавленные строки

Блокировка данных, которые читаются и изменяются при проведении
Для блокирования данных установим управляемую блокировку в значение «истина». Изменение значения представлено на рисунке 2.100.

```

//Установим необходимость блокировки данных в регистрах СтоимостьМатериалов и ОстаткиМатериалов
Движения.СтоимостьМатериалов.БлокироватьДляИзменения = Истина;
Движения.ОстаткиМатериалов.БлокироватьДляИзменения = Истина;

```

Рисунок 2.100 – Изменение значения управляемой блокировки

2.12 План видов характеристик

Вначале создадим справочник «ВариантыНоменклатуры» и в списке его владельцев укажем справочник «Номенклатура». Затем создадим справочник «ДополнительныеСвойстваНоменклатуры».

Создадим план видов характеристик «СвойстваНоменклатуры». Параметры плана видов характеристик представлены на рисунке 2.101.

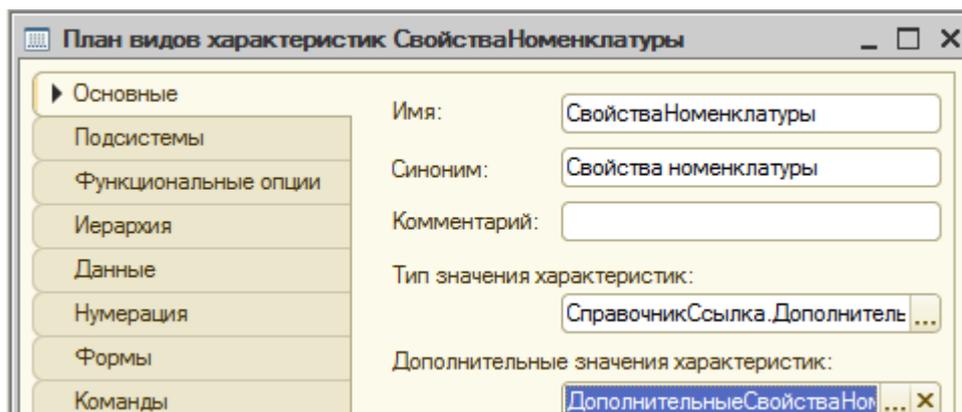


Рисунок 2.101 – Параметры плана видов характеристик

У справочника «ДополнительныеСвойстваНоменклатуры» укажем в качестве владельца план видов характеристик «СвойстваНоменклатуры».

Создадим регистр сведений «ЗначенияСвойствНоменклатуры». Заданные параметры представлены на рисунке 2.102.

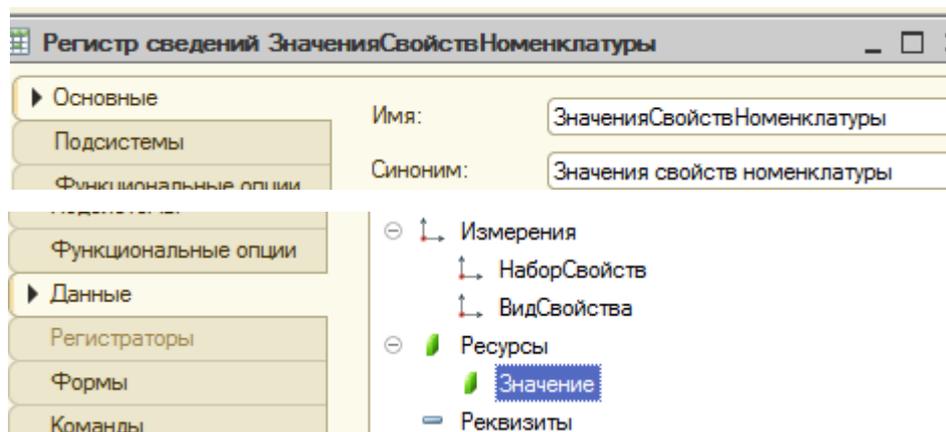


Рисунок 2.102 – Параметры регистра сведений

В справочнике «Номенклатура» заменим синоним стандартного реквизита «Родитель» на «Группа номенклатуры». Замена представлена на рисунке 2.103.

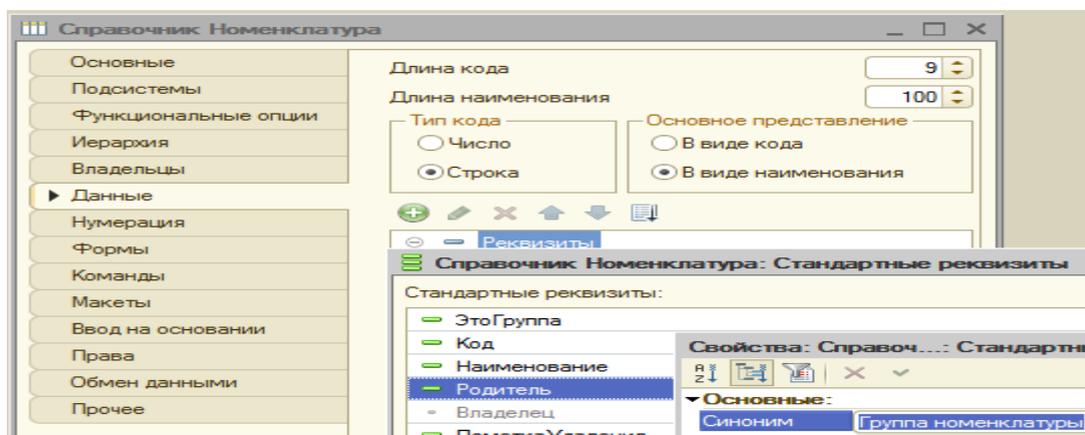


Рисунок 2.103 – Замена синонима реквизита

Справочник «Варианты номенклатуры»

В справочнике «Варианты номенклатуры» создадим основную форму списка и в событии «ПриСозданииНаСервере» пропишем код представленный на рисунке 2.104.

```

«НаСервере
Процедура ПриСозданииНаСервере (Отказ, СтандартнаяОбработка)
Если Параметры.Отбор.Свойство ("Владелец") Тогда
    Элементы.Код.Видимость = Ложь;
КонецЕсли;
КонецПроцедуры
    
```

Рисунок 2.104 – Код процедуры «ПриСозданииНаСервере»

Добавим основное свойство «Представление объекта» как «Вариант номенклатуры».

Создадим основную форму элемента с параметрами, представленными на рисунке 2.105.

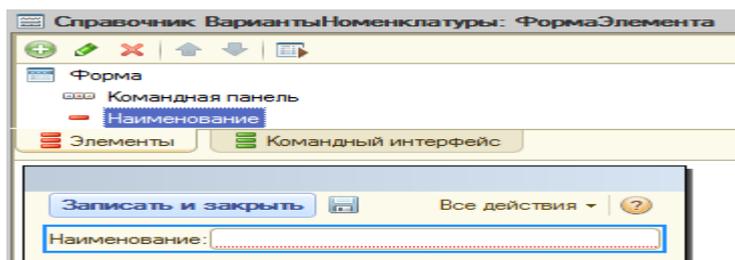


Рисунок 2.105 – Форма элемента

Заменяем синоним реквизита «Наименование» как «Название». Укажем в качестве представления списка регистра сведений «ЗначенияСвойстваНоменклатуры» «Состав варианта номенклатуры».

Регистр «Значения свойств номенклатуры»

В регистре сведений «ЗначенияСвойствНоменклатуры» измерению «ВидСвойства» укажем синоним «Свойство».

Для того, чтобы убрать поле «Набор свойств» создадим форму списка и в процедуре «ПриСозданииНаСервере» пропишем код представленный на рисунке 2.106.

```
    &НаСервере  
    Процедура ПриСозданииНаСервере (Отказ, СтандартнаяОбработка)  
    |    Если Параметры.Отбор.Свойство ("НаборСвойств") Тогда  
        Элементы.НаборСвойств.Видимость = Ложь;  
    КонецЕсли;  
    КонецПроцедуры
```

Рисунок 2.106 – Процедура «ПриСозданииНаСервере»

Далее в регистре сведений «ЗначенияСвойствНоменклатуры» зададим «Представление записи» как «Свойство и значение», а потом создадим основную форму записи и удалим поле «НаборСвойств».

Доработка учетных механизмов

Доработаем регистр «Остатки материалов». Создадим в нем новое измерение «НаборСвойств».

Доработаем документ «ПриходнаяНакладная». Добавим реквизит табличной части «НаборСвойств» и отобразим этот реквизит в форме документа. В конце добавим в тексте процедуры «ОбработкаПроведения» добавим строку, представленную на рисунке 2.107.

```
>>>| Движение.Материал = ТекСтрокаМатериалы.Материал;  
    Движение.НаборСвойств = ТекСтрокаМатериалы.НаборСвойств;  
    Движение.Склад = Склад;  
    Движение.Количество = ТекСтрокаМатериалы.Количество;
```

Рисунок 2.107 – Добавление строки в процедуру «ОбработкаПроведения»

Аналогичным образом доработаем документ «Оказание услуги».

Отчет, использующий характеристики

Добавим новый отчет «ОстаткиМатериаловПоСвойствам». Заданные параметры представлены на рисунках 2.108 и 2.109.

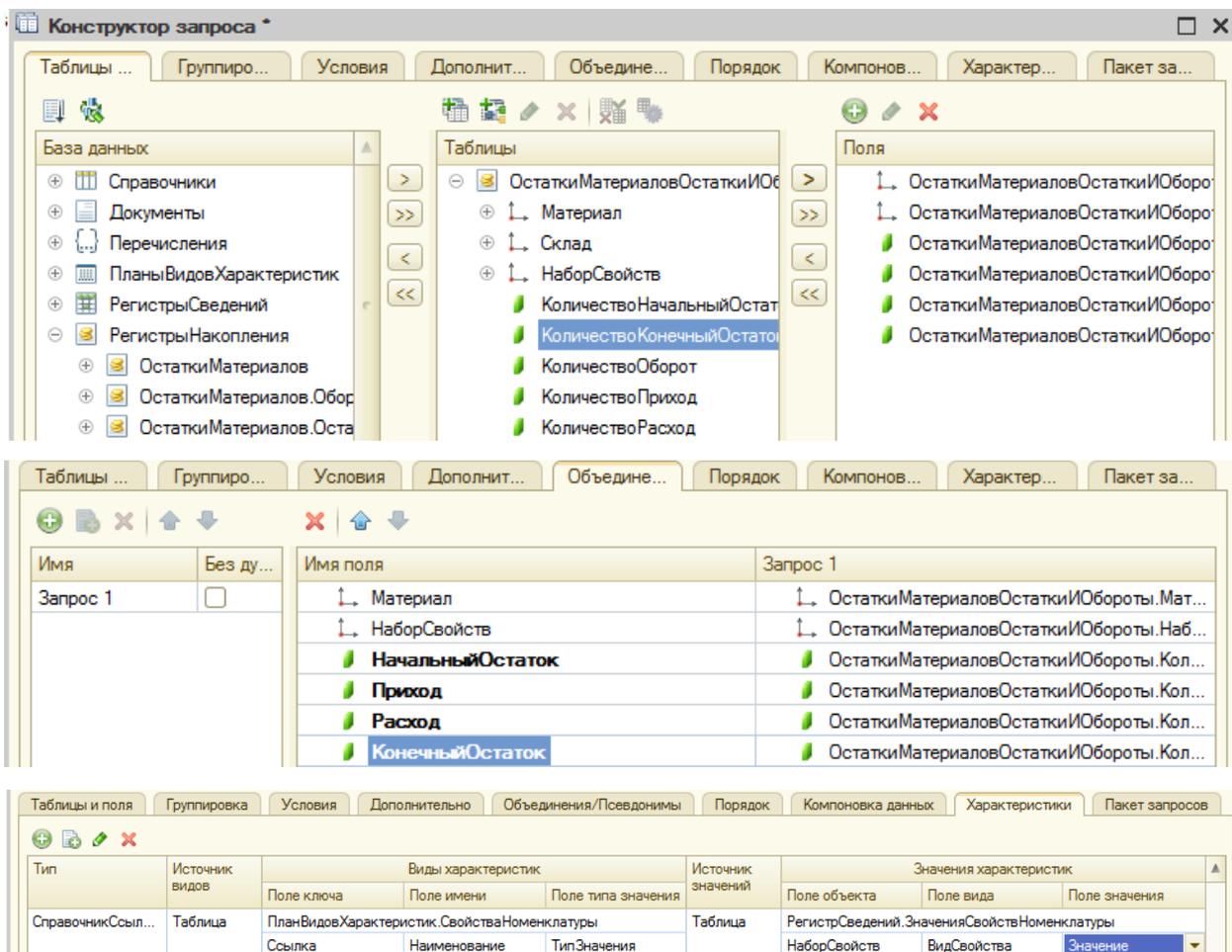


Рисунок 2.108 – Параметры отчета «ОстаткиМатериаловПоСвойствам»

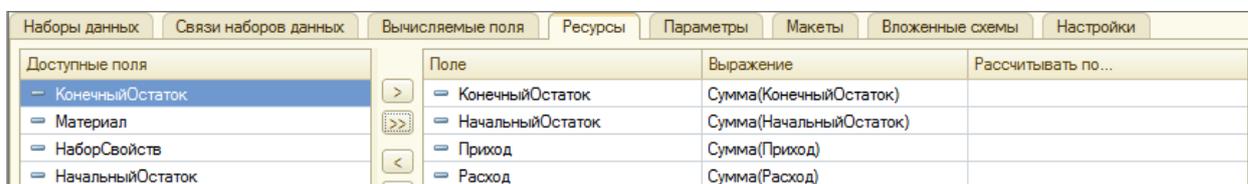


Рисунок 2.109 – Параметры отчета «ОстаткиМатериаловПоСвойствам»

Добавление «Плана видов характеристик»

Создадим план видов характеристик «ВидыСубконто».

Создадим справочник «Субконто» и в качестве владельца справочника укажем план видов характеристик «ВидыСубконто».

Параметры при создании плана видов характеристик «ВидыСубконто» представлены на рисунках 2.110 и 2.111.

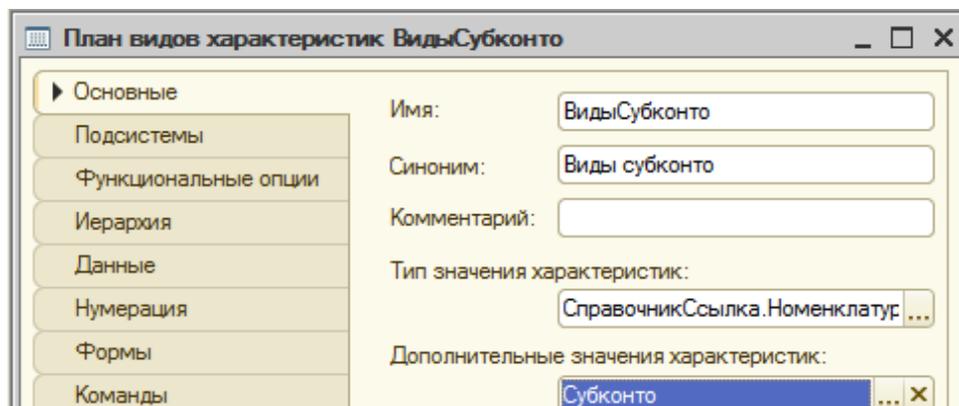


Рисунок 2.110 – параметры плана видов характеристик «ВидыСубконто»

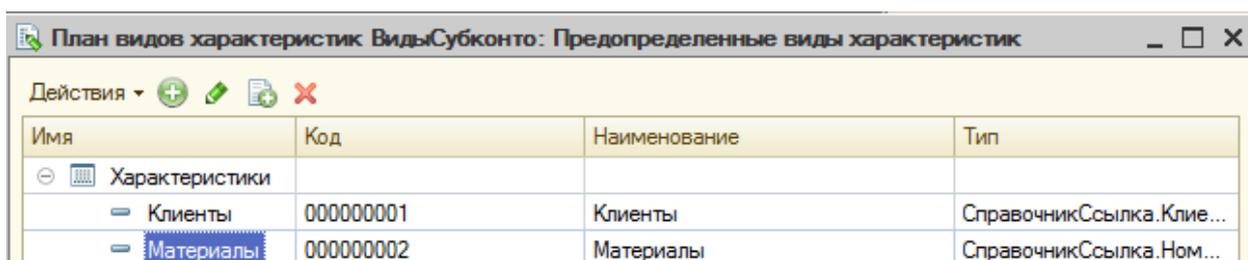


Рисунок 2.111 – параметры плана видов характеристик «ВидыСубконто»

Добавление «Плана счетов»

Создадим план счетов «Основной». Заданные параметры при создании представлены на рисунках 2.112 – 2.115.

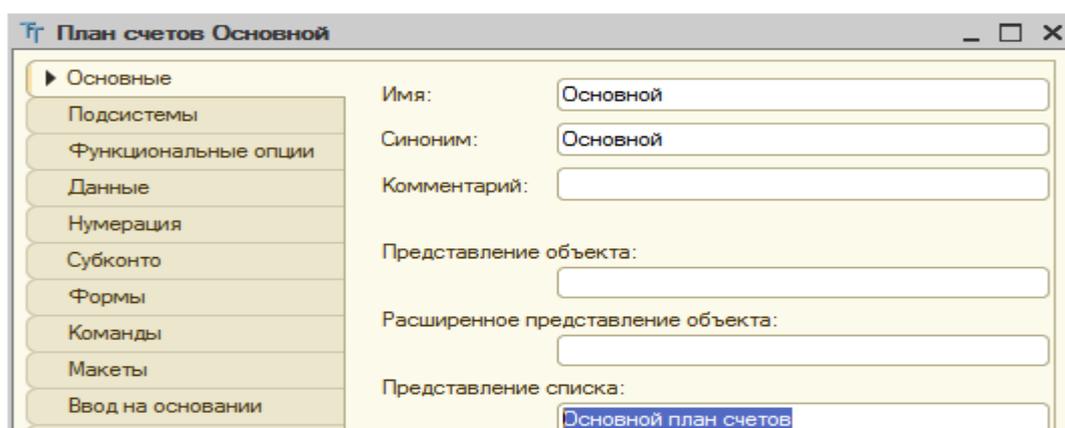


Рисунок 2.112 – Параметры плана счетов «Основной»

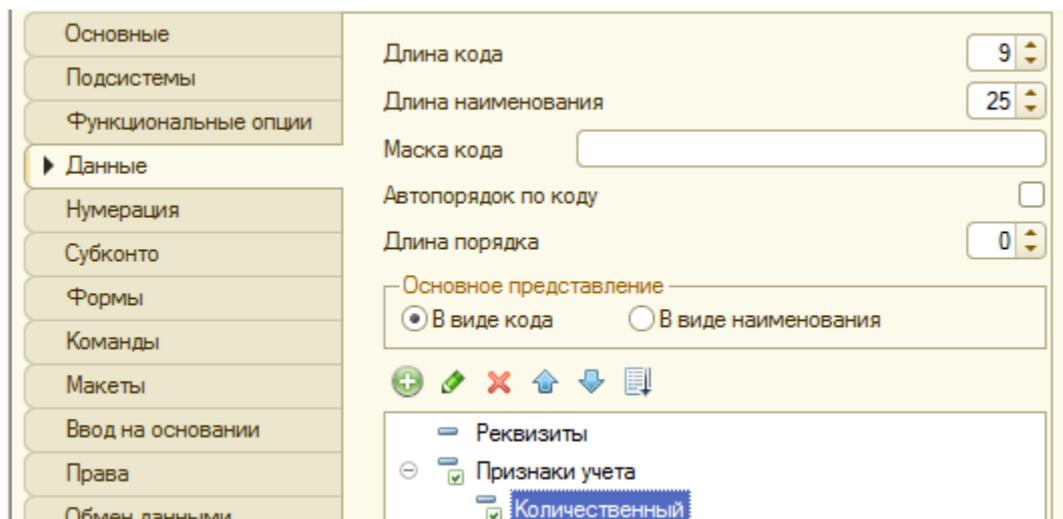


Рисунок 2.113 – Параметры плана счетов «Основной»

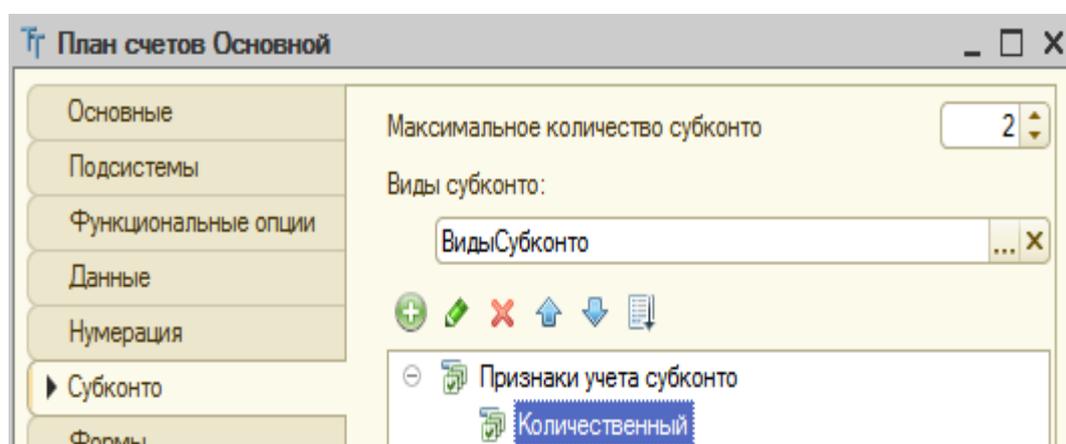


Рисунок 2.114 – Параметры плана счетов «Основной»

План счетов Основной: Предопределенные счета

| Имя | Код | Наименование | Вид | Забалансо... | Порядок | Количеств... | Субконто 1 | Субконто |
|-----------|-----|---------------------------|--------------|--------------|---------|--------------|------------|----------|
| Счета | | | | | | | | |
| Товары | 41 | Товары | Активный | | | ✓ | Материалы | |
| Расчет... | 60 | Расчеты с поставщиками | Активный/... | | | | | |
| Дебит... | 62 | Дебиторская задолженность | Активный/... | | | | Клиенты | |
| Капитал | 90 | Капитал | Активный/... | | | | | |

Рисунок 2.115 – Параметры плана счетов «Основной»

Добавление регистра бухгалтерии

Создадим регистр бухгалтерии «Управленческий». Заданные параметры представлены на рисунках 2.116 и 2.117.

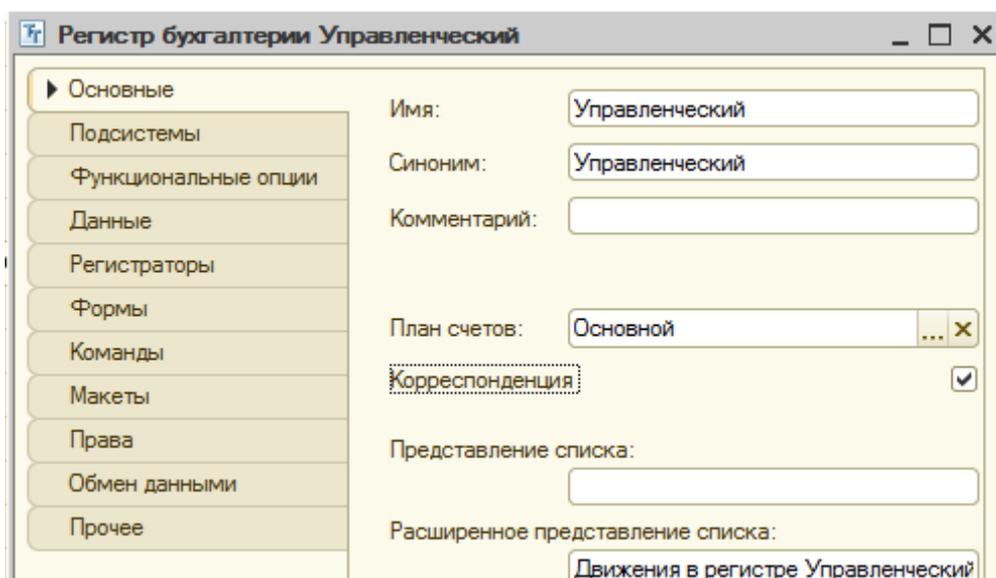


Рисунок 2.116 – Параметры регистра бухгалтерии «Управленческий»

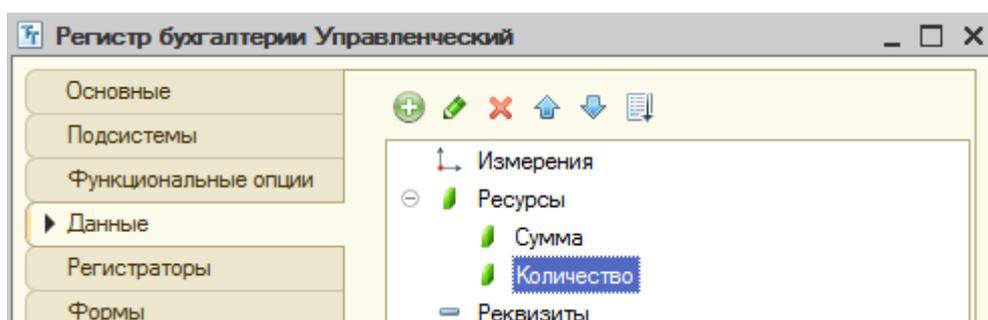


Рисунок 2.117 – Параметры регистра бухгалтерии «Управленческий»

Изменим процедуру проведения документа «Приходная накладная». После изменения процедура выглядит как представлено на рисунке 2.118.

```

Документ ПриходнаяНакладная: Модуль объекта
// регистр ОстаткиМатериалов Приход
Движения.ОстаткиМатериалов.Записывать = Истина;
Движения.СтоимостьМатериалов.Записывать = Истина;
Движения.Управленческий.Записывать = Истина;
Для Каждого ТекСтрокаМатериалы Из Материалы Цикл

    //Регистр Управленческий
    Движение = Движения.Управленческий.Добавить ();
    Движение.СчетДт = ПланыСчетов.Основной.Товары;
    Движение.СчетКт = ПланыСчетов.Основной.РасчетыСпоставщиками;
    Движение.Период = Дата;
    Движение.Сумма = ТекСтрокаМатериалы.Сумма;
    Движение.КоличествоДт = ТекСтрокаМатериалы.Количество;
    Движение.СубконтоДт [ПланыВидовХарактеристик.ВидыСубконто.Материалы] = ТекСтрокаМатериалы.Материал;
КонецЦикла;

```

Рисунок 2.118 – Процедура проведения

Доработаем документ «Оказание услуги». В процедуру проведения документа добавим код представленный на рисунке 2.119.

```
//Регистр Управленческий
//Первая проводка: Д 62 (ДебиторскаяЗадолженность) - К 90 (Капитал) Розничная сумма
Движение = Движения.Управленческий.Добавить ();
Движение.СчетДт = ПланыСчетов.Основной.ДебиторскаяЗадолженность;
Движение.СчетКт = ПланыСчетов.Основной.Капитал;
Движение.Период = Дата;
Движение.Сумма = ВыборкаДетальныеЗаписи.СуммаВДокументе;
Движение.СубконтоДт[ПланыВидовХарактеристик.ВидыСубконто.Клиенты] = Клиент;
//Вторая проводка Д 90 (Капитал) - К 41 (Товары) - себестоимость
Движение = Движения.Управленческий.Добавить ();
Движение.СчетДт = ПланыСчетов.Основной.Капитал;
Движение.СчетКт = ПланыСчетов.Основной.Товары;
Движение.Период = Дата;
Движение.Сумма = СтоимостьМатериала * ВыборкаДетальныеЗаписи.КоличествоВДокументе;
Движение.КоличествоКт = ВыборкаДетальныеЗаписи.КоличествоВДокументе;
Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Материалы] = ВыборкаДетальныеЗаписи.Номенклатура;
```

Рисунок 2.119 – Процедура проведения документа

Создадим оборотно-сальдовую ведомость. Построим запрос. Параметры запроса представлены на рисунке 2.120.

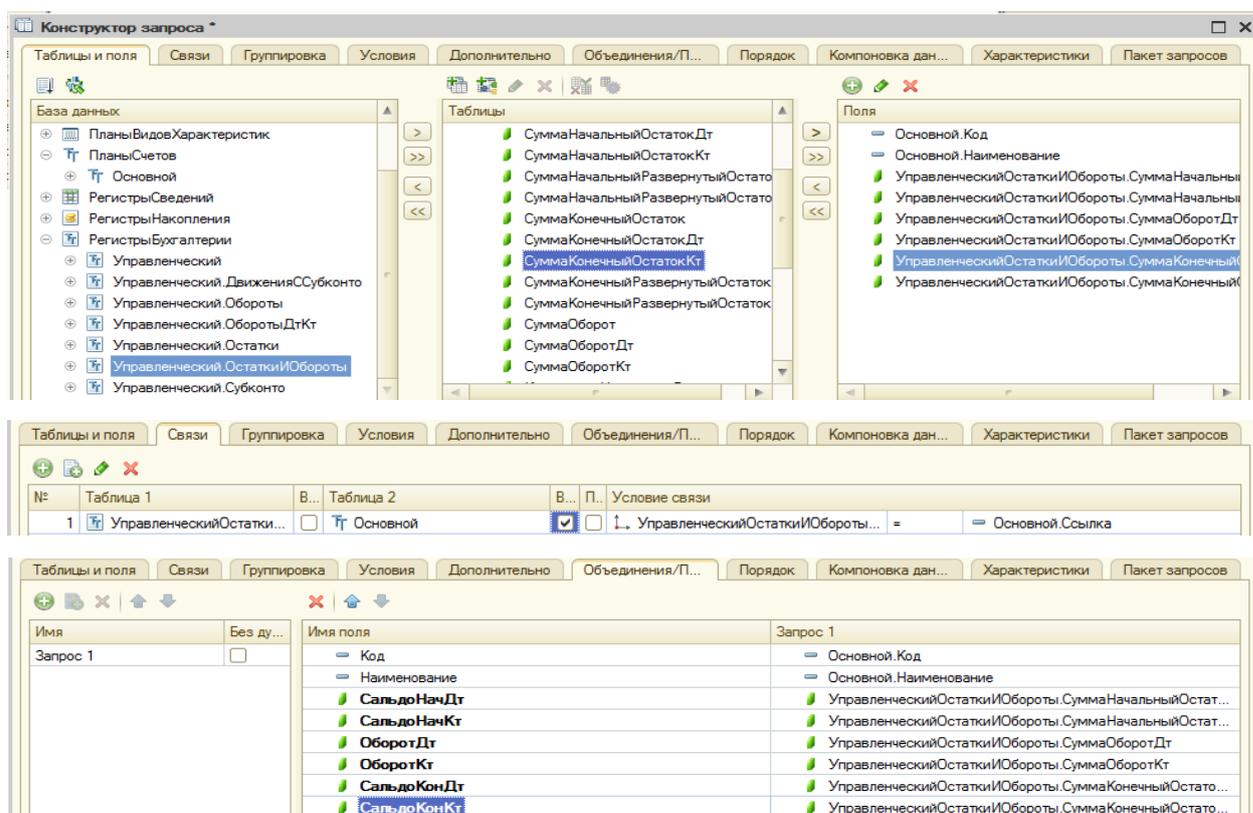


Рисунок 2.120 – Параметры оборотно-сальдовой ведомости

2.13 План видов расчета, регистр расчета

Добавим план видов расчета. Заданные при создании параметры представлены на рисунках 2.121 и 2.122.

Панель параметров «План видов расчета Основные Начисления»:

- Имя: Основные Начисления
- Синоним: Основные начисления
- Комментарий: (пустое поле)
- Представление объекта: (пустое поле)
- Расширенное представление объекта: (пустое поле)
- Представление списка: Виды расчетов
- Использует период действия
- Зависимость от базы:
 - Не зависит
 - Зависит по периоду действия
 - Зависит по периоду регистрации
- Базовые планы видов расчета:
 - Основные Начисления

Рисунок 2.121 – Параметры плана видов расчета

Таблица «Предопределенные виды расчета»:

| Имя | Код | Наименование |
|---------|---------|--------------|
| Невьход | Невьход | Невьход |
| Оклад | Оклад | Оклад |
| Премия | Премия | Премия |

Рисунок 2.122 – Параметры плана видов расчета

Создадим новый справочник «ВидыГрафиковРаботы». Параметры справочника представлены на рисунке 2.123.

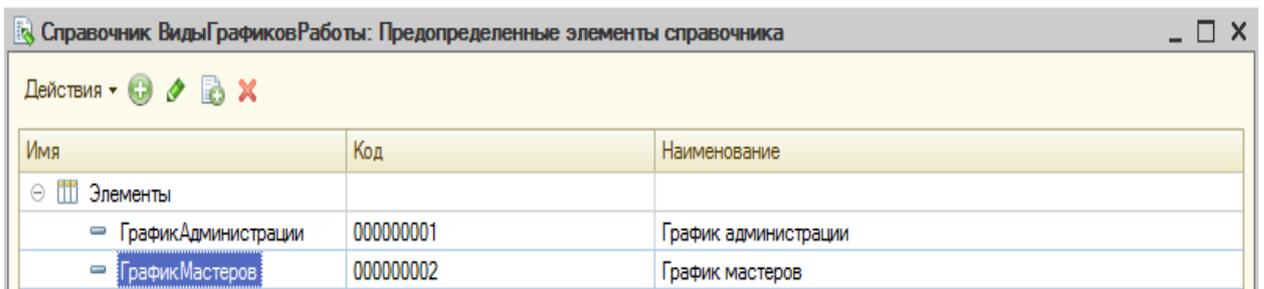


Рисунок 2.123 – Параметры справочника «ВидыГрафиковРаботы»

Создадим регистр сведений «ГрафикиРаботы». Параметры регистра представлены на рисунке 2.124.

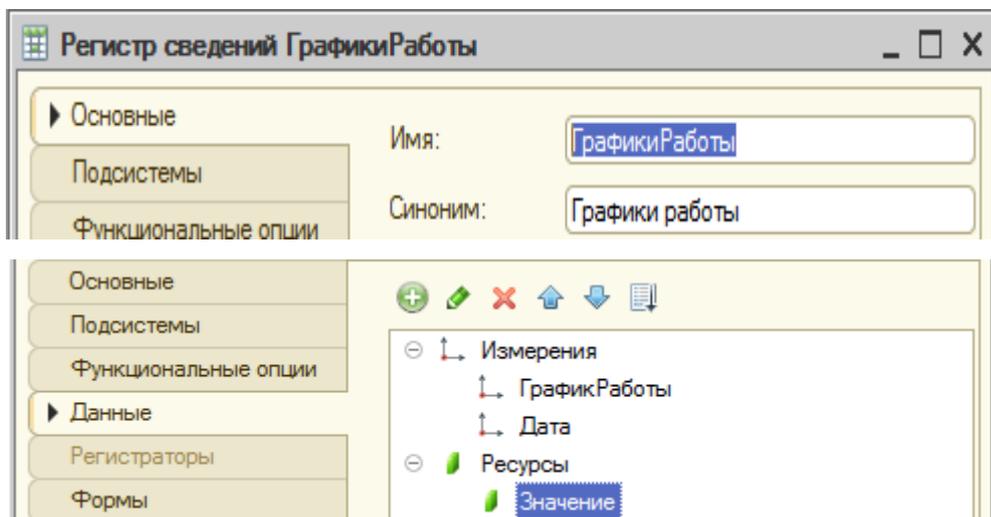


Рисунок 2.124 – Параметры регистра «ГрафикиРаботы»

Создадим регистр расчета «Начисления». Параметры представлены на рисунке 2.125 и 2.126.

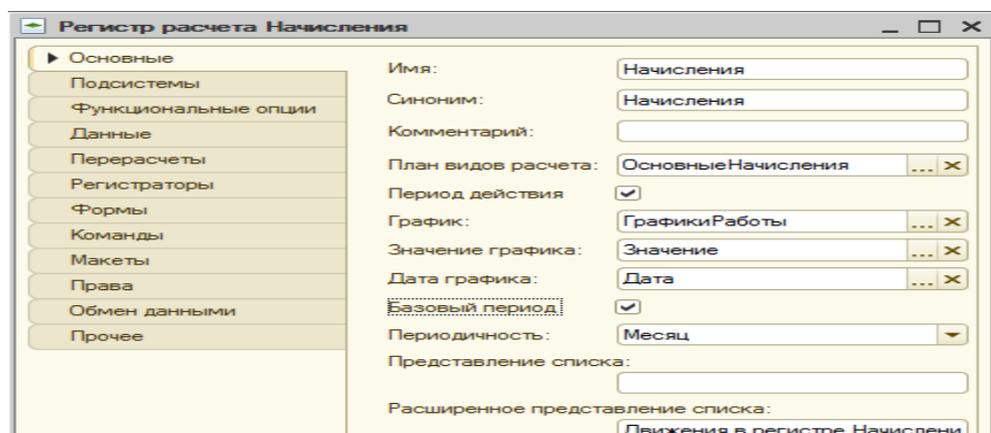


Рисунок 2.125 – Параметры регистра расчета «Начисления»

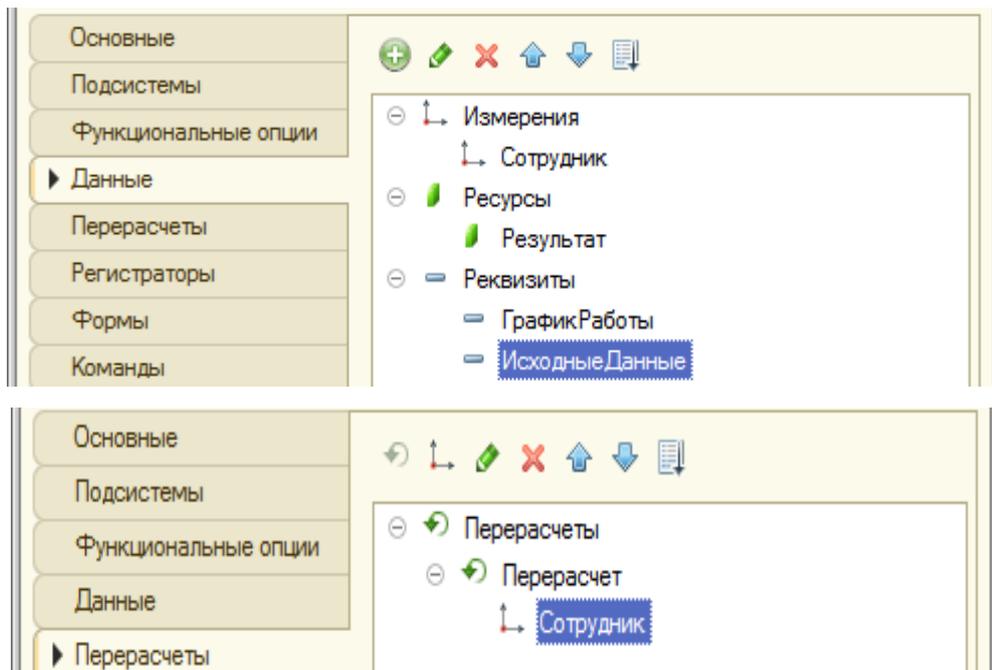


Рисунок 2.126 – Параметры регистра расчета «Начисления»

Создадим документ «НачисленияСотрудникам». Сконструируем движение. Движение представлено на рисунке 2.127.

```

Процедура ОбработкаПроведения(Отказ, Режим)
  {{{_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
  // Данный фрагмент построен конструктором.
  // При повторном использовании конструктора, внесенные вручную изменения будут ;

  // регистр Начисления
  Движения.Начисления.Записывать = Истина;
  Для Каждого ТекСтрокаНачисления Из Начисления Цикл
    Движение = Движения.Начисления.Добавить ();
    Движение.Сторно = Ложь;
    Движение.ВидРасчета = ТекСтрокаНачисления.ВидРасчета;
    Движение.ПериодДействияНачало = ТекСтрокаНачисления.ДатаНачала;
    Движение.ПериодДействияКонец = КонецДня(ТекСтрокаНачисления.ДатаОкончания);
    Движение.ПериодРегистрации = Дата;
    Движение.БазовыйПериодНачало = ТекСтрокаНачисления.ДатаНачала;
    Движение.БазовыйПериодКонец = КонецДня(ТекСтрокаНачисления.ДатаОкончания);
    Движение.Сотрудник = ТекСтрокаНачисления.Сотрудник;
    Движение.ГрафикРаботы = ТекСтрокаНачисления.ГрафикРаботы;
    Движение.ИсходныеДанные = ТекСтрокаНачисления.Начислено;
  КонецЦикла;

  }}}_КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
КонецПроцедуры

```

Рисунок 2.127 – Процедура проведения документа

2.14 Иллюстрация механизмов вытеснения и зависимости от базы

Отчет по перерасчетам

Создадим отчет по перерасчетам. Параметры отчета представлены на рисунке 2.128.

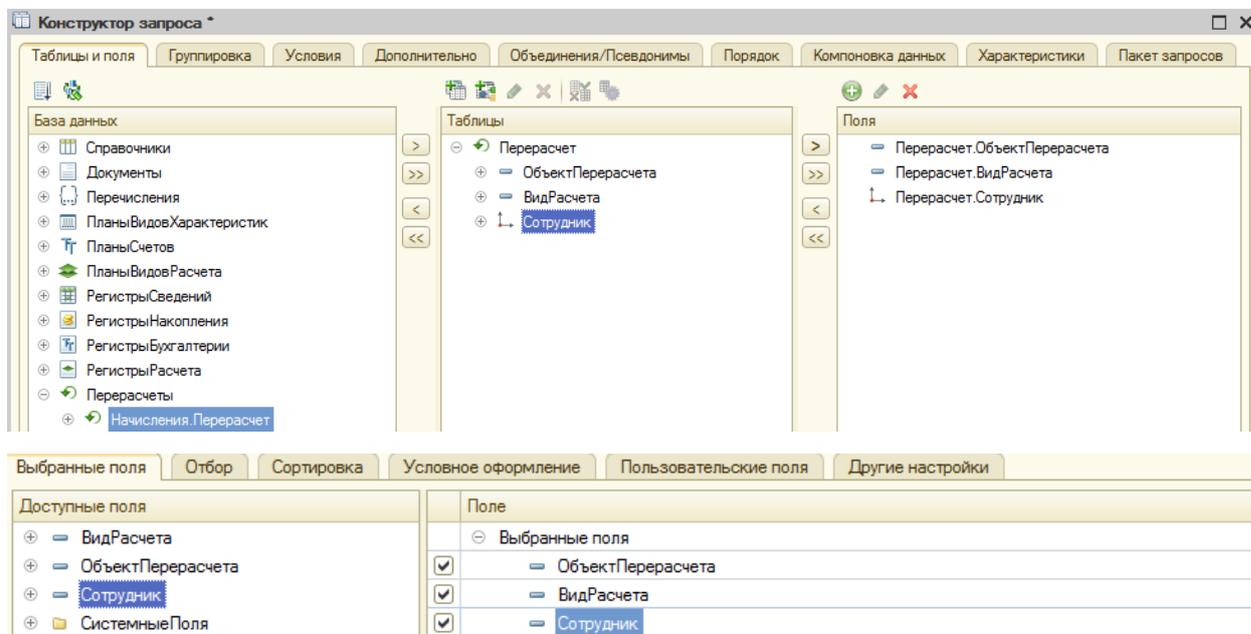


Рисунок 2.128 – Параметры отчета

Процедура расчета записей регистра расчета

Допишем процедуру проведения в документе «Начисления сотрудникам». Процедура представлена на рисунке 2.129.

```
//Записываем движение регистров.  
Движения.Начисления.Записать();  
  
//Получим список всех сотрудников, содержащихся в документе.  
Запрос = Новый Запрос(  
"ВЫБРАТЬ РАЗЛИЧНЫЕ  
| НачисленияСотрудникамНачисления.Сотрудник  
|ИЗ  
| Документ.НачисленияСотрудникам.Начисления  
|КАК НачисленияСотрудникамНачисления  
|  
|ГДЕ  
| НачисленияСотрудникамНачисления.Ссылка = &ТекущийДокумент");  
  
Запрос.УстановитьПараметр("ТекущийДокумент", Ссылка);  
  
//Сформируем список сотрудников  
ТаблЗнач = Запрос.Выполнить().Выгрузить();  
МассивСотрудников = ТаблЗнач.ВыгрузитьКолонку("Сотрудник");  
  
//Вызов процедуры РассчитатьНачисления из общего модуля.  
ПроведениеРасчетов.РассчитатьНачисления(Движения.Начисления, ПланыВидовРасчета.ОсновныеНачисления.Оклад, МассивСотрудников);  
Движения.Начисления.Записать(, Истина);
```

Рисунок 2.129 – Процедура проведения документа

Создадим новый общий модуль «ПроведениеРасчетов». Параметры модуля представлены на рисунке 2.130.

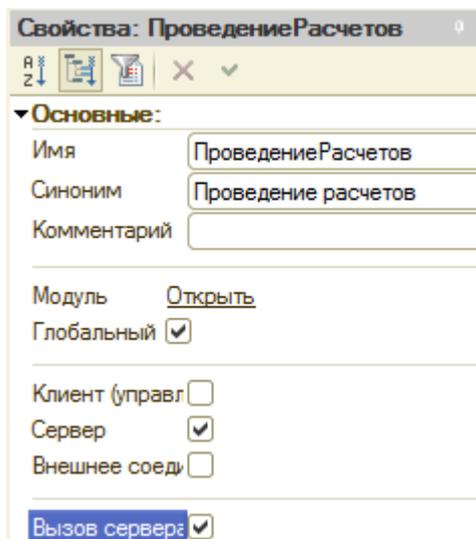


Рисунок 2.130 – Модуль «ПроведениеРасчетов»

В модуле напишем процедуру «РассчитатьНачисления», а также изменим процедуру «ОбработкаПроведения» в документе «НачисленияСотрудникам».

Отчет о начислениях сотрудникам

Создадим новый отчет «НачисленияСотрудникам». Параметры отчета представлены на рисунке 2.131.

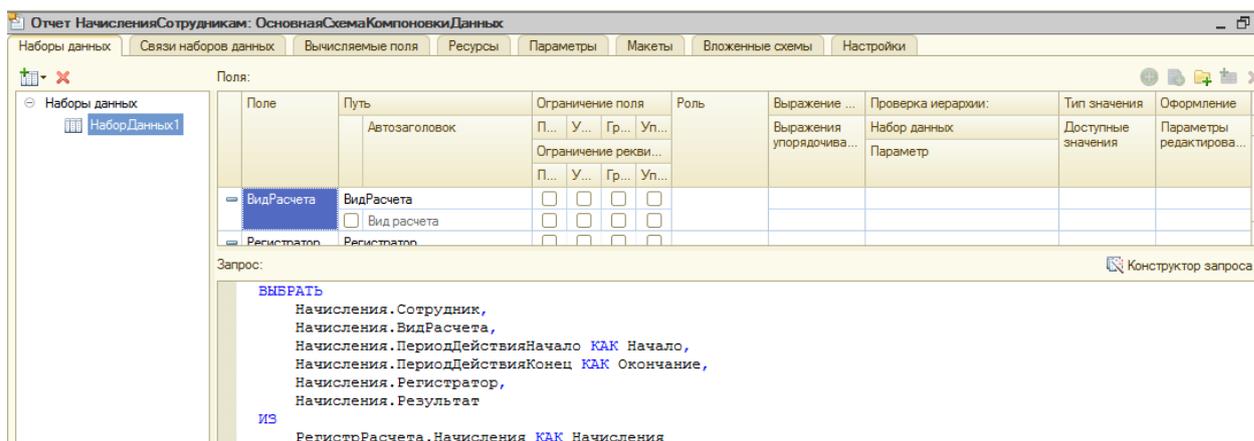


Рисунок 2.131 – Параметры отчета «НачисленияСотрудникам»

Диаграмма Ганта

Создадим новый отчет «**Диаграмма Ганта**». Параметры отчета представлены на рисунке 2.132.

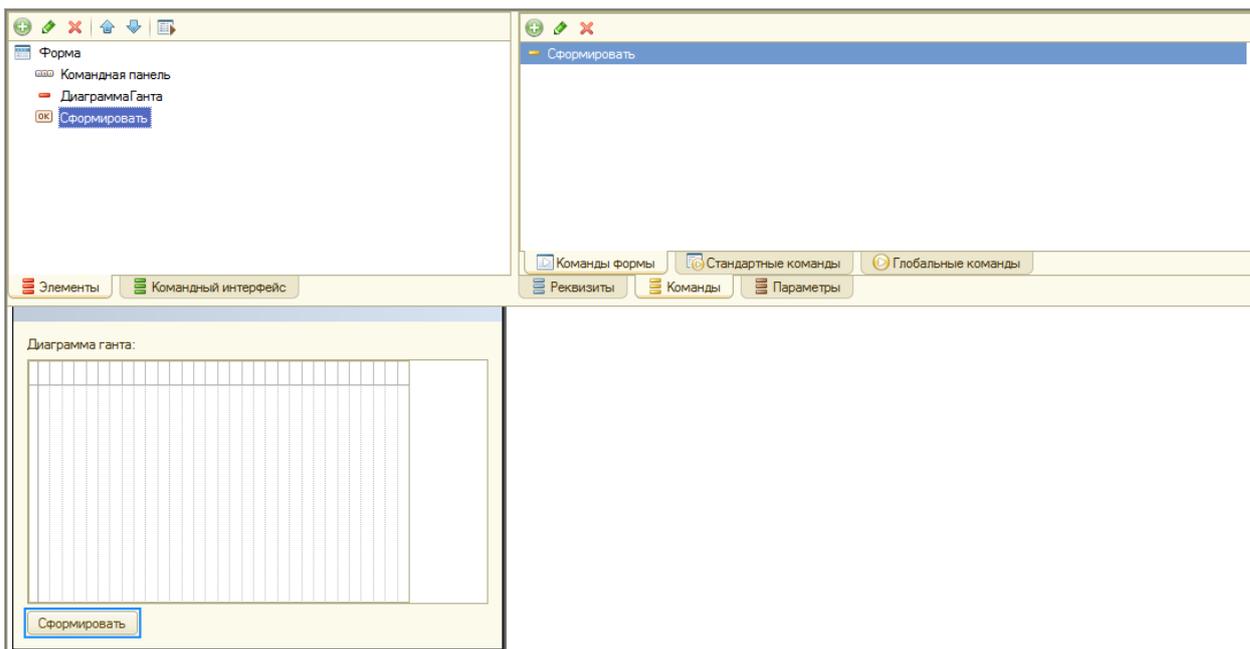


Рисунок 2.132 – Параметры отчета

Редактирование движений в форме документа

Создадим документ «**Ввод Начальных Остатков Номенклатуры**».

Параметры документа представлены на рисунке 2.133.

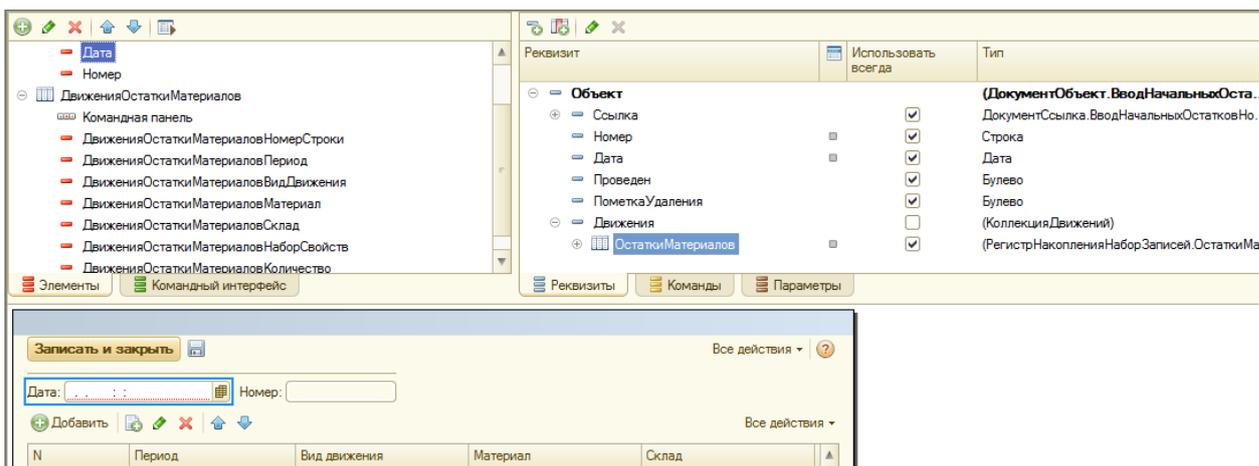


Рисунок 2.133 – Параметры документа

Рабочий стол и настройка командного интерфейса

Командный интерфейс разделов

Изменим командный интерфейс подсистемы «УчетМатериалов». Изменения представлены на рисунке 2.134.

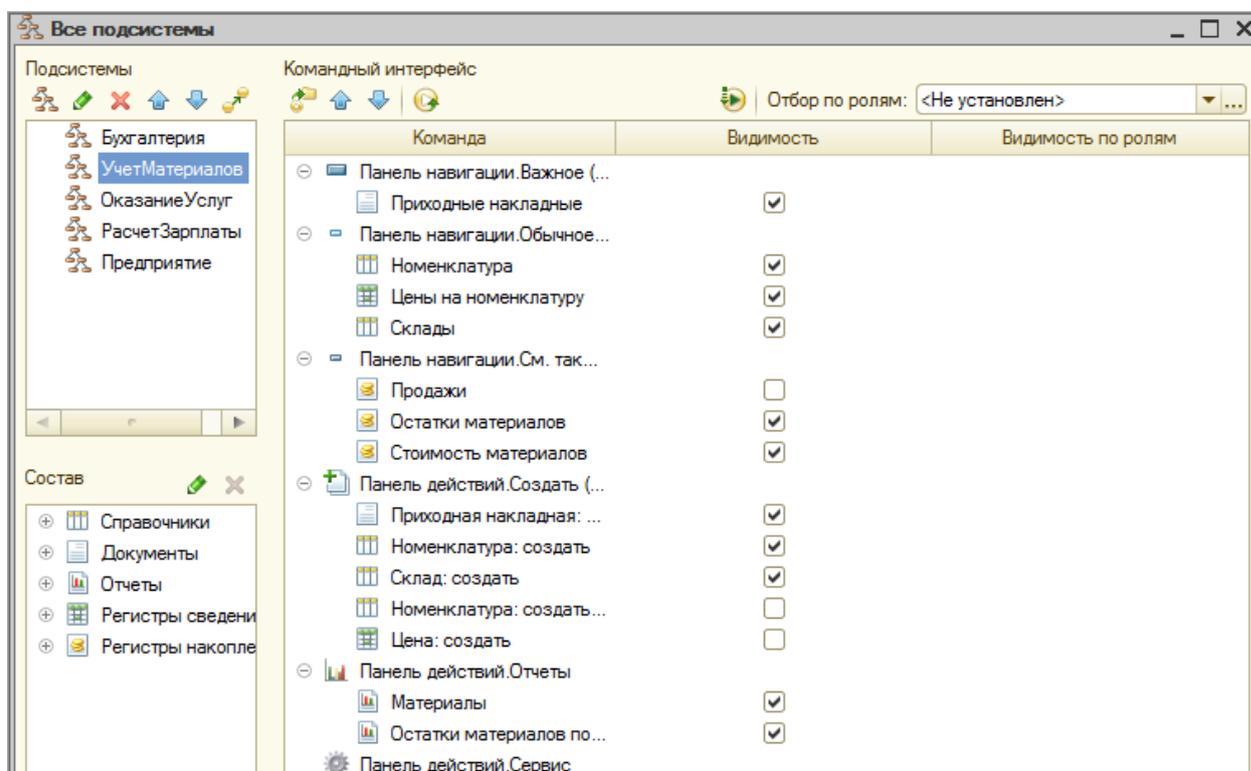


Рисунок 2.134 – Командный интерфейс подраздела «УчетМатериалов»

Таким же образом изменим командный интерфейс и в остальных подразделах.

Рабочий стол

Настроим рабочий стол. Добавим в левую колонку информацию об оказании услуг, а в правую информацию о приходных накладных и клиентах. Настройки рабочего стола представлены на рисунке 2.135.

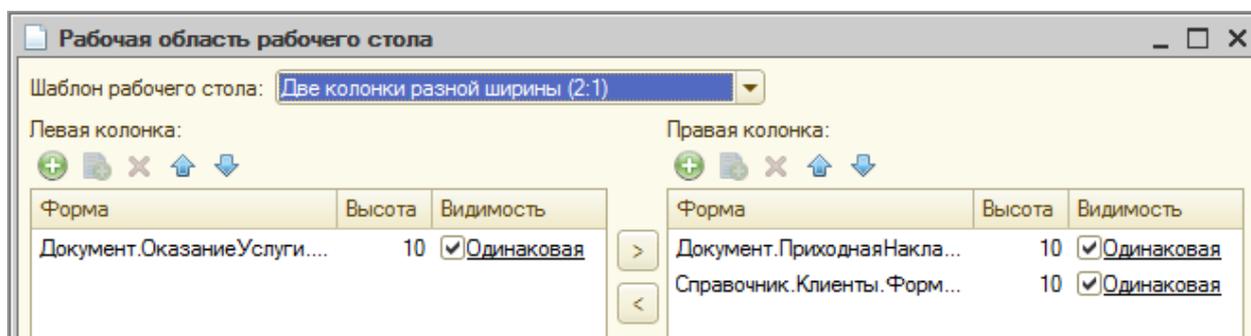


Рисунок 2.135 – Параметры рабочего стола

ЗАКЛЮЧЕНИЕ

На данный момент 1С является одной из передовых российских компаний, специализирующихся на дистрибуции, поддержке и разработке компьютерных программ и баз данных делового и домашнего назначения. В связи с развитием информационных технологий и продвижении многоплатформенности приложений 1С создает программные продукты как для различных ОС персональных компьютеров, так и для ОС мобильных устройств.

Целью курсовой работы является автоматизация документооборота ООО «На все руки мастер», а именно создание новой конфигурации «Мастерская» на платформе 1С:Предприятие.

Основные задачи курсовой работы – изучить на каких операционных системах работает 1С, что такое «1С-Битрикс» и создать конфигурацию «Мастерская» для ООО «На все руки мастер», которая содержит справочники (Клиенты, Сотрудники, Номенклатура, Склады, Варианты номенклатуры, Дополнительные свойства номенклатуры, Субконто, Виды графиков работы), документы (Приходная накладная, Оказание услуги, Начисления сотрудникам, Ввод начальных остатков номенклатуры), отчеты (Материалы, Реестр документов оказания услуг, Рейтинг услуг, Выручка мастеров, Перечень услуг, Рейтинг клиентов, Универсальный, Остатки материалов по свойствам, Оборотно-сальдовая ведомость, Перерасчет, Начисления сотрудникам, Диаграмма начислений), планы видов характеристик (Свойства номенклатуры, Виды субконто), план счетов (Основной), планы видов расчета (Основные начисления), регистры сведений (Цены, Значения свойств номенклатуры, Графики работы), регистры накопления (Остатки материалов, Стоимость материалов, Продажи), регистры бухгалтерии (Управленческий), регистры расчета (Начисления).

Таким образом, мы создали собственную конфигурацию «Мастерская» в системе 1С для ООО «На все руки мастер», в которой содержатся

инструменты и настройки свойственные для автоматизации документооборота на нашем предприятии и не содержится лишних инструментов, которые не используются.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Многоплатформ-ть. URL: http://v8.1c.ru/overview/Term_000000666.htm
(Дата обращения: 05.03.2018).

2 1С:Предприятие и MAC OS X. URL: <https://rapid-brain.ru/1c-mac-os-x>.
(Дата обращения 10.04.2018).

3 Радченко М. Г., Е. Ю. 1С:Предприятие 8.2 Практическое пособие разработчика, Москва: ООО «1С Пабблишинг», 2009 – 859 с.

4 Работа на мобильных устройствах. URL:
http://v8.1c.ru/overview/Term_000000804.htm. (Дата обращения: 12.05 2018).