

Назначение.

Модуль обмена с ПО Гифтоман представляет собой конфигурацию 1С и предназначен для обмена с ПО Гифтоман информацией о продажах.

Предназначается для конфигураций 1С: Розница 2.1 и 1С: Розница 2.2 версии ПРОФ клиент-серверный вариант.

Поставка включает в себя файлы Гифтоман 8_3 Розница 2_1, 2_2 клиент сервер.cf

Встраивание модуля обмена с ПО Гифтоман в 1С:Розница производится методом объединения конфигураций по подсистемам и ручной вставки кода. Также можно перенести объекты подсистемы «Гифтоман» предоставленной конфигурации вручную.

Важно: В зависимости от особенностей учета и модифицированности основной конфигурации 1С: Розница может потребоваться дополнительная адаптация под конкретную конфигурацию.

Важно: если конфигурация была типовой, после встраивания модуля конфигурация 1С: Розница будет нетиповой, автоматическое обновление конфигурации не проводить.

Важно: Работы по объединению конфигураций должен выполнять специалист .

Важно: Перед началом работ сделать архивную копию программы.

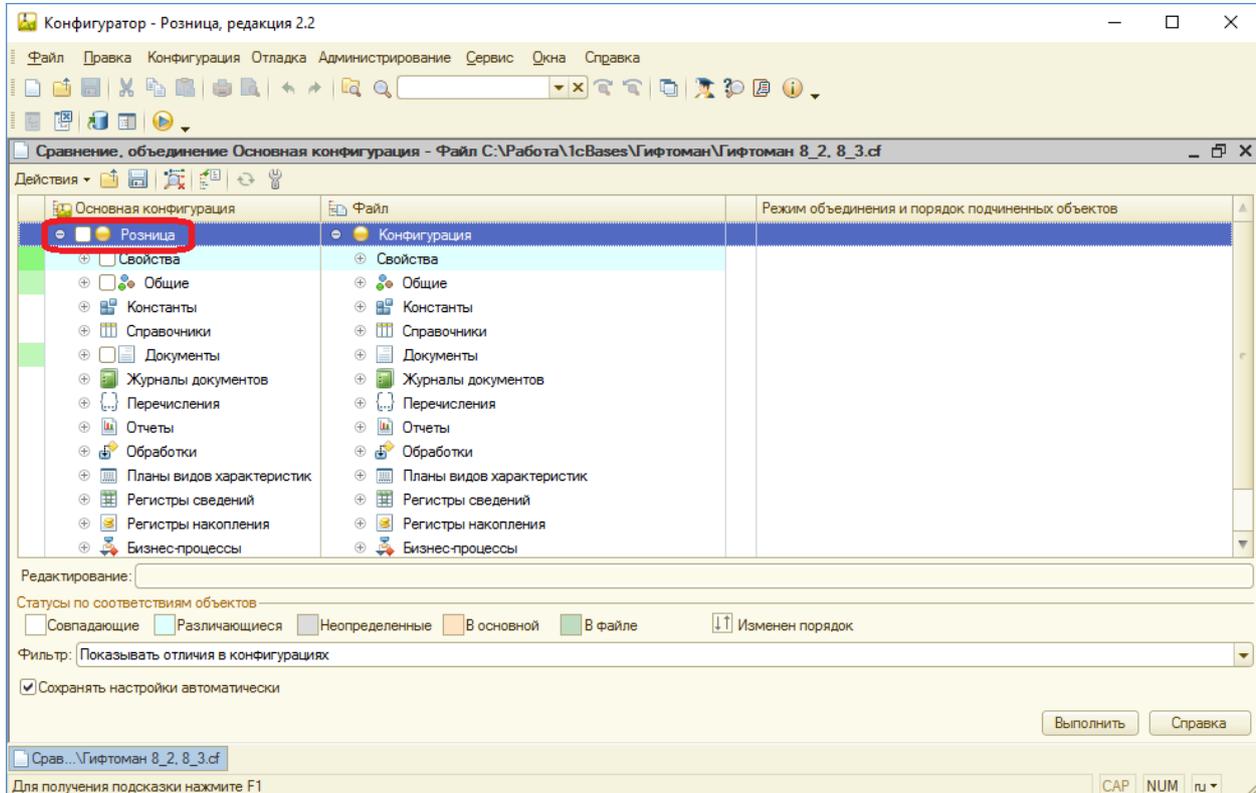
Важно: Проверить работоспособность на тестовой базе.

Особенности.

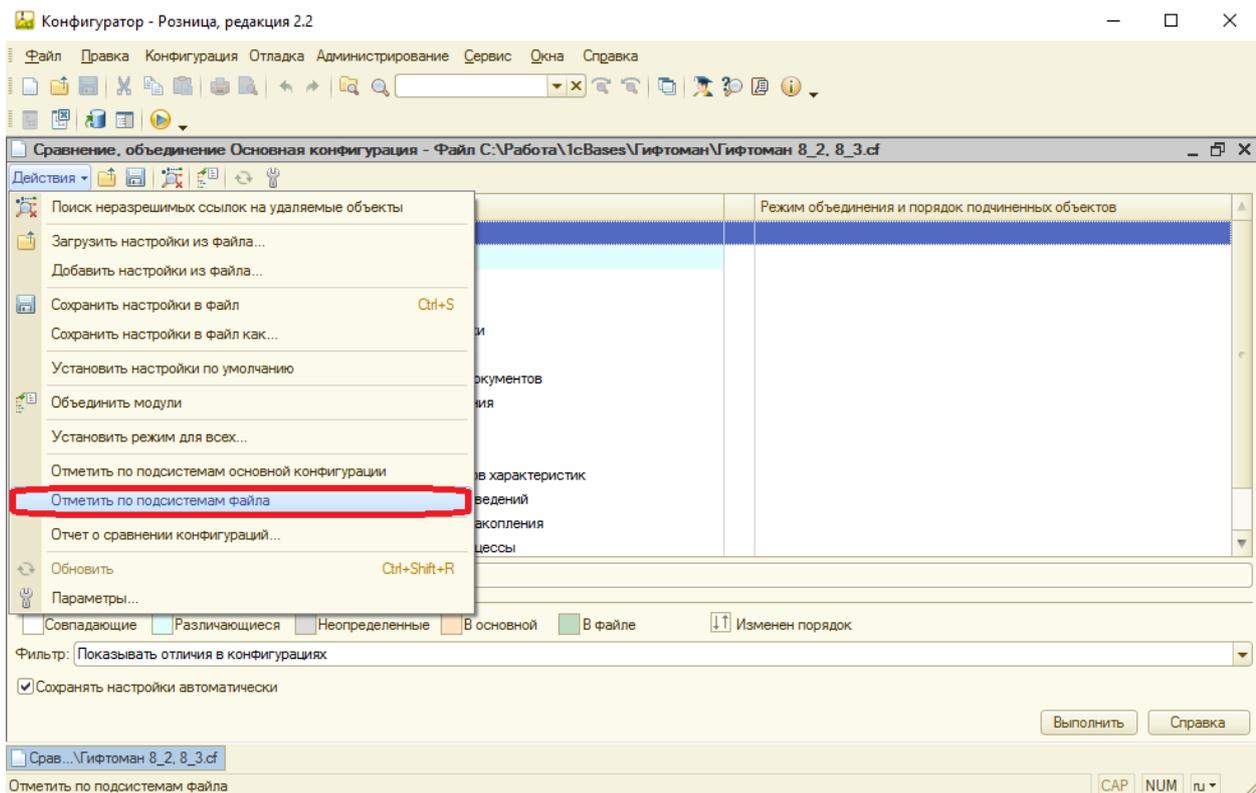
Подразумевается, что работа кассиров ведется в РМК кассира. Возврат от покупателей после закрытия осуществляется из РМК кассира через обработку «Возврат от розничного покупателя» . В ином случае встроить вызов функций в формы документов «ЧекККМ» и «ВозвратТоваровОтПокупателя».

Объединение по подсистеме

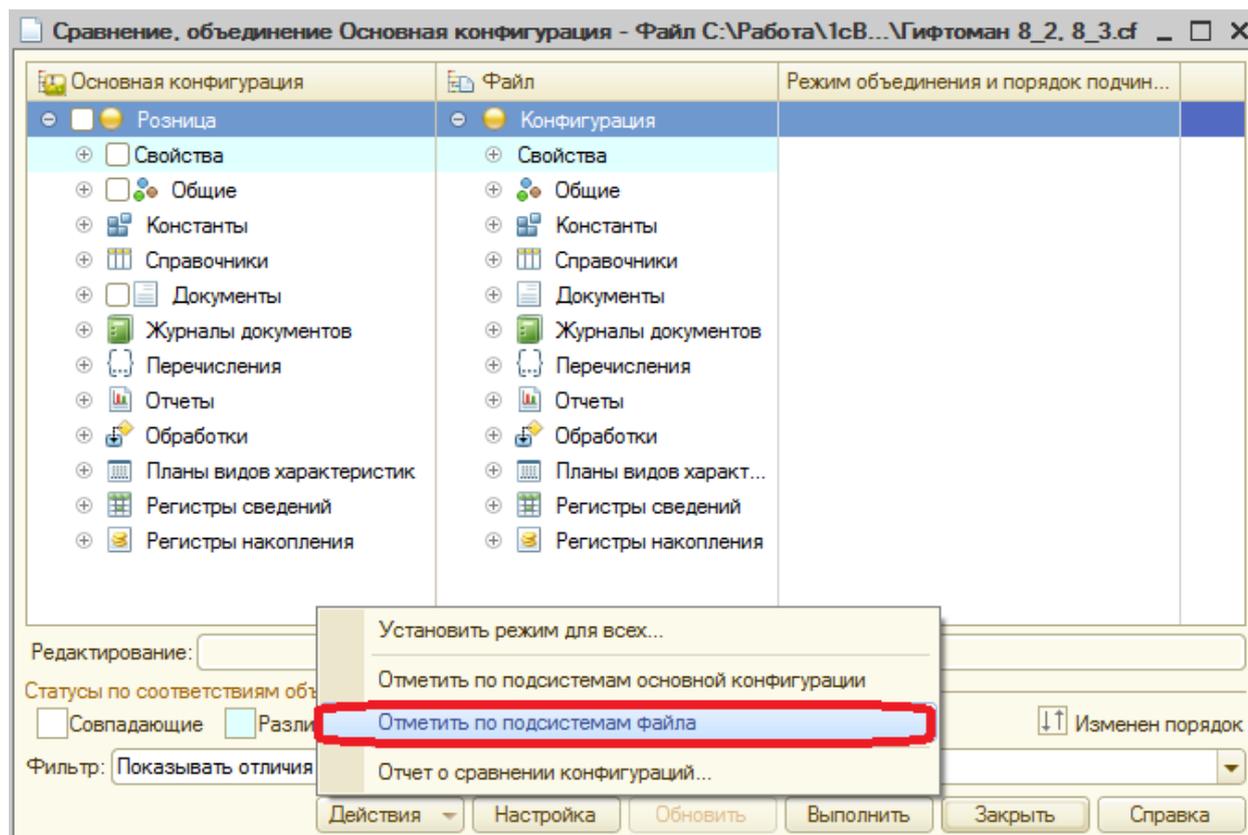
Нажимаем «Конфигурация»- «Сравнить, объединить с конфигурацией из файла... В открывшемся окне выбора файла - выбрать файл из поставки: файлы Гифтоман 8_3 Розница 2_1, 2_2 клиент сервер. В открывшемся окне снимаем галочку в поле «Розница»



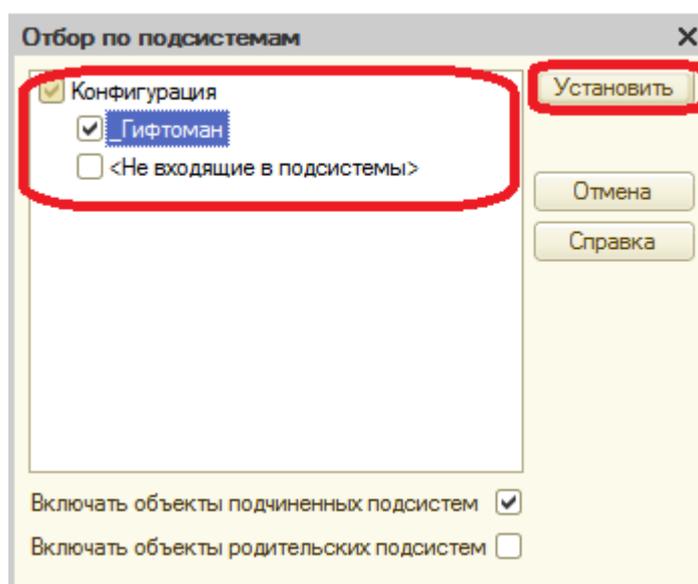
Выбираем в меню «Действия» - «Отметить по подсистемам файла»



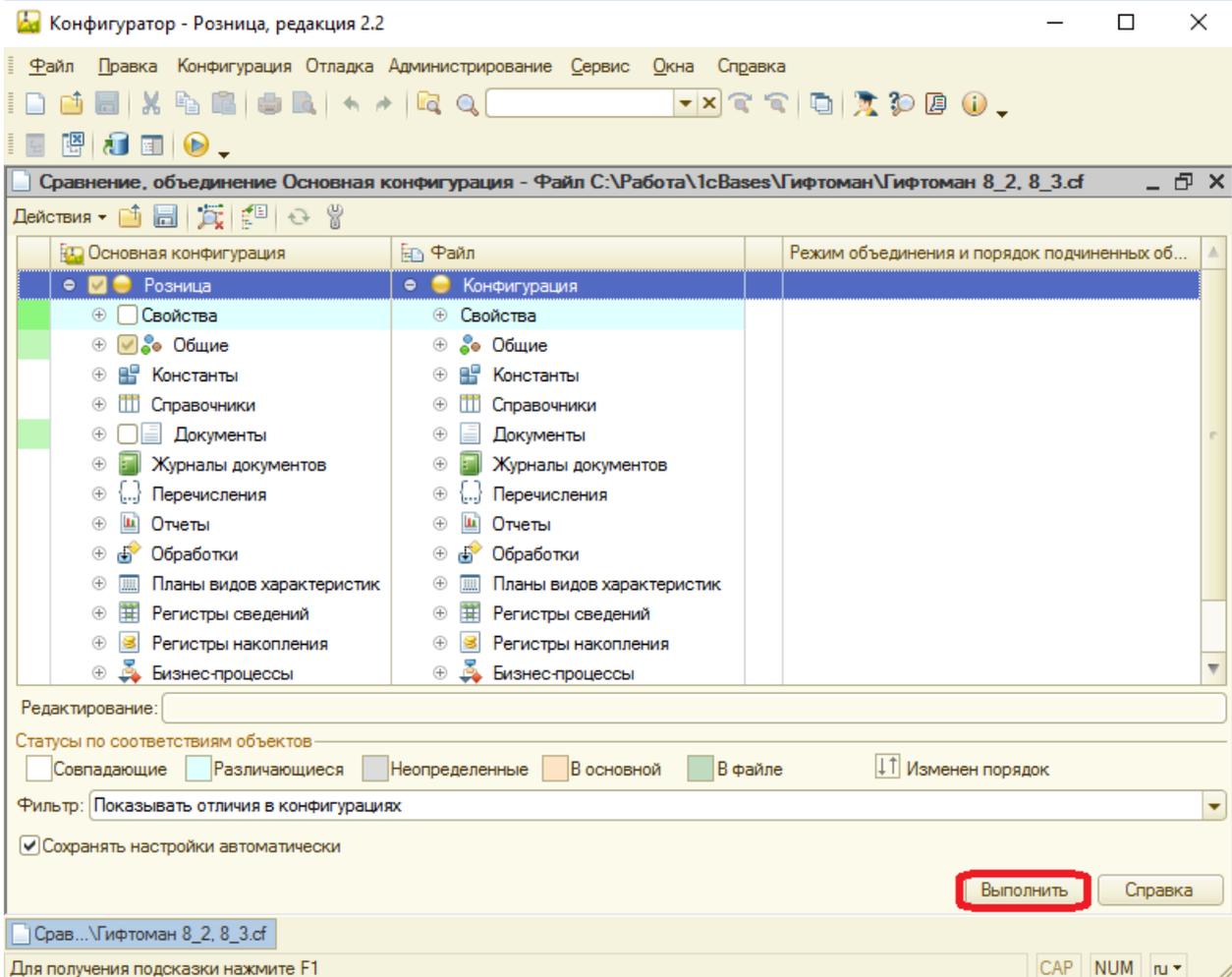
В зависимости от версии платформы пункт меню «Действия» может располагаться внизу.



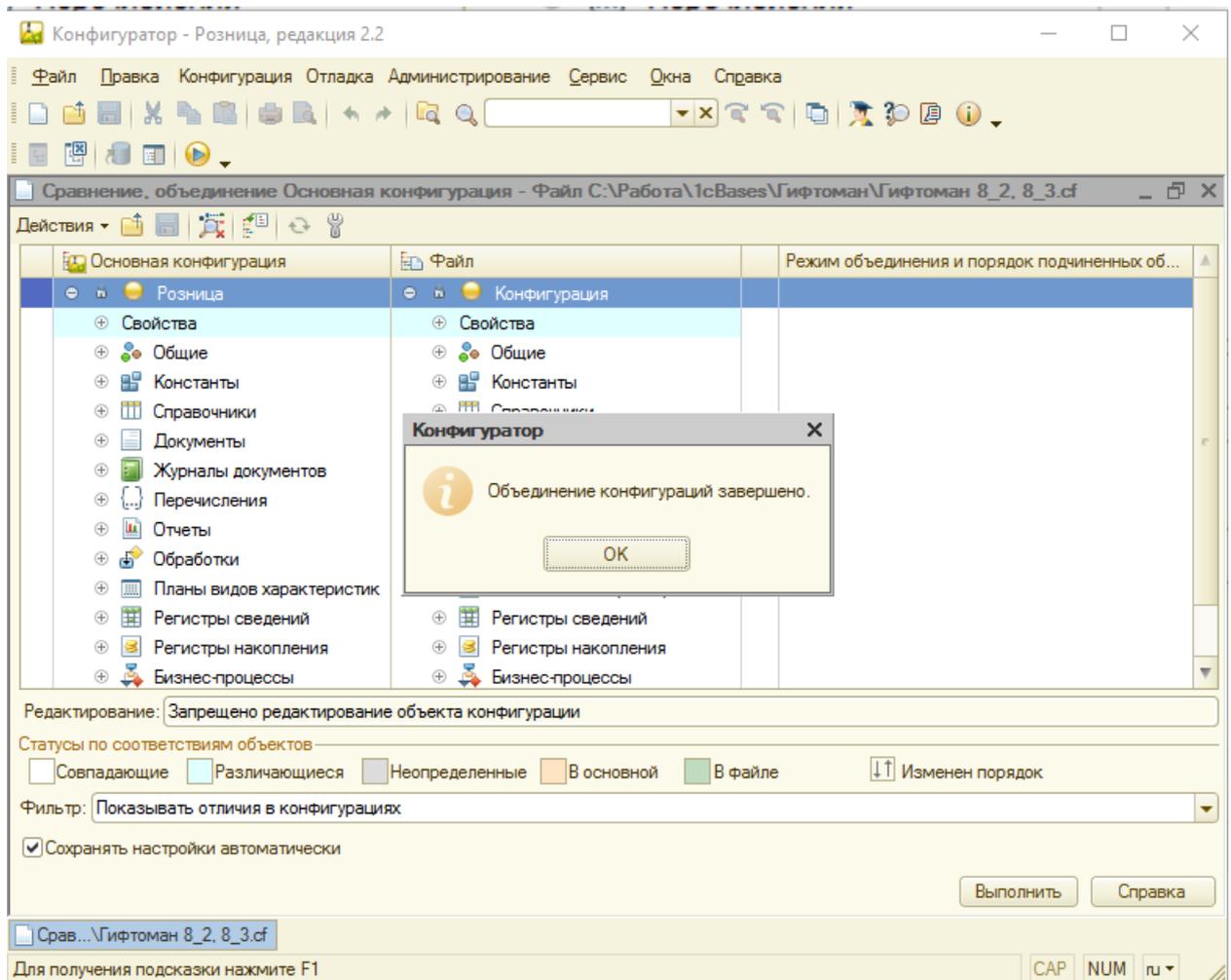
В открывшемся окне выбрать подсистему «Гифтоман», как показано на рисунке ниже и нажать кнопку «Установить».



Далее нажать кнопку «Выполнить».



После этого должно выйти сообщение:



Далее обновляем конфигурацию информационной базы.

Доработка объектов основной конфигурации.

В зависимости от особенностей(доработок) вашей конфигурации необходимо обеспечить вызов процедур предоставленного общего модуля _ГифтоманКлиент:

Возможны два варианта доработок. Вариант №1 можно использовать если в настройках пользователя указывается касса ККМ по умолчанию (обязательное условие), иначе использовать Вариант №2.

Вызов процедуры передачи данных после отбития чека на клиенте обработки «РМКУправляемый режим». (ОтправитьДанныеЧека(МассивЧеков))

Вызов процедуры передачи данных после создания документов возврата на клиенте из обработки «ВозвратТоваровОтРозничногоПокупателя». (ОтправитьДанныеВозвратаПокупателя(КассаККМ))

Для типовой конфигурации вызовы процедур предоставлены ниже.

Вариант 1

Вызов процедуры передачи данных после отбития чека на клиенте обработки «РМКУправляемый режим». (ОтправитьДанныеЧека(МассивЧеков))

Вызов процедуры передачи данных после создания документов возврата на клиенте из обработки «ВозвратТоваровОтРозничногоПокупателя». (ОтправитьДанныеВозвратаПокупателя(КассаККМ))

1.1 Изменения в обработке РМКУправляемый режим.

&НаКлиенте

Процедура СоздатьЧекиЗавершение(Отказ, ПечатьПакетаДокументовВозврат, МассивСсылокЧеков, Печать = Ложь)

Если Не Отказ Тогда

Если ПечатьПакетаДокументовВозврат Тогда

ИмяМакета = «КМЗ»;

ПолноеИмяОбъектаМетаданных = «Документ.ЧекККМ»;

УправлениеПечатьюКлиент.ВыполнитьКомандуПечати(ПолноеИмяОбъектаМетаданных, ИмяМакета, МассивСсылокЧеков, ЭтотОбъект, Неопределено);

КонецЕсли;

Если Печать Тогда

НапечататьЧекиККМ(МассивСсылокЧеков)

КонецЕсли;

КонецЕсли;

Если НЕ Отказ Тогда

ПорядковыйНомерПродажи = ПорядковыйНомерПродажи + 1;

ВыполнитьОбменДаннымиПоРабочемуМесту();

КонецЕсли;

//+++Гифтоман

Если Не отказ Тогда

_ГифтоманКлиент.ОтправитьДанныеЧека(МассивСсылокЧеков);

КонецЕсли;

//---Гифтоман

КонецПроцедуры

1.2 Изменения в обработке «ВозвратТоваровОтРозничногоПокупателя

Процедура СоздатьДокументы(Команда)

Отказ = НЕ ПроверитьЗаполнение();

Если Отказ Тогда

 Возврат;

КонецЕсли;

Результат = СоздатьДокументыСервер();

Если Результат Тогда

 МодифицированыСохраняемыеДанные = Ложь;

КонецЕсли;

УстановитьДоступностьСоздатьДокументы();

ПереключательОплатаСложнаяПриИзменении(Неопределено);

//+++Гифтоман

**КассаККМГифтоман = ПредопределенноеЗначение
("Справочник.КассыККМ.ПустаяСсылка");**

**ЗначениеНастроекВызовСервера.ЗаполнитьКассуККМПоУмолчанию(КассаКК
МГифтоман);**

**Если ЗначениеЗаполнено(КассаККМГифтоман) Тогда
 _ГифтоманКлиент.ОтправитьДанныеВозвратаПокупателя(КассаККМГифтома
н);**

КонецЕсли;

//---Гифтоман

КонецПроцедуры

Вариант2

2.1. Изменения в обработке РМКУправляемый режим.

1. &НаКлиенте

Процедура ВыполнитьПО16ВвестиВозвратТовараОтРозничногоПокупателя()

ВывестиИнформациюНаДисплейПокупателя(«ВозвратТовара»);

ПараметрыФормы = Новый Структура;

ПараметрыФормы.Вставить(«Магазин», Объект.Магазин);

Если НЕ ТаблицаКассККМОрганизаций.Количество() = 0 Тогда

ПараметрыФормы.Вставить(«Организация»,ТаблицаКассККМОрганизаций[0].Организация);

//+++Гифтоман

ПараметрыФормы.Вставить(«КассаККМГифтоман»,ТаблицаКассККМОрганизаций[0].КассаККМ);

//---Гифтоман

КонецЕсли;

ОбработчикОповещения = Новый

ОписаниеОповещения(«ОповещениеОткрытьФормуВозвратаТоваровОтРозничногоПокупателя»,
ЭтотОбъект);

Режим = РежимОткрытияОкнаФормы.БлокироватьВесьИнтерфейс;

ОткрытьФорму(«Обработка.ВозвратТоваровОтРозничногоПокупателя.Форма.Форма»,
ПараметрыФормы,,,,, ОбработчикОповещения, Режим);

КонецПроцедуры

2. &НаКлиенте

Процедура СоздатьЧекиЗавершение(Отказ, ПечатьПакетаДокументовВозврат,
МассивСсылокЧеков, Печать = Ложь)

Если Не Отказ Тогда

Если ПечатьПакетаДокументовВозврат Тогда

ИмяМакета = «КМЗ»;

ПолноеИмяОбъектаМетаданных = «Документ.ЧекККМ»;

УправлениеПечатьюКлиент.ВыполнитьКомандуПечати(ПолноеИмяОбъектаМетаданных,

ИмяМакета, МассивСсылокЧеков, ЭтотОбъект, Неопределено);

КонецЕсли;

Если Печать Тогда

 НапечататьЧекиККМ(МассивСсылокЧеков)

КонецЕсли;

КонецЕсли;

Если НЕ Отказ Тогда

 ПорядковыйНомерПродажи = ПорядковыйНомерПродажи + 1;

 ВыполнитьОбменДаннымиПоРабочемуМесту();

КонецЕсли;

//+++Гифтоман

Если Не отказ Тогда

_ГифтоманКлиент.ОтправитьДанныеЧека(МассивСсылокЧеков);

КонецЕсли;

//---Гифтоман

КонецПроцедуры

2.2 Изменения в обработке «ВозвратТоваровОтРозничногоПокупателя

1. Добавляем новый реквизит обработки:

КассаККМГифтоман тип СправочникСсылка.КассыККМ

2. &НаСервере

Процедура ИнициализироватьОбработку(Параметры)

 Объект.Дата = ТекущаяДатаСеанса();

 Объект.Основание = НСтр("ru = 'Возврат товаров'");

 Объект.Приложение = НСтр("ru = 'Заявление от'" + " " + Формат(Объект.Дата, "ДФ = DD");

 Объект.Контрагент = Константы.КонтрагентРозничныйПокупатель.Получить();

 Параметры.Свойство("Склад" ,Объект.Склад);

 Параметры.Свойство("Магазин",Объект.Магазин);

 Параметры.Свойство("Касса" ,Объект.Касса);

 Параметры.Свойство("Организация" ,Объект.Организация);

//+++Гифтоман

Параметры.Свойство("КассаККМГифтоман" ,Объект.КассаККМГифтоман);

//---Гифтоман

 Объект.Ответственный = Пользователи.ТекущийПользователь();

.....

3. &НаКлиенте

Процедура СоздатьДокументы(Команда)

Отказ = НЕ ПроверитьЗаполнение();

Если Отказ Тогда

 Возврат;

КонецЕсли;

Результат = СоздатьДокументыСервер();

Если Результат Тогда

 МодифицированыСохраняемыеДанные = Ложь;

КонецЕсли;

УстановитьДоступностьСоздатьДокументы();

ПереключательОплатаСложнаяПриИзменении(Неопределено);

//+++Гифтоман

Если Результат Тогда

**_ГифтоманКлиент.ОтправитьДанныеВозвратаПокупателя(Объект.КассаККМГифто
ман);**

КонецЕсли;

//---Гифтоман

КонецПроцедуры

Опционально

1. Если используется функция «Пробить непробитый чек» добавить следующий код в процедуру «ОповещениеОткрытьФормуВыбораЧекаПриПробитииЧекаЗавершение» обработки РМКУправляемый режим.

&НаКлиенте

Процедура

ОповещениеОткрытьФормуВыбораЧекаПриПробитииЧекаЗавершение(РезультатОперации, Параметры) Экспорт;

Если РезультатОперации.ЧекПробит Тогда

ЧекККМВОбработке =

ПредопределенноеЗначение("Документ.ЧекККМ.ПустаяСсылка");

Иначе

Возврат;

КонецЕсли;

ВывестиДополнительноеСообщение = Ложь;

ПараметрыИнформации =

ОбщегоНазначенияРТКлиентСервер.ПолучитьСтруктуруВыводимойВРМКИнформации();

ПровестиЧекККМ(Параметры.ЧекККМСсылка, РезультатОперации,

Параметры.МассивСсылокЧеков, ВывестиДополнительноеСообщение,

ПараметрыИнформации);

//+++Гифтоман

_ГифтоманКлиент.ОтправитьДанныеЧека(Параметры.МассивСсылокЧеков);

//---Гифтоман

.....

КонецПроцедуры

2. Если используется Проверка непробитых чеков при закрытии смены, доработать форму «ПроверкаНепробитыхЧеков» обработки ЗакрытиеКассовойСмены.

&НаКлиенте

Процедура ПробитиеЧековЗавершение(Результат, ПараметрыВыполнения) Экспорт

Если Результат Тогда

Если ПараметрыВыполнения.ТекущийЭлемент <

ПараметрыВыполнения.КоличествоЭлементов Тогда

// Следующая итерация рекурсии "оповещениями" пробитие документов.

ЭлементОсновнойТаблицы =

ОсновнаяТаблица.Получить(ПараметрыВыполнения.ТекущийЭлемент);

ПараметрыВыполнения.ТекущийЭлемент =
ПараметрыВыполнения.ТекущийЭлемент + 1;

Если ЭлементОсновнойТаблицы.Пометка И
ЭлементОсновнойТаблицы.ВозможностьИзменения Тогда

ОповещениеПриЗавершении = Новый ОписаниеОповещения("ПробитиеЧековЗавершение",
ЭтотОбъект, ПараметрыВыполнения);

ДокументНаККМ = ЭлементОсновнойТаблицы.ДокументНаККМ;

Если ЭлементОсновнойТаблицы.ВидДокумента = "ЧекККМ" Тогда

ПробитьЧекВыполнить(ОповещениеПриЗавершении, ДокументНаККМ);

//+++Гифтоман

_ГифтоманКлиент.ОтправитьДанныеЧека(ДокументНаККМ);

//---Гифтоман

.....

КонецПроцедуры