

Sphinx format for Latex and HTML

Meher Krishna Patel

Created on : October, 2017

Last updated : мая, 2023

More documents are freely available at [PythonDSP](#)

Описание:

Описание:	i
Список иллюстраций	ii
Список таблиц	iii
1 Общее описание	1
1.1 Документация по блоку обмена между Artix и УНФ в части выгрузки остатков из УНФ в кассовый сервер Artix и загрузки кассовых смен и продаж в УНФ.	1
2 Реализация в УНФ	3
3 Внешняя программа	6
4 Файлы	7
5 Порядок внедрения касс в работу с Артикс:	8
5.1 1. Создать магазин + касса на сервере.	8
5.2 2. Настройка ПОС	8
6 Добавление группы сертификатов со сроком действия через REST API.	9
7 Работа с весами	12
8 Дополнительные материалы:	16
8.1 1. Список директорий и файлов основных настроек.	16
8.2 2. Особенности настройки “Сбербанк”	16
8.3 3. Ссылка на скачивание дистрибутива uuki	17
8.4 4. Остатки	17
8.5 5. Дисконтные карты (создание)	17
8.6 6. Настройка действий и контекста клавиш	17
8.7 7. Конфигурирование дисконтной системы	17
9 Индексы и таблицы	21

Список иллюстраций

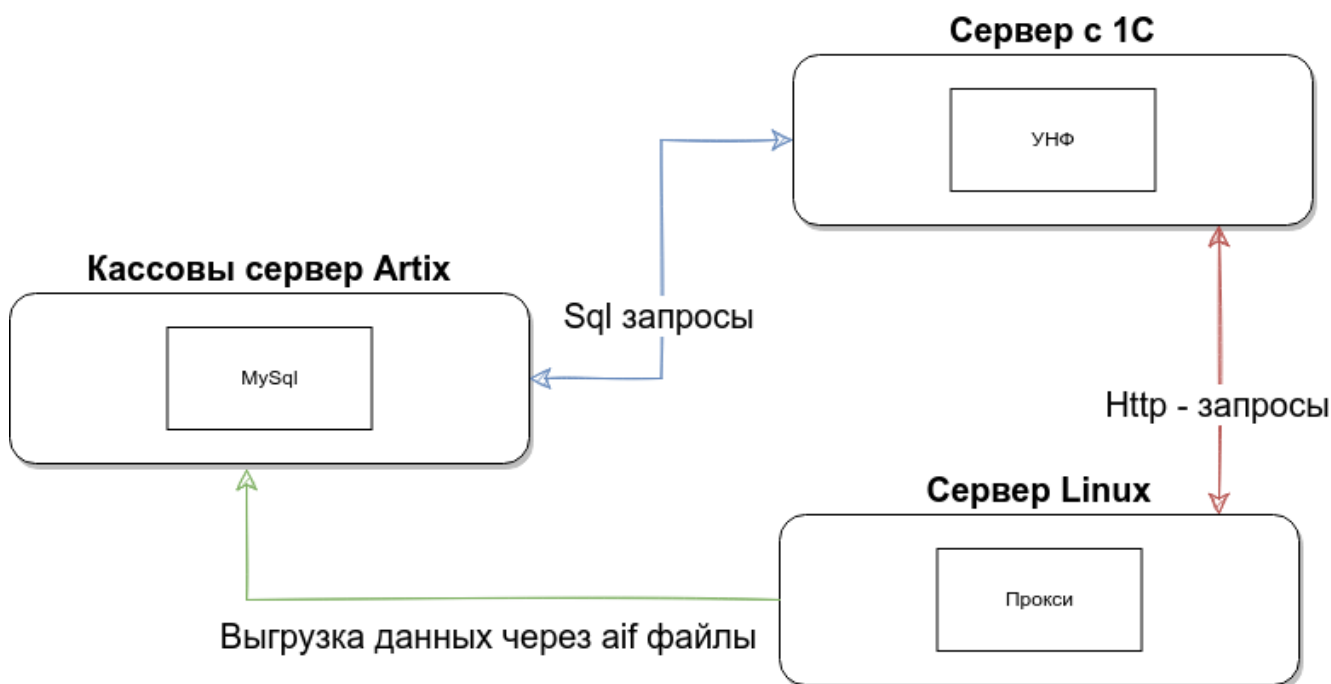
Список таблиц

Глава 1

Общее описание

1.1 Документация по блоку обмена между Artix и УНФ в части выгрузки остатков из УНФ в кассовый сервер Artix и загрузки кассовых смен и продаж в УНФ.

Связь между 1С и Artix осуществляется с помощью Http-сервисов реализованных на стороне УНФ. Данные по продажам 1С забирает из базы данных Кассового сервера с помощью Sql запросов. Таблицы из БД Кассового сервера подключены в УНФ в качестве внешнего источника данных. Прокси сервер реализован на Python и располагается на отдельном сервере с Linux.



Работа происходит следующим образом: В УНФ в справочнике «крю_Настройки» в предопределенном реквизите «КаталогКонтроляОбменаArtix» хранится путь до каталога интеграции с Artix, который имеет следующую структуру -

При проведении документа товародвижения срабатывает расширение «крюВыгрузкаАртикс», которое проверяет интерактивное проведение документа или нет и в случае интерактивного проведения в каталоге соответствующем номеру магазина создается файл-флаг вида: «1681466893_chg», где число это псевдослучайное число образованное разностью дат, для достижения уникальности имени файла.

Номенклатура относящаяся к кулинарии (категория номенклатуры «Кулинария»), выгружается с **пустым**

/	
unf_teleport.	корневой каталог
21.	каталог магазина (соответствует номеру магазина)
22.	каталог магазина (соответствует номеру магазина)
23.	каталог магазина (соответствует номеру магазина)
25.	каталог магазина (соответствует номеру магазина)
78.	каталог магазина (соответствует номеру магазина)

полем остатка, так как всегда продается в минус. Устанавливать поле остатка в «0» нельзя, так Artix считает это за остаток равный «0» и соответственно не дает продавать.

Важно: У номенклатуры для выгрузки в Artix должна быть установлена цена, остаток больше нуля или пустой (для кулинарии), штрихкод с установленной единицей измерения, единица измерения упаковки. При отсутствии какого-то из перечисленных свойств, позиция номенклатуры не выгрузится в Artix

Программа Brocker запускаясь с указанной настройкой сервера периодичностью начинает обрабатывать кассовые смены. Вначале в УНФ передаются данные по вновь открытым сменам в Artix. УНФ получая информацию создает новые документы «Кассовая смена» со статусом «Открытая» и автоматом создается документ «ОРП» у которого заполнена только шапка и имеется ссылка на кассовую смену. После этого Brocker отправляет информацию по закрытым кассовым сменам. УНФ заполняет оставшиеся реквизиты в документе «Кассовая смена» и присваивает документу статус «Закрыта (чеки за архивированы)». Далее согласно созданному в УНФ расписанию запускается обработка по формированию «ОРП». Данные для заполнения табличных частей «ОРП» берутся запросом из кассового сервера, таблицы которого подключены в 1С в качестве внешнего источника данных.

Далее программа прокси отправляет запрос («GetChangeShop») в УНФ для получения номеров магазинов по которым были изменения остатков в ту или иную сторону. УНФ «просматривает» каталоги в папке интеграции и возвращает список номеров магазинов в каталогах которых, были обнаружены файлы флаги. Затем программа Brocker перебирая номера магазинов формирует в УНФ запрос остатков по каждому. Получив остатки, Brocker обрабатывает данные - для номенклатуры с аналогами переносит суммируя все остатки на головную номенклатуру, а аналоги удаляет из выгрузки, таким образом на кассы падает в остатках, только головная номенклатура. После формирования остатков, на основе этих данных формируется файл формата aif и выгружается в соответствующий каталог магазина. Кассовый сервер через определенное время «забирает» файл и формирует номенклатуру с остатками на магазине.

План: Планируется добавить модуль контроля товаро-остатков пришедших из УНФ и выгруженных в кассовый сервер

Глава 2

Реализация в УНФ

Функционал работы с Artix со стороны УНФ реализован с помощью:

1. Расширения **«крюВыгрузкаАртикс»**. Расширение учитывает следующие виды документов

- 1) Поступление товаров
- 2) Списание товаров
- 3) Оприходование товаров
- 4) Перемещение товаров
- 5) Возврат товаров поставщику
- 6) Отчет о розничных продажах

В расширении для формы документа реализована процедура **«ПослеЗаписиНаСервере»** с аннотацией **«&После»**, что позволяет выполниться коду из расширения, после выполнения основного кода данной процедуры формы. Реализовано для того, что бы изменения остатков фиксировались только при интерактивном проведении документа и не и не обрабатывались при проведении программном.

```
крюОбменСArtixСервер.БылоИзменениеУчетногоКоличества(ТекущийОбъект, ПараметрыЗаписи);
```

Вызывается процедура **«БылоИзменениеУчетногоКоличества»** общего модуля **«крюОбменСArtixСервер»**

```
Процедура БылоИзменениеУчетногоКоличества(ТекущийОбъект, ПараметрыЗаписи) Экспорт
    Если ПараметрыЗаписи.РежимЗаписи = РежимЗаписиДокумента.Проведение ИЛИ ПараметрыЗаписи.
    →РежимЗаписи = РежимЗаписиДокумента.ОтменаПроведения Тогда
        Если ТекущийОбъект.Ссылка.Метаданные().Имя = "ПеремещениеЗапасов" Тогда
            ЗаписатьФлагИзменений(ТекущийОбъект.СтруктурнаяЕдиница);
            ЗаписатьФлагИзменений(ТекущийОбъект.СтруктурнаяЕдиницаПолучатель);
        Иначе
            ЗаписатьФлагИзменений(ТекущийОбъект.СтруктурнаяЕдиница);
        КонецЕсли;
    КонецЕсли;
КонецПроцедуры
```

Процедура проверяет режим записи документа и тип и вызывает функцию **«ЗаписатьФлагИзменений»**, для записи файла флага в каталог с имя которого совпадает с номером магазина переданного документа.

```
Функция ЗаписатьФлагИзменений(Магазин) Экспорт
    НомерМагазина = ПолучитьНомерМагазина(Магазин);
    ПутьКФайлам = Справочники.крю_Настройки.ВернутьЗначениеНастройки(
    →"КаталогКонтроляОбменаArtix");
    ПрефиксФайла = Формат(ТекущаяДата() - дата(1970,1,1,1,0,0), "ЧГ=0");
    ИмяФайлаВыгрузки =ПутьКФайлам+ НомерМагазина + "\" + ПрефиксФайла + "_chg.flg";
    ТекстовыйФайл = Новый ТекстовыйДокумент;
```

(continues on next page)

(продолжение с предыдущей страницы)

```
ТекстовыйФайл.Записать(  
    ИмяФайлаВыгрузки // путь для сохранения  
);  
Возврат Неопределено;  
КонецФункции
```

2. Http-сервисы:

- Http-сервис Workshift
- GetChangeShop
- GetProductRemains

3. Фоновых заданий и обработок

Обработка по формированию ОРП подключается в дополнительные обработки и для нее настраивается расписание запуска. Обработка ищет непроведенные документы ОРП, которые связаны с документом «Кассовая смена», со статусом «Закрыта». И выполняет заполнение этих документов продажами из таблиц базы данных «Кассового сервера» Artix, подключенными в программу, как внешние источники данных.

Настройки публикации базы

Публикация на веб-сервере [X]

Основные OpenID Прочие

Имя:

Веб-сервер:

Каталог:

☒ Публиковать доступ для клиентских приложений

☒ Публиковать стандартный интерфейс OData

☒ Публиковать систему аналитики





Web-сервисы HTTP сервисы

☒ Публиковать HTTP сервисы по умолчанию

	Имя	Корневой URL
<input checked="" type="checkbox"/>	ПередачаДанных	dt
<input checked="" type="checkbox"/>	СервисЛояльности	loyaltyservice
<input checked="" type="checkbox"/>	СинхронизацияМПУНФ	synchronization
<input checked="" type="checkbox"/>	УдалитьvatsYandexTelephonyApi	yandex-api
<input checked="" type="checkbox"/>	ЭДО	edi
<input checked="" type="checkbox"/>	test_s	test_s
<input checked="" type="checkbox"/>	ctrl	ctrl
<input checked="" type="checkbox"/>	GetChangeShop	GetChangeShop
<input checked="" type="checkbox"/>	Workshift	Workshift

☐ Публиковать HTTP сервисы расширений по умолчанию

☐ Публиковать дистрибутив тонкого клиента

Расположение публикуемого дистрибутива:    

Адрес перехода при окончании работы веб-клиента:

Наименование прогрессивного веб-приложения:

Опубликовать Отключить Сохранить Загрузить Закреть Справка

Глава 3

Внешняя программа

Внешняя программа реализована на языке Python и расположена на отдельном сервере с установленной серверной операционной системой Linux Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-69-generic x86_64)

Программа состоит из двух блоков, первый отвечает за получение остатков товаров из обработки и передачи на кассовый сервер Artix, второй за отслеживание открытия и закрытия смен на кассах и периодической передачи этих данных в УНФ.

Общее описание

1. Установка и настройка

Программа стартует с блока «main». При старте проверяется наличие конфигурационного файла strset и в случае его отсутствия программа завершает свою работу выводя сообщения в файл логов. Если файл настроек прочитан, то вызывается функция «main» и программа начинает работу.

```
m_conf = m_config.m_Config()
rc = m_conf.loadConfig()
if not rc == None:
    main()
else:
    logger.error('Configuration file not found')
    logger.info('The program has finished its work')
```

Вначале происходит работа с кассовыми сменами

2. Кассовые смены

После обработки кассовых смен, выполняется проверка и при необходимости загрузка остатков.

3. Работа с остатками

Глава 4

Файлы

Скачать эту документацию в pdf.

Глава 5

Порядок внедрения касс в работу с Артикс:

5.1 1. Создать магазин + касса на сервере.

1. Создание Магазина - Вводить Код, Название, Формат справочников (из 1С), метка дисконта (выбор из созданных)
2. Создание Кассы - Вводить Код, Название, Хост (IP), Тип кассы (Рабочее место кассира), Формат выгрузки продаж (с КС в 1С), Категория справочников (если используется), Состояние обмена (рекоменд. Выключен). При использовании Конфигураций - Выбрать конфигурацию, и окружение (для рабочей кассы - Боевое)
 - Создать структуру Магазин-Кассы можно/нужно заранее.

5.2 2. Настройка ПОС

Полуручной:

1. Самый простой способ предварительно настроить 1 пос, и сохранить нужные файлы настроек (настройки, оборудование и т.д.). Далее скопировать на другую ПОС в требуемые директории.

Автоматически:

1. Использовать менеджер-конфигураций (что есть в КС), одна тонкость - возможно не все настройки можно произвести через него.
2. Включить обмен на сервере и запустить полную выгрузку справочников с сервера. После полной загрузки справочников - перезапустить кассовую программу на ПОС

Глава 6

Добавление группы сертификатов со сроком действия через REST API.

1. Добавление группы сертификатов со сроком действия через REST API.

- a. Добавить группу сертификатов в БД КС, сформировав запрос типа POST на url: <http://<host>:<port>/Csrest/rest/dictionaries/certificates>
- b. Тело запроса в формате Json должно быть подобного формата:

```
{
    "code": "1",
    "rangefrom": 30,
    "rangeto": 40,
    "name": "Сертификаты",
    "sum": 300,
    "inputmask": 31,
    "discountcampaign": "test campaign"
}
```

- c. Атрибут свойства discountcampaign должен содержать yaml акции, в которой как раз можно указать сроки действия. Например, если смотреть на уже существующую группу сертификатов, созданную вручную через web-интерфейс LM с указанием срока действия (БД КС ArtixAll, таблица certificateTemp), то в поле discountcampaign увидим запись вида:

```
!!python/object:artixds.domain.DiscountCampaign
active: true
beginBonusesDateCalculate: null
beginBonusesIntervalCondition: !!python/unicode 'date'
beginBonusesIntervalMetric: !!python/unicode 'day'
beginBonusesIntervalValue: 0
beginDate: null
beginDateWithBeginCurrentMetric: !!python/unicode 'day'
beginDateWithBeginNextMetric: !!python/unicode 'day'
beginRangeDate: 2022-02-17
beginTime: null
checkedTags: null
daysOfWeek: []
discounts: []
endBonusesDateCalculate: null
endBonusesIntervalCondition: !!python/unicode 'date'
endBonusesIntervalMetric: !!python/unicode 'day'
endBonusesIntervalValue: 0
endDate: null
```

(continues on next page)

(продолжение с предыдущей страницы)

```

endRangeDate: 2022-02-17
endTime: null
id: 57026161735168
interactionType: all
labels: null
manualActivate: null
name: !!python/unicode ''
parent: null
parentGroup: null
priority: null
qualifiers: !!set {}
tagsIsAllowed: false
weight: 300

```

То есть в этом случае срок действия для группы сертификатов определен свойствами beginRangeDate и endRangeDate (от 2022-02-17 до 2022-02-17).

Берем этот Yaml как шаблон и изменяем даты beginRangeDate и endRangeDate на нужные.

- d. Далее этот Yaml необходимо переделать в однострочный формат. Для этого после каждой строки добавляем `rn`, и переносим все строки в одну. В итоге должна получиться строка следующего вида:

```

!python/object:artixds.domain.DiscountCampaign\r\nactive: true\r\n
↪nbeginBonusesDateCalculate: null\r\nnbeginBonusesIntervalCondition: !!python/
↪unicode 'date'\r\nnbeginBonusesIntervalMetric: !!python/unicode 'day'\r\n
↪nbeginBonusesIntervalValue: 0\r\nnbeginDate: null\r\n
↪nbeginDateWithBeginCurrentMetric: !!python/unicode 'day'\r\n
↪nbeginDateWithBeginNextMetric: !!python/unicode 'day'\r\nnbeginRangeDate: 2022-
↪02-17\r\nnbeginTime: null\r\nncheckedTags: null\r\ndaysOfWeek: []\r\ndiscounts:
↪[]\r\nendBonusesDateCalculate: null\r\nendBonusesIntervalCondition: !!python/
↪unicode 'date'\r\nendBonusesIntervalMetric: !!python/unicode 'day'\r\n
↪nendBonusesIntervalValue: 0\r\nnendDate: null\r\nnendRangeDate: 2022-02-17\r\n
↪nendTime: null\r\nnid: 62331387691573\r\nninteractionType: all\r\nnlabels: null\r\n
↪nmanualActivate: null\r\nnname: !!python/unicode ''\r\nnparent: null\r\n
↪nparentGroup: null\r\nnpriority: null\r\nnqualifiers: !!set {}\r\nntagsIsAllowed:
↪false\r\nnweight: 300

```

И в таком виде эту строку можно добавить в тело POST-запроса (пункт 1.1) вместо записи test campaign.

2. Отправляем POST-запрос, указав данные для авторизации (как от WEB КИ)

Welcome	Request1
---------	----------

POST ▾

http://192.169.1.159:8080/Csrest/rest/dictionaries/certificates

JSON ▾	HEADERS	QUERY STRING	BASIC ▾	ASSERTION ▾
--------	---------	--------------	---------	-------------

USERNAME

admin

PASSWORD

admin

☒ Show Password

Далее убеждаемся что запрос прошел без ошибок и проверяем в web-интерфейсе LM, что создалась группа сертификатов. Открываем эту группу на редактирование и убеждаемся, что сроки действия прописаны корректно.

[Не сохранять группу сертификатов](#)

Редактирование группы сертификатов

Код группы	<input type="text" value="3"/>	Номинал	
Название группы *	<input type="text" value="Сертификаты"/>	Минимальная сумма	
Начало диапазона номеров	<input type="text" value="30"/>	Максимальная сумма	
Окончание диапазона номеров	<input type="text" value="40"/>	Кратность	<input type="text"/>
Код налога	<input type="text"/>	Код товара	<input type="text"/>
Признак предмета расчета	<input type="text"/>	Код отдела	<input type="text"/>
Начало действия сертификатов - 17 февраля 2022 Окончание действия сертификатов - 17 февраля 2022		<input type="checkbox"/> Полное списание <input type="checkbox"/> Игнорировать после	

[Изменить](#)

3. Затем добавляем сертификат на сервер POST-запросом `http://<хост>:<порт бонусного сервера/сервера сертификатов>/ACC/rest/v1/cards`.

Тело запроса должно быть вида:

```
{
  "number": "30",
  "accountNumber": "30",
  "status": "EARN_PAY"
}
```

Важно, чтобы значения `number` и `accountNumber` входили в диапазон номеров созданной группы сертификатов.

Глава 7

Работа с весами

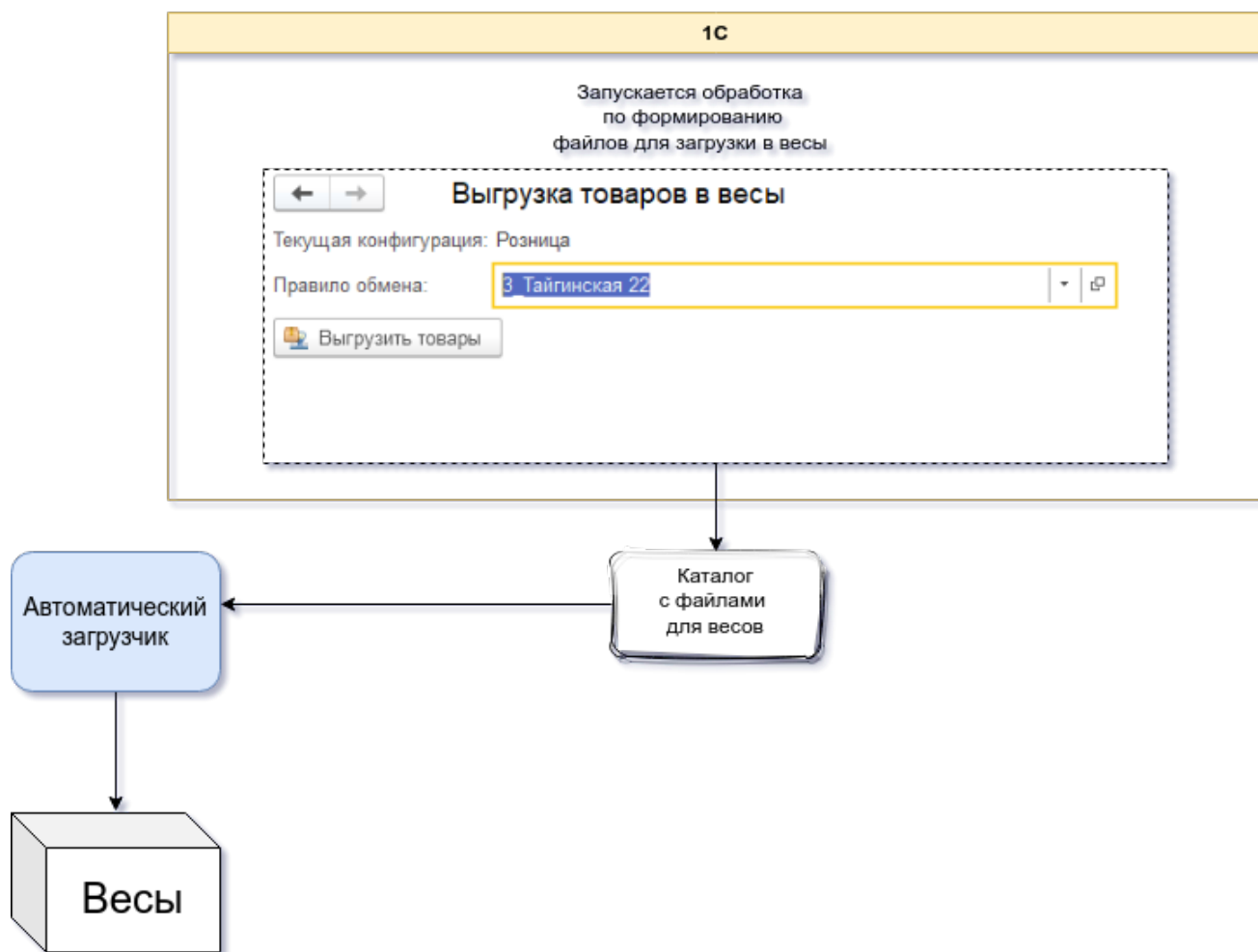
Текущий вариант выгрузки в весы



Пример файла выгрузки формируемый из 1С

```
82;Шейки цыплят в/к;;550;0;0;0;10007;0;0;0;01.01.01;0;0;0;0;01.01.01
2082;Шейки цыплят в/к;;550;0;0.006;0;10007;0;0;0;01.01.01;0;0;0;0;01.01.01
3082;Шейки цыплят в/к;;550;0;0.008;0;10007;0;0;0;01.01.01;0;0;0;0;01.01.01
4082;Шейки цыплят в/к;;550;0;0.012;0;10007;0;0;0;01.01.01;0;0;0;0;01.01.01
5082;Шейки цыплят в/к;;550;0;0.02;0;10007;0;0;0;01.01.01;0;0;0;0;01.01.01
6082;Шейки цыплят в/к;;550;0;0.014;0;10007;0;0;0;01.01.01;0;0;0;0;01.01.01
```


Вариант выгрузки в весы используя типовой механизм Штрих-м



Формат загрузки и выгрузки товаров

Каждая строка файла загрузки товаров должна быть в кодировке ОС Windows CP1251 и должна соответствовать формату:

[PLU#][P][ITEM NAME 1][P][ITEM NAME 2][P][UNIT PRICE][P][LIFE][P][TARE][P][GROUP CODE][P][ITEM CODE][P][MESSAGE#][P][IMAGE#][P][CERTIFICATE][P][SALE DATE][P][ITEM TYPE][P][BC STRUCTURE][P][LABEL FORMAT][P][BC PREFIXTYPE][P][P][PRODUCTION DATE]

Обозначение поля	Название поля	Представляет тип	Формат/длина
PLU#	Номер ПЛУ	Целое число	XXXXX
ITEM NAME 1	Наименование 1	Строка	28 символов
ITEM NAME 2	Наименование 2	Строка	28 символов
UNIT PRICE***	Цена товара	Дробное число	XXXX.XX
LIFE	Срок годности, в днях	Целое число	XXXX
TARE***	Тара, в кг	Дробное число	XX.XXX
GROUP CODE	Групповой код товара	Целое число	XXXX
ITEM CODE	Код товара	Целое число	XXXXXX
MESSAGE#	Номер сообщения	Целое число	XXXX
IMAGE***	Номер изображения	Целое число	X
CERTIFICATE	Код РОСТЕСТ	Строка	4 символа
SALE DATE	Дата реализации	Дата	ДД.ММ.ГГ
ITEM TYPE*	Тип товара	Целое число	X
BC STRUCTURE*	Структура ШК	Целое число	XX
LABEL FORMAT*	Формат этикетки	Целое число	XX
BC PREFIXTYPE*	Тип префикса	Целое число	X
PRODUCTION DATE	Дата изготовления	Дата	ДД.ММ.ГГ
P****	Разделитель	Символ	

* Возможные значения: для ITEM TYPE - 0 (весовой товар) или 1 (штучный товар); для BC STRUCTURE - 0 (по умолчанию), 1..14 (форматы ШК); для LABEL FORMAT - 0 (по умолчанию), 1..15 (форматы этикетки); для BC PREFIXTYPE - 0 (по умолчанию), 1 (весовой префикс), 2 (штучный префикс), 3 (итоговый префикс), 4 (префикс GS1).

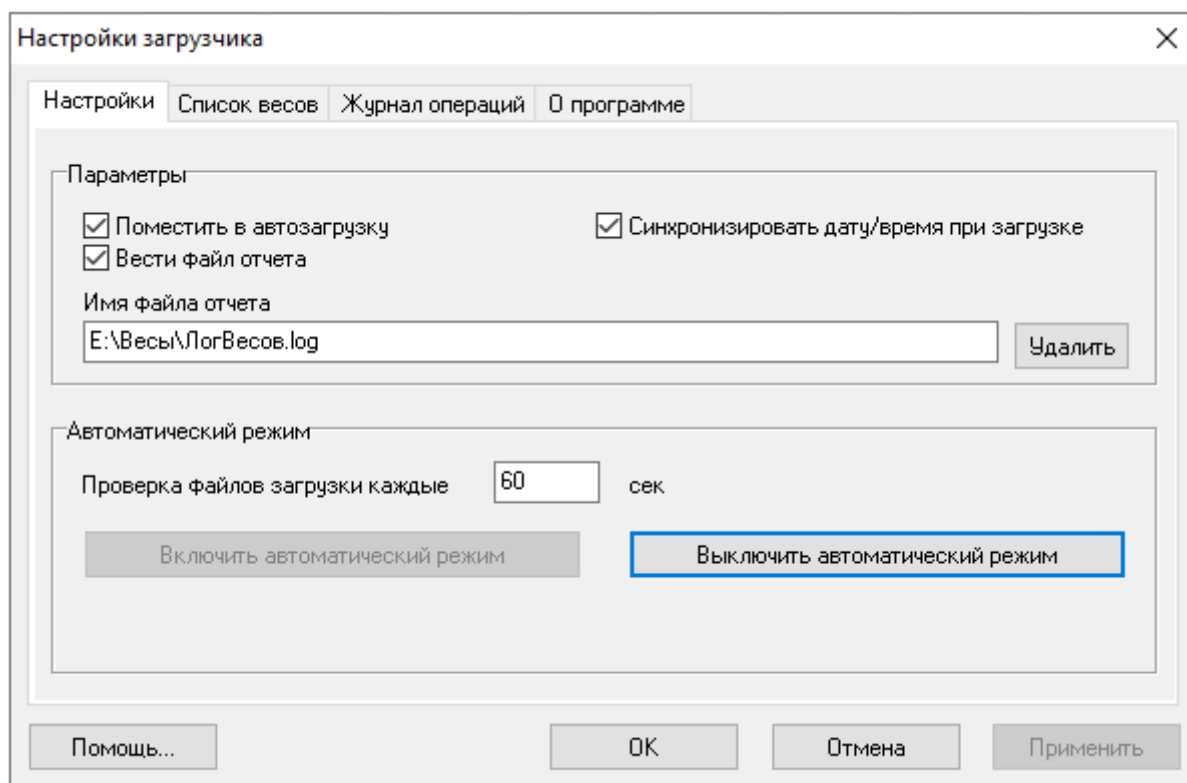
** Является битовым полем: бит 0 - изображение 1, бит 1 - изображение 2, бит 2 - изображение 3, бит 3 - изображение 4. Остальные биты зарезервированы и должны быть равны 0.

*** В качестве разделителя целой и дробной частей используется символ из настроек Windows.

**** В качестве разделителя в файлах должен быть использован символ ";".

Настройка автоматического загрузчика

Указываем путь до файлов лога, включаем автоматический режим и устанавливаем интервал запуска.



Формируем список весов для загрузки

Настройка весов

Настройка весов

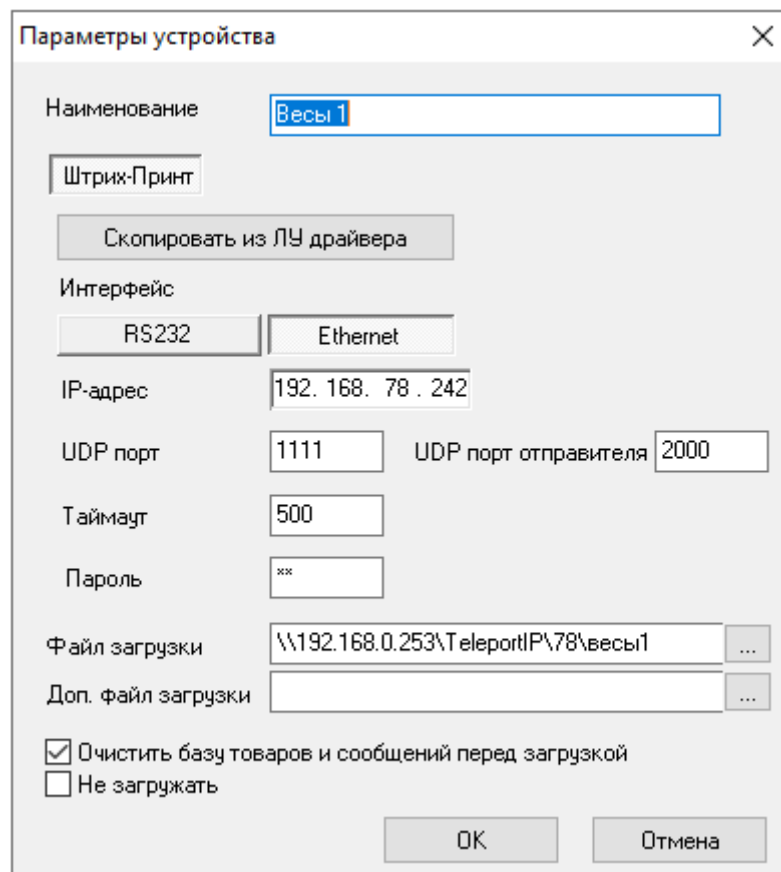
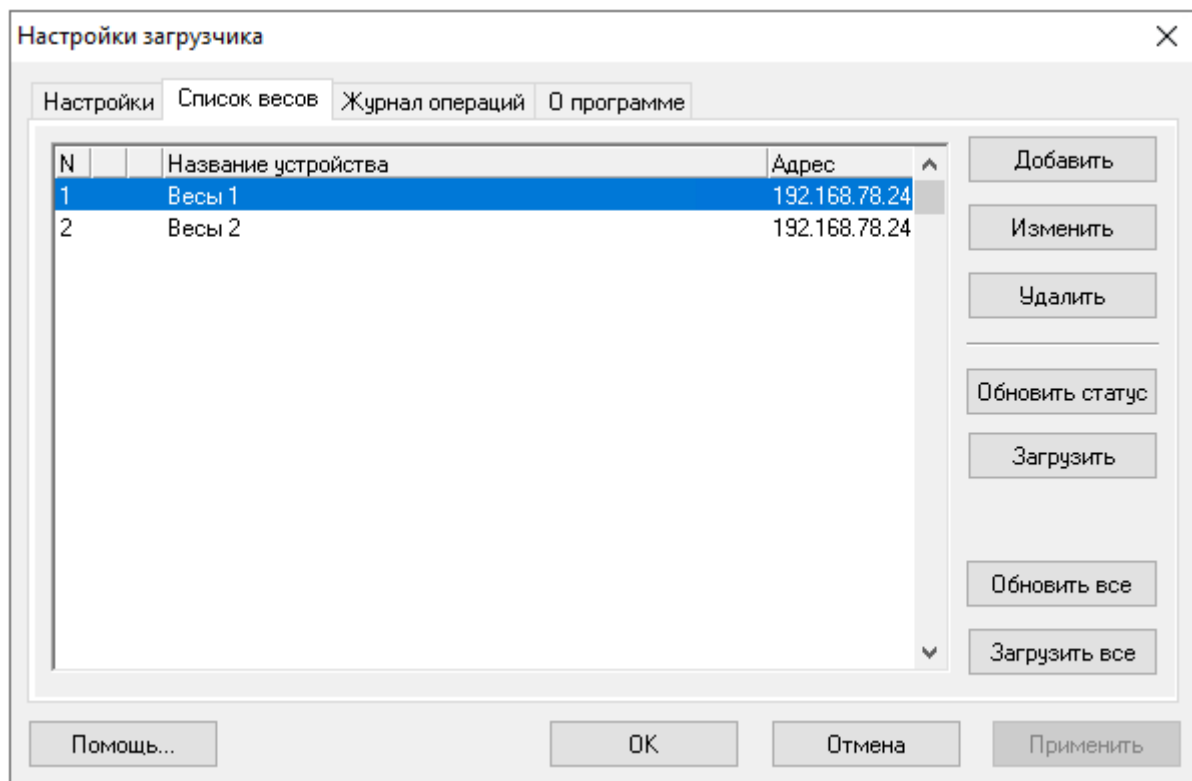
Необходимо в весах установить префикс весового товара в «20».

Настройки штрихкодирования на весах:

Правила разбора штрих-кода фиксируются в конфигурационном файле `/linuxcash/cash/conf/bcode.ini` и применяются в заданном порядке. Также осуществляется считывание конфигурации `bcodeparser` из директории `/linuxcash/cash/conf/bcode.ini.d/` (считывание файлов происходит в алфавитном порядке). Итоговая конфигурация выводится в основной лог программы `/linuxcash/logs/current/terminal.log`.

Внимание: При указании правил разбора штрих-кода в конфигурационном файле `/linuxcash/cash/conf/bcode.ini` и в файлах директории `/linuxcash/cash/conf/bcode.ini.d/` сначала происходит считывание файла `/linuxcash/cash/conf/bcode.ini`.

Ссылка на сайт с [документацией](#) (рус.)



Глава 8

Дополнительные материалы:

8.1 1. Список директорий и файлов основных настроек.

1. `/linuxcash/cash/conf/`
 - 1) `ncash.ini` - основной файл настроек
 - 2) `bcode.ini` - правила разбора ШК
 - 3) `frinit.conf` - инициализация ФР
 - 4) `cliche.txt` и `tail.txt` - Клише/подвал чека
 - 5) `artixsb.conf`
2. `/linuxcash/cash/conf/ncash.ini.d/`
 - 1) `artixbonus.ini` - работа с бонусным сервером
 - 2) `artixcertificate.ini`
 - 3) `deferredcheck.ini` - отложенные чеки
 - 4) `fr.ini`
 - 5) `gui.ini` - выбор интерфейса ПО
 - 6) `motp.ini` - сигареты
 - 7) `markedgoods.ini` - другая маркировка
3. `/linuxcash/cash/conf/drivers/` - Настройки подключения оборудования

Сначала настраивается одна ПОС, полученные настройки используем как эталонные. После установки ПО на новую ПОС копируем настройки, включая “Подключаемое оборудование”, проверяем и при необходимости вносим правки.

8.2 2. Особенности настройки “Сбербанк”

Для работы с процессингом сбербанк необходимо установить пакет - *artix45-plugin-sb*. В директорию `"/linuxcash/cash/paysystems/sb/"` разместить файлы приложения сбербанк полученного от сотрудников банка. Настроить подключение ПинПад согласно инструкции банка (`pinpad.ini`). Следует обратить внимание что при подключении пинпад через USB, необходимо правильно создать символьную ссылку на устройство:

```
ln -s /dev/serial/by-id/{Имя_Устройства} /linuxcash/cash/paysystems/sb/ttyS99
```

Для работы с терминалом PAX SP30 - дополнительно установить пакет *ttypos*

8.3 3. Ссылка на скачивание дистрибутива yuki

<http://update.artix.su/static/yuki/>

8.4 4. Остатки

в конфигурационном файле `/linuxcash/cash/conf/ncash.ini` Изменить:

```
;calculateRemainsOfGoods = false
.....
;considerRemainsOfGoods = false
```

на:

```
calculateRemainsOfGoods = true
.....
considerRemainsOfGoods = true
```

После изменения настроек надо перезапустить кассовое По. Тогда касса будет считать остатки и учитывать их при продаже

Для товара, у которого не нужно контролировать остатки (кулинария) при выгрузке не указываем дату остатков - *remaindate*, а поле *remain* оставляем **пустым!!!** (не «0»!!!).

8.5 5. Дисконтные карты (создание)

1. создать счет на бонусном сервере
2. привязать номер карты к счету
3. создать карту на кассовом сервере (группа должна быть создана заранее)

1 и 2 это работа с апи бонусного сервера

3 это работа с файлом айф, или через рестапи загрузки справочников с Кассовым севером.

<https://docs.artix.su/pages/viewpage.action?pageId=27530936>

8.6 6. Настройка действий и контекста клавиш

8.7 7. Конфигурирование дистконтной системы

Ссылка на сайт с полной [инструкцией](#)

Дисконтная система поставляется отдельным пакетом `artix45-discountsystem`.

Наличие минимальной цены влияет на методы распределения скидок на чек и наборы. Минимальные цены не учитываются на этапе расчета скидок, дополнительное ограничение минимальной цены накладывается на этапе применения.

Сумма скидки на чек распределяется в два этапа:

- распределение по позициям пропорционально стоимости каждой позиции,
- размещение остатка максимально возможными суммами начиная с первой позиции.

Правила распределения остатка определяется параметром **distributeReceiptDiscountFully** в конфигурационном файле `/linuxcash/cash/conf/ncash.ini` в секции `[Discount]`:

Клавиши			
Код	Значение	Действие	Контекст
1068	ALT+D	Ввести 00	Все режимы
330	DEL	Сторно текущей позиции	Формирование товарной части документа
324	ALT+F12	Подитог	Формирование товарной части документа
320	ALT+F8	Выбрать по наименованию	Формирование товарной части документа
319	ALT+F7	Выбрать по штрих-коду	Формирование товарной части документа
318	ALT+F6	Карта: добавить	Формирование товарной части документа
316	ALT+F4	Сторно текущей позиции	Формирование товарной части документа
315	ALT+F3	Возврат по чеку продажи	Формирование товарной части документа
314	ALT+F2	Сторно всех позиций	Формирование товарной части документа
295	CTRL+F7	Выход	Формирование товарной части документа
288	SHIFT+F12	Применить модификаторы к позиции (Кол-во и.т.д.)	Формирование товарной части документа
278		Выбрать тип оплаты	Оплата документа
276	F12	Изменить количество	Формирование товарной части документа
272	F8	Оплатить бонусами	Оплата документа
271	F7	Подитог по карте	Формирование товарной части документа
270	F6	Выбрать тип оплаты	Оплата документа
84	SHIFT+T		Формирование товарной части документа
83	SHIFT+S		Формирование товарной части документа
82	SHIFT+R		Формирование товарной части документа
81	SHIFT+Q		Формирование товарной части документа
80	SHIFT+P		Формирование товарной части документа
79	SHIFT+O		Формирование товарной части документа
78	SHIFT+N		Формирование товарной части документа
77	SHIFT+M	Пицца3	Формирование товарной части документа
76	SHIFT+L		Формирование товарной части документа
75	SHIFT+K		Формирование товарной части документа
74	SHIFT+J		Формирование товарной части документа
73	SHIFT+I		Формирование товарной части документа
72	SHIFT+H		Формирование товарной части документа
71	SHIFT+G		Формирование товарной части документа
70	SHIFT+F	Келлерс	Формирование товарной части документа
69	SHIFT+E	Пицца1	Формирование товарной части документа
68	SHIFT+D		Формирование товарной части документа
67	SHIFT+C		Формирование товарной части документа
66	SHIFT+B		Формирование товарной части документа
65	SHIFT+A		Формирование товарной части документа
47	/	Итог	Все режимы

- при значении параметра `true` остаток скидки распределяется максимально возможными суммами, учитывая ограничения минимальной ценой товара. Выполняется только первый этап распределения скидки;
- при значении параметра `false` сумма скидки на чек распределяется по позициям пропорционально их стоимости.

Размер скидки на позицию ограничивается минимальной ценой товара. В случае предоставления скидок на наборы минимальную цену на товары можно игнорировать для того, чтобы иметь возможность задавать акции вида «купи две штуки и одну получи в подарок», когда скидка назначается только на один из товаров. Возможность игнорирования минимальной цены при предоставлении скидки на наборы задается параметром **ignoreMinimalPriceForKit** в конфигурационном файле `/linuxcash/cash/conf/ncash.ini` в секции `[Discount]`:

- при значении параметра `true` минимальная цена при применении скидки на набор игнорируется,
- при значении параметра `false` скидка на комплект будет предоставлена с учетом минимальной цены на товар.

В основе дисконтной системы заложена идея использования любых доступных элементов чека (товарная позиция, дисконтная карта и т.д.) для проверки условий и вычисления ставок. Так как условия скидок могут быть сложными, то после изменения состава чека выполняется отмена ранее примененных скидок, последующий пересчет и применение новых. Пересчет скидок может занимать продолжительное время, это может происходить из-за недостатка ресурсов машины, на которой осуществляется работа, или из-за сложности самих скидок. Кассовое ПО Artix может быть настроено таким образом, чтобы расчет скидок происходил как в процессе формирования чека, так и только при переходе к оплате документа. Момент расчета скидок задается параметром **recalcDiscInFormingMode** в конфигурационном файле `/linuxcash/cash/conf/ncash.ini` в секции `[Check]`:

- при значении параметра `true` скидки пересчитываются в режиме формирования документа,
- при значении параметра `false` скидки пересчитываются при переходе в подитог.

Если расчет скидок происходит в режиме формирования документа, то пересчет скидок выполняется после:

- добавления или сторнирования товарной позиции,
- добавления или удаления карты,
- изменения модификаторов (цена, количество и т.п.) товарной позиции.

При использовании акций со ставкой 0% (например, рекламных) можно контролировать запись информации о сработавшем дисконтом воздействии на позицию в БД `Documents` таблицу `Discitem`. Возможность сохранения скидки 0% на позицию в БД задается параметром **saveNullDiscountForPosition** в конфигурационном файле `/linuxcash/cash/conf/ncash.ini` в секции `[Discount]`:

- при значении параметра `true` скидки 0% на позицию сохраняются в БД,
- при значении параметра `false` скидки 0% на позицию не сохраняются в БД.

Наименование	Тип данных	Возможные значения	Описание	Примечания
<code>distributeReceiptD</code>	логический	<code>true</code> , <code>false</code>	Полностью распределять по позициям сумму скидки на чек	По умолчанию <code>true</code>
<code>ignoreMinimalPriceForKit</code>	логический	<code>true</code> , <code>false</code>	Игнорировать минимальную цену при применении скидки на комплект	По умолчанию <code>true</code>
<code>recalcDiscInFormingMode</code>	логический	<code>true</code> , <code>false</code>	Пересчитывать скидки в режиме формирования чека	По умолчанию <code>true</code>
<code>saveNullDiscountForPosition</code>	логический	<code>true</code> , <code>false</code>	Сохранять скидку 0% на позицию в базу данных	По умолчанию <code>false</code>

S

Пример настроек

[Discounts]

```
;
; Скидки
;
...
; Полностью распределять по позициям сумму скидки на чек
distributeReceiptDiscountFully = false
...
; Сохранять скидку 0% на позицию в базу данных
saveNullDiscountForPosition = true
; Игнорировать минимальную цену при применении скидки на комплект
ignoreMinimalPriceForKit = false
```

[Check]

```
;
; Настройки чека
;
...
; Пересчитывать скидки в режиме формирования чека, по-умолчанию true.
recalcDiscInFormingMode = true
```


Глава 9

Индексы и таблицы

- `genindex`
- `modindex`
- `search`

Что реализовать:

План: Планируется добавить модуль контроля товаро-остатков пришедших из УНФ и выгруженных в кассовый сервер

([исходный элемент](#) находится в `/home/bat/Project/docs/Artix_Global/source/description.rst`, строка 52.)