

Мобильная платформа 1С:Предприятие 8 – первые шаги

Аналогичная статья опубликована на ресурсе infostart.ru.

Текущая версия статьи модифицирована и актуализирована под новые версии (на февраль 2014 года).

Оглавление

Что такое мобильная платформа 1С и с чем ее едят?

Где работает данная платформа?

Что может мобильная платформа?

Скорость работы мобильной платформы

Выбор устройства для работы с мобильной платформой

Подготавливаем софт для работы с мобильной платформой

Настройка платформы 1С

Установка Apache

Установка Apache ant

Установка Java JDK

Установка мобильной платформы для разработчика

Установка Android эмулятора AVD и Android SDK

Установка приложения в Андроид

Пишем и переносим простейшую конфигурацию на мобильное устройство

Основные свойства, которые надо заполнить у конфигурации

Платформа для разработчика

Конфигурация для компиляции приложения

Мобильная платформа, конфигурация, обновление

Каким образом можно разрабатывать под мобильную платформу

Обновляемая конфигурация

Фиксированная конфигурация

Что такое мобильная платформа 1С и с чем ее едят?

Где работает данная платформа?

Android устройства:

- Версия Android не ниже 2.2;
- Доступной памяти у телефона не меньше 100 Мб.

iOS устройства:

- iPhone – 3GS и выше;
- iPad – 1,2,3;
- iPod Touch – начиная с 3 поколения.

На ПК в режиме эмуляции, при помощи программы BlueStacks (это самый простой вариант):

- Скачать с сайта www.bluestacks.com программу;
- Установить на ПК с ОС Windows XP и выше или Mac OSX.

Отдельно дополню – желательно любое новое устройство с Андроид 4+, либо «флагманы» прошлых годов. На остальных «железках» программа хоть и работает, но только с небольшими объемами данных.

Внимание! Вся информация, описанная в статье, касается только ОС Андроид!

Версия платформы 1С 8.3.4.365, версия мобильной платформы 1С 8.3.4.9.

Что может мобильная платформа?

Тут следует немного отвлечься от 1С и разобраться в политике безопасности Андроида.

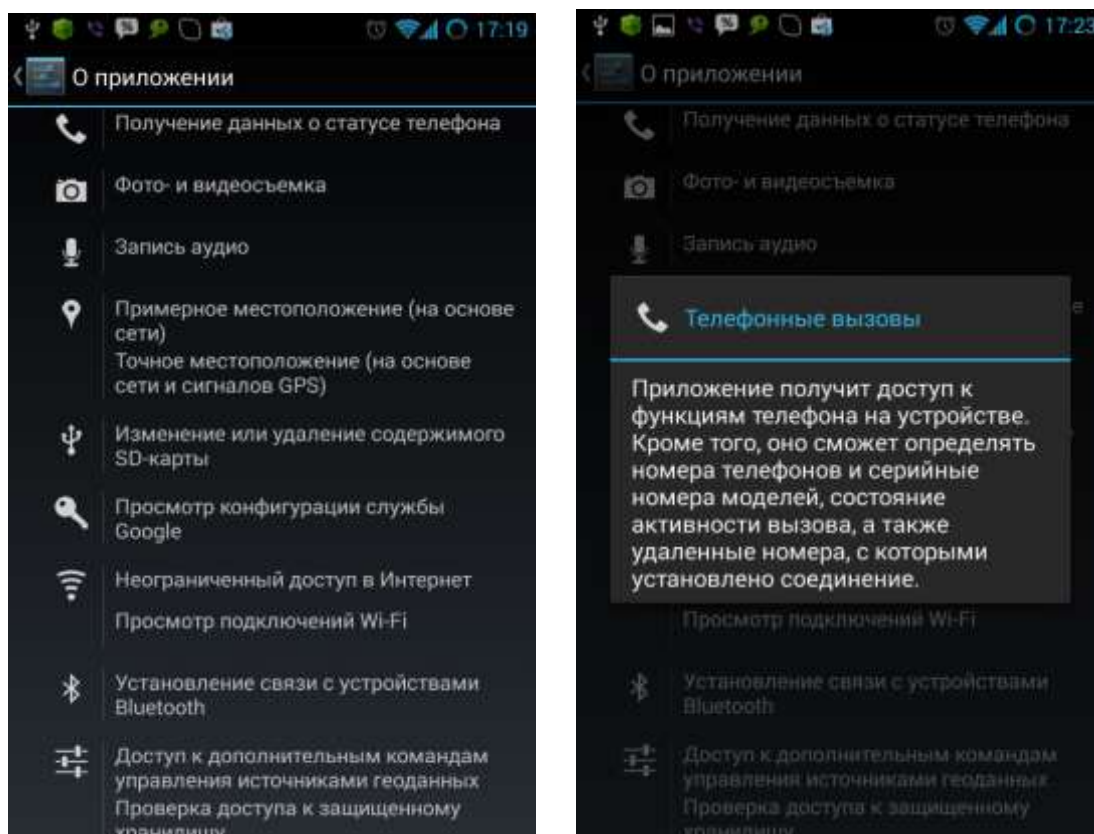
А тут все достаточно прозрачно, основная идея заключается в том, что в манифесте вашего приложения должны быть прописаны все права доступа к телефону. Т.е. если вы реализовали в вашем приложении возможность работы с видеокамерой, но не прописали ее в манифесте, то работать с ней вы не сможете.

Что же это нам дает? А дает следующее – мы можем, глядя на эти права, сразу понять – с чем наше приложение может работать, а с чем не может вообще.

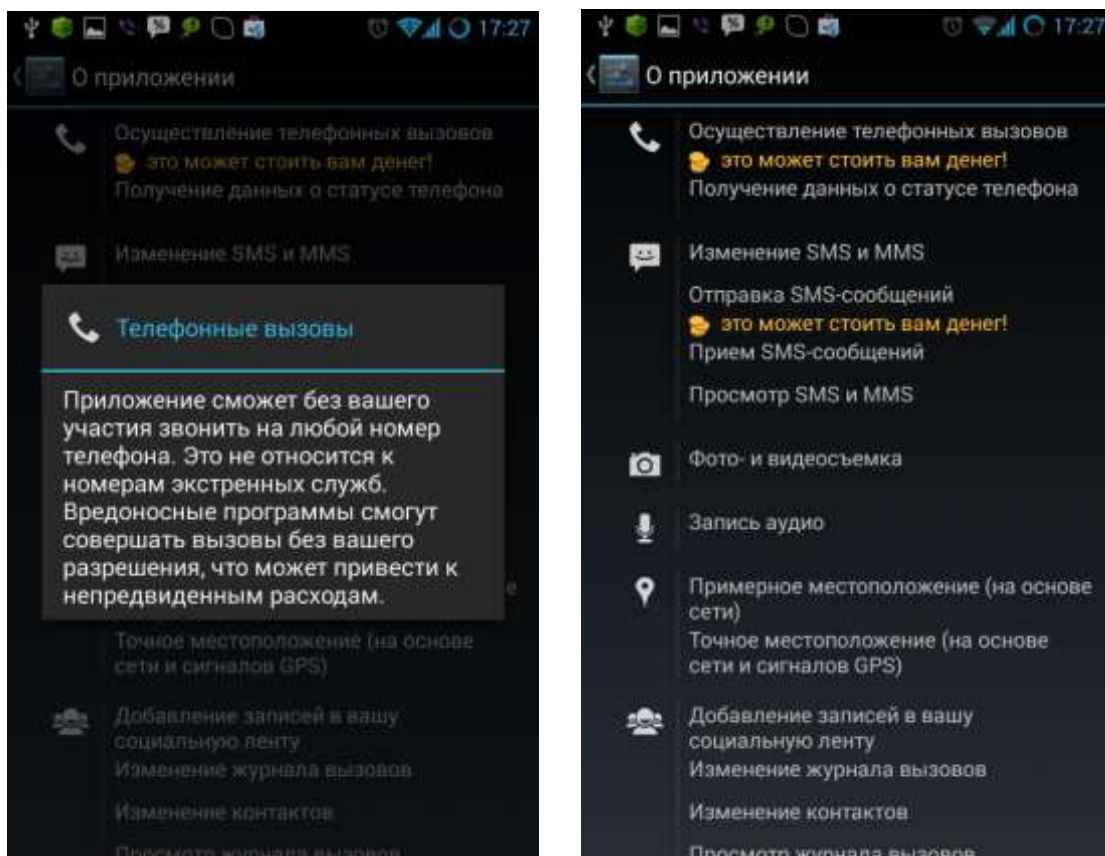
Для наглядности давайте посмотрим свойства мобильной платформы 1С и мессенджера Viber. Для этого зажимаем палец на приложении и ведем его вверх (О программе), или заходим в **Настройки > Приложения > нажимаем на наше приложение.**

*Примечание! Путь может меняться, например, после того, как вы зашли в приложения – необходимо пройти еще одно меню, нажав **Управление приложениями.***

И смотрим внизу список прав. Нажав на любое право, мы узнаем, что оно позволяет делать приложению. Для МП он будет выглядеть вот так:



А теперь посмотрим на права, которые требует Viber:



На что я тут хотел бы обратить внимание – когда говорят, что 1С может звонить или отправлять смс, то это не так. 1С может вызвать окно набора номера или отправки смс, но сама она ни звонить, ни отправлять, ни принимать смс не может! Так как это, как минимум, не разрешит ей сделать Андроид (это можно обойти, но об этом не сейчас).

Так же эти права показываются в момент установки, так что, если вы устанавливаете игру, а она имеет возможно совершать звонки или отправлять смс (желтеньким выделяется), то насторожитесь, может это не такая и простая игра.

Скорость работы мобильной платформы

Так как мобильник – это не компьютер, и мы не можем просто купить его, и, в случае медленной работы, добавить ему оперативки или заменить диск, то тут надо подходить к вопросу с осторожностью.

Программировать под МП необходимо на том устройстве, с которым в последствии будет работать конечный пользователь. Что бы понимать на сколько это серьезно, я написал статью «[Тестирование мобильной платформы 1С](#)», в которой я постарался дать

ответ на один вопрос – на сколько критично влияет оборудование на производительность в МП.

Давайте я приведу несколько цифр:

	Samsung Galaxy Note	Lenovo A706	GSmart G1355	Samsung Galaxy S3	iPad mini	iPhone 4s	ПК	Google Nexus 4
Вариант А	29 524 00:29	37 705 00:37	99 463 01:39	30 771 00:30	22 216 00:22	26 724 00:26	3 740 00:03	33 867 00:33
Вариант Б	216 462 03:36	287 927 04:47	767 806 12:47	261 531 04:21	166 589 02:46	236 590 03:56	28 029 00:28	251 441 04:11
Вариант В	692 722 11:32	789 088 13:09	2 747 151 45:47	611 554 10:11	465 420 07:45	558 062 09:18	88 176 01:28	689 191 11:29
Вариант Г	1 299 707 21:39	1 586 674 26:26		1 236 880 20:36	938 234 15:38	1 141 971 19:01	161 014 02:41	1 526 536 25:26

Перейдя в самую статью, вы можете более детально ознакомиться с тем, что это за варианты, и даже протестировать свои устройства. Скажу лишь одно – от *Варианта А* до *Варианта Г* –увеличивается количество создаваемых элементов, документов и т.д., т.е. в *Варианте А* – самое маленькое количество документов и элементов справочника.

На текущий момент нас интересуют только цифры. Если быть точнее, то разница между устройствами. Почему это так важно? Ответ очевиден – нам нужно понимать на сколько может затянуться синхронизация в том или ином случае.

Как вы могли увидеть, разница между стационарным компьютером и мобильной платформой 1С отличается по скорости в 10 раз, в среднем.

В самом начале, я бы рекомендовал эмулировать ситуацию на вашей фирме, для того, чтобы понять – с чем нам стоит дальше работать, и, как следствие, в каком ключе.

Приведу пример, если у вас в фирме есть порядка 500 номенклатур и в среднем десяток документов, то можно не заморачиваться с разными сложными механизмами обмена, регистрации и т.д., а просто при каждой синхронизации обновлять все с нуля. Это достаточно удобно.

Однако, если у вас 50 000 товаров, 300 000 характеристик, 1 000 000 штрихкодов и т.д., то тут уже надо бы призадуматься над более сложными механизмами синхронизации.

Выбор устройства для работы с мобильной платформой

Эта тема тоже достаточно актуальная, так как необходимо понимать, на что стоит обратить внимание кроме производительности.

Ну, во-первых, если вы планируете использовать GPS, то, ясное дело, он должен быть в устройстве. Это весьма явный вывод, однако, есть и несколько не явных вещей.

Если вы хотите использовать видеокамеру телефона для сканирования штрихкодов (1С вроде как обещали эту функцию), то тут сразу необходимо обратить внимание на то, что она должна быть с автофокусом. Если автофокуса нет, то программа может или вообще не использовать эту функцию, или сканирование одного штрихкода у вас займет порядка 10 минут. Кроме этого всего, у телефона должно быть достаточно встроенной памяти.

Фонарик желательно иметь, если вы планируете сканировать в темных помещениях, ну или даже просто фотографировать.

Если же вам необходимо в конечном итоге получить программу для инвентаризаций, то тут же встанет вопрос о сканировании товара, при чем встанет достаточно остро, и камерой телефона тут не обойтись. А это значит, что надо параллельно с телефоном – купить Bluetooth сканер, ну или если у вас в устройстве есть USB OTG, то HID совместимый проводной сканер.

Конечно же батарея – ее должно хватать на нормальный рабочий день, прогоните устройство пару раз и убедитесь, что заряда хватит.

Если же вы планируете купить ТСД на Андроид, то тут сходу надо понимать, что в эти ТСД встроены сканеры, которые работают по умолчанию только с предустановленными программами, и если вы попытаетесь просканировать что-то в 1С, то у вас может ничего не получиться.

Прежде чем брать такой сканер – убедитесь, что он может сканировать в абсолютно любые формы (например, в смс сообщения, или в текстовый редактор) и после сканирования – отрабатывает функция перевода строки. Это все в будущем Вам облегчит страдания.

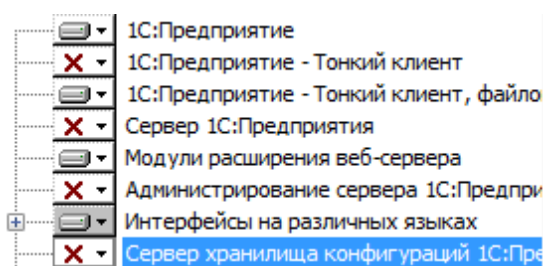
Подготавливаем софт для работы с мобильной платформой

Как вы могли догадаться – для работы с МП нам необходим соответствующий инструментарий.

Давайте создадим некий каталог, куда мы с вами будем скидывать все, что нам понадобится. Путь к этому каталогу должен быть коротким, не содержать пробелов и состоять только из латиницы. У меня он будет таким: **C:\MobilePI**. В нем я создам папку **Install**, куда я буду закидывать все установочные файлы. И у нас будет еще одна папка **Soft**, куда мы будем скидывать подготовленные приложения, устанавливая новые и ссылаясь на этот путь везде, где потребуется.

Настройка платформы 1С

Для начала убедитесь, что у вас установлены следующие модули 1С:



Если чего-то не хватает – установите.

Так как МП по своим свойствам и возможностям похожа на работу в тонком клиенте, то нам нужно установить модуль, позволяющий работать в режиме тонкого клиента в файловом режиме.

Мы, в том числе, будем рассматривать вариант обмена данными при помощи *web-сервисов*, поэтому нам необходим модуль для работы с веб-серверами.

Установка Apache

Так как 1С сама по себе не может выступать в роли веб-сервиса, то нам необходимо настроить или IIS или Apache. Мы будем рассматривать вариант с Apache, так как он является более легким в настройке и кроссплатформенным.

Для начала нам необходимо скачать Apache. Сделать это можно так:

Заходим на сайт <http://www.apache.org/mirrors/>. Это список всех доступных зеркал, на которых можно скачать Apache.

Я выбрал сайт <http://apache-mirror.rbc.ru/pub/apache/>, он для России. Это значения по большому счету не имеет, так как вес его маленький.


Переходя по папкам, мы доберемся вот сюда: <http://apache-mirror.rbc.ru/pub/apache/httpd/binaries/win32/>.

Тут нас интересует файл [httpd-2.2.25-win32-x86-no_ssl.msi](#). Если вы используете SSL, то качайте тот, что ниже. Если не знаете, что это такое – качайте любой.

Во время установки программа спросит про порт, который она будет прослушивать. По умолчанию – это порт 80, его и выбираем. Обратите внимание, что на 80 порту любят висеть всякие программы, которые в последствии не дадут запустить вам Apache. Чаще всего – это:

- *TeamViewer* – он любит слушать 80 порт, чтобы отключить это – перейдите в *Опции > Дополнительно > Показать дополнительные настройки >* и поставьте галочку *Не использовать входящий 80 порт*.
- *Skype* – чтобы отключить, следуем в *Инструменты – Настройки – Дополнительно – Соединения* и меняем порт.
- *ISS* – само собой. Но если вы его ставили, то вы знаете как его отключить или сменить порт.

Я рекомендую потом назначить свой порт. Чтобы убедиться, что выбранный Вами порт, на текущий момент, не использует ни одна программа – зайдите в Диспетчер задач > Производительность – Монитор ресурсов – Сеть – Прослушиваемые порты. Убедитесь, что порт не занят и смело его вбивайте. По опыту, я использую порты в диапазоне 40 000 – 60 000.

После установки у вас в трее должен появиться такой значок , если же там красный квадрат, то, вероятнее всего, что-то уже прослушивает этот порт. Что именно – можно узнать через *Монитор ресурсов*.

Проверить работоспособность Apache можно, забив в браузере адрес *127.0.0.1* или *localhost*, или IP адрес вашего компьютера (при условии того, что открыт доступ к 80 порту). Если все хорошо, то вы увидите надпись **It works!** Если ее нет, то, вероятнее всего, вы или не указали порт, или у вас в *hosts* не прописаны эти адреса.

Установка Apache ant

Если вы пока не планируете компилировать приложение до арк-файла, то можно этот пункт пока пропустить.

Это программа для сборки приложений. Нам она нужна только для сборки конечных арк-файлов.

Скачать ее можно с того же ftp, что и Apache.

Путь будет таким – <http://apache-mirror.rbc.ru/pub/apache/ant/binaries/>. На текущий момент там лежит версия [apache-ant-1.9.2-bin.zip](http://apache-mirror.rbc.ru/pub/apache/ant/binaries/apache-ant-1.9.2-bin.zip). Ее мы и качаем в нашу папку *Install*. Так как это просто архив, который не требует инсталляции, мы его сразу разархивируем в папку *C:\MobilePI\Soft\ant*. Обратите внимание, что это конечный путь, где лежат файлы. Т.е. я не перенес папку *apache-ant-1.9.2-bin*, так как, в случае обновления, вам нужно будет только удалить эти файлы и скопировать сюда новые, при этом все пути сохранятся.

Для чего еще ее можно использовать – можно почитать тут:

- <http://www.quizful.net/post/ant-tutorial>
- <http://www.artlebedev.ru/tools/technogrette/soft/eclipse-ant/>

Эти все ссылки я привел для понимания того, что 1С его использует не спроста. Это мировая тенденция. Ранее, в версии 8.3.3 мобильной платформы был сборщик приложений, работал он на основе жестко прописанных путей и кода. Однако, при помощи ант можно задать скрипт действия, по которому он будет компилироваться. Их даже можно несколько собрать. Например, собирать приложения под разные имена пакетов и разные ключи разработчика.

Установка Java JDK

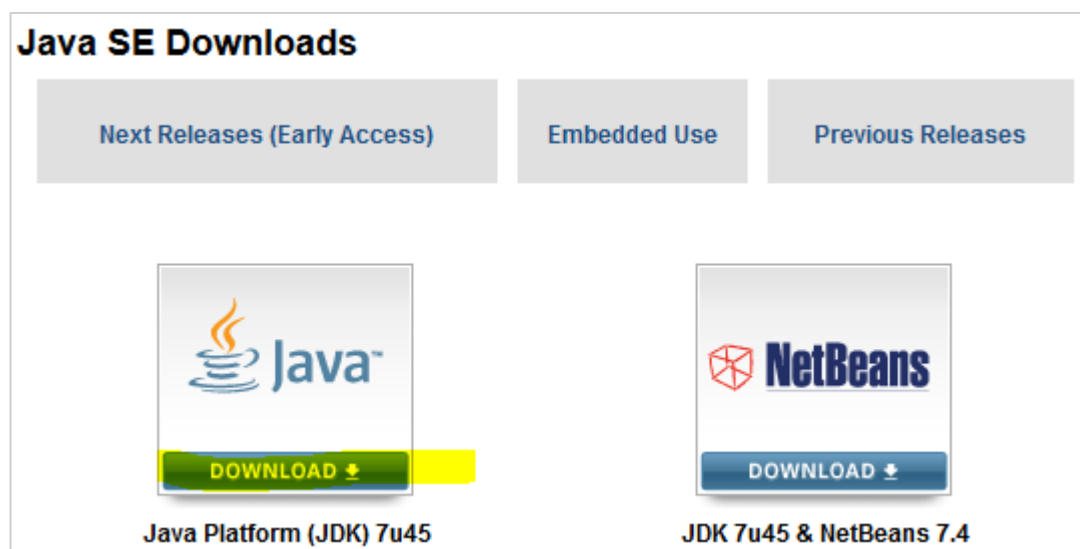
Если вы пока не планируете компилировать приложение до арк-файла, и использовать виртуальную машину, то можно этот пункт пока пропустить.

Так как Android, по сути, эмулирует виртуальную Java машину, то и нам от Java не уйти. Рассматриваемая нами версия мобильной платформы уже нормально работает с Java 1.7, вот ее мы и установим.

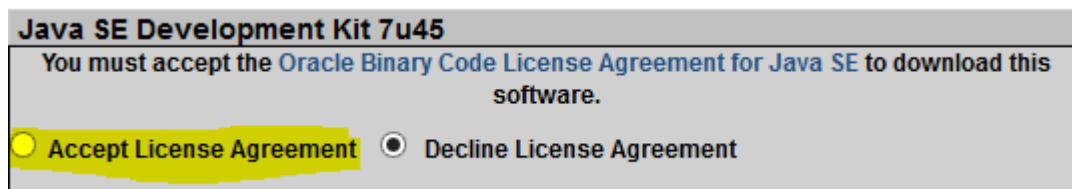
К сожалению, прямого линка на скачку нет, поэтому придётся проходить квест на скачку новой версии. На текущий момент, ссылка вот такая: <http://download.oracle.com/otn-pub/java/jdk/7u45-b18/jdk-7u45-windows-i586.exe>.

Идем на сайт <http://www.oracle.com/technetwork/java/javase/downloads/index.html>.

Видим такую картинку:



Нажимаем на JDK Download (выделил желтым).



Соглашаемся с лицензией и качаем нужную нам версию. Есть x64 и x86, так как мы не работаем с большими объемами, то, в независимости от разрядности вашей операционной системы, качаем x86.

Java SE Development Kit 7u45		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux ARM v6/v7 Hard Float ABI	67.67 MB	jdk-7u45-linux-arm-vfp-hflt.tar.gz
Linux ARM v6/v7 Soft Float ABI	67.68 MB	jdk-7u45-linux-arm-vfp-sflt.tar.gz
Linux x86	115.62 MB	jdk-7u45-linux-i586.rpm
Linux x86	132.9 MB	jdk-7u45-linux-i586.tar.gz
Linux x64	116.91 MB	jdk-7u45-linux-x64.rpm
Linux x64	131.7 MB	jdk-7u45-linux-x64.tar.gz
Mac OS X x64	183.84 MB	jdk-7u45-macosx-x64.dmg
Solaris x86 (SVR4 package)	139.93 MB	jdk-7u45-solaris-i586.tar.Z
Solaris x86	95.02 MB	jdk-7u45-solaris-i586.tar.gz
Solaris x64 (SVR4 package)	24.6 MB	jdk-7u45-solaris-x64.tar.Z
Solaris x64	16.23 MB	jdk-7u45-solaris-x64.tar.gz
Solaris SPARC (SVR4 package)	139.38 MB	jdk-7u45-solaris-sparc.tar.Z
Solaris SPARC	98.17 MB	jdk-7u45-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	23.91 MB	jdk-7u45-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	18.26 MB	jdk-7u45-solaris-sparcv9.tar.gz
Windows x86	123.49 MB	jdk-7u45-windows-i586.exe
Windows x64	125.31 MB	jdk-7u45-windows-x64.exe

Сохраняем его в наш каталог *Install*. Заранее создадим папку для этой программы – *C:\MobilePI\Soft\jdk* и *C:\MobilePI\Soft\jre*. Запускаем программу.

Когда программа спросит – куда установить ее, укажите путь *C:\MobilePI\Soft\jdk*. Потом она спросит еще раз – куда установить *jre*. Указываем путь *C:\MobilePI\Soft\jre*.

На этом установка *Java* закончена.

Установка мобильной платформы для разработчика

Нам необходимо скачать архив с сайта 1С. Я использую платформу [8.3.4.9](#). Переименовываем архив в соответствии с версией, которая в нем находится, и копируем как есть в папку *C:\MobilePI\Install*.

Давайте разберем структуру этого архива. Внутри него есть три папки:

- *Android*, в которой есть 4 файла:

- *1cem-arm.apk* и *1cem-x86.apk* – платформа разработчика.
- *prjandroid-arm* и *prjandroid-x86* – файлы для компиляции.
- iOS, тут файлы для сборки под iPhone;
- *MobileAppMaker* – тут находится конфигурация «Сборщик приложений».

В чем отличие файлов *1cem-arm.apk* и *1cem-x86.apk*? Отличие их заключается в оптимизации под одну или другую архитектуру процессоров. Чаще всего используется архитектура *arm*. Однако, если у вас в телефоне установлен процессор от Intel, то, вероятнее всего, у вас используется архитектура *x86*.

Теперь надо определиться – куда мы хотим установить мобильную платформу? У нас есть два варианта:

- На мобильное устройство. Этот вариант самый идеальный, так как телефон будет работать значительно стабильней, чем компьютер.
- Эмулятор андроида на ПК:
 - Если без дополнительных танцев с бубном, то можно установить эмулятор [BlueStacks](#). Однако придётся пожертвовать стабильностью. Те, кто работали под ним, изредка замечали странное поведение форм и вылеты. Тогда как на телефоне было все отлично. Однако с ним удобно работать – достаточно установить его и нажать два раза на акр-файле, и он сразу установиться в эмулятор.
 - Эмулятор Android Virtual Device (AVD). Он идет в комплекте с Android SDK. Этот эмулятор – самый приближенный к реальности. Однако стоит отметить, что, как и установка его самого, так и приложений под него – задача не из самых приятных.

Установка Android эмулятора AVD и Android SDK

Если вы пока не планируете компилировать приложение до арк-файла, и использовать виртуальную машину, то можно этот пункт пока пропустить.

Для установки AVD, нам потребуется скачать Android SDK, так как нам все это понадобится далее – не удаляйте эти файлы.

Теперь давайте скачаем Android SDK – идем по этой ссылке: <https://developer.android.com/intl/ru/sdk/index.html>.

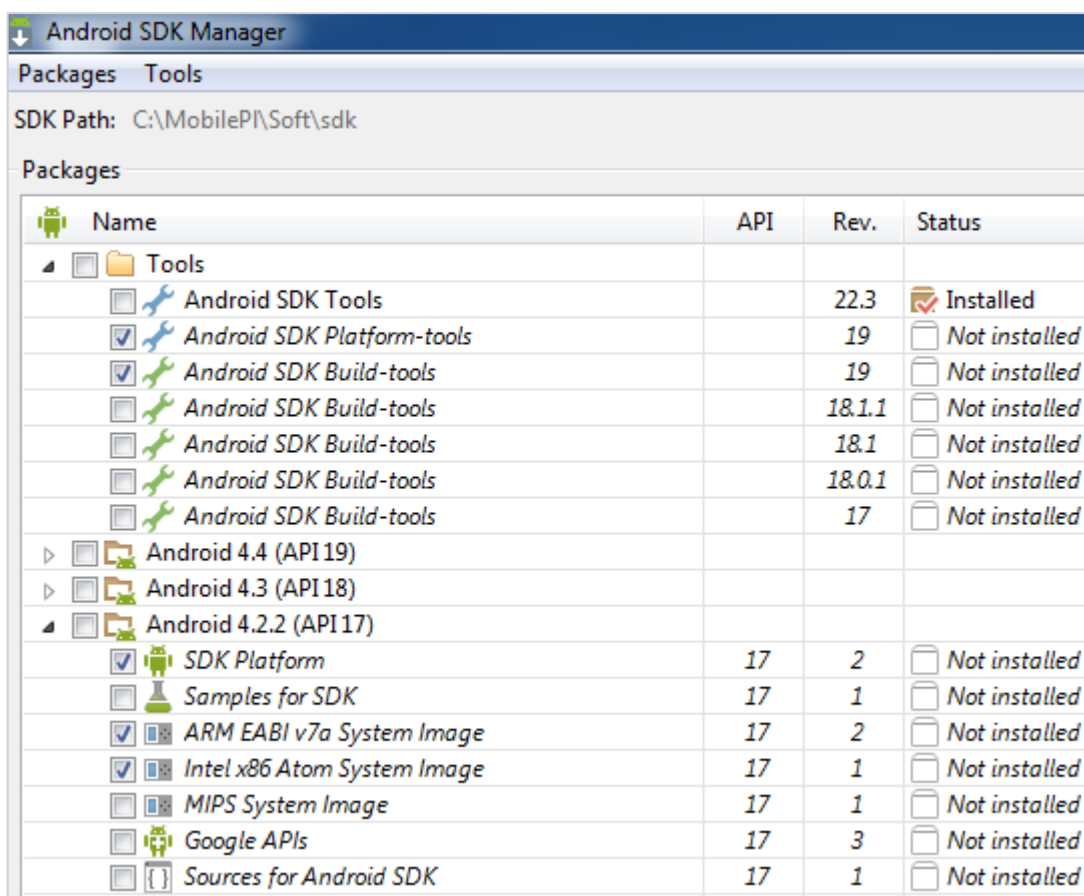
Там раскрываем USE AN EXISTING IDE (так как наша среда разработки 1С, то другой нам не надо), и нажимаем Download the SDK Tools for Windows. Сохраняем его в нашу папку *Install*.

Начинаем установку – запускайте файл. Для установки указываем путь *C:\MobilePI\Soft\sdk*.

Обратите внимание, если вы не установили ранее JDK, то программа не разрешит установку.

После установки программа спросит разрешения запустить себя, соглашаемся. Если же вы убрали галочку, то запускаем файл *C:\MobilePI\Soft\sdk\SDK Manager.exe*.

Ставим нужные нам опции:



Extras			
<input type="checkbox"/>	Android Support Repository	3	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Android Support Library	19	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google AdMob Ads SDK	11	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Analytics App Tracking SDK	3	<input type="checkbox"/> Not installed
<input type="checkbox"/>	[Deprecated] Google Cloud Messaging for Android	3	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play services for Froyo	12	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play services	13	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Repository	4	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play APK Expansion Library	3	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play Billing Library	5	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play Licensing Library	2	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/>	Google USB Driver	8	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/>	Google Web Driver	2	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/>	Intel x86 Emulator Accelerator (HAXM)	3	<input type="checkbox"/> Not installed

Вот тут остановимся немного поподробней. Эмулятор Андроида, сам по себе, весьма тормознут. Однако, есть специальное решение, которое, используя инструкции процессора, ускоряет его, скажем так – прилично ускоряет. Но, как и всегда, есть нюансы – ваш процессор должен соответствовать определенным условиям:

Аппаратное обеспечение

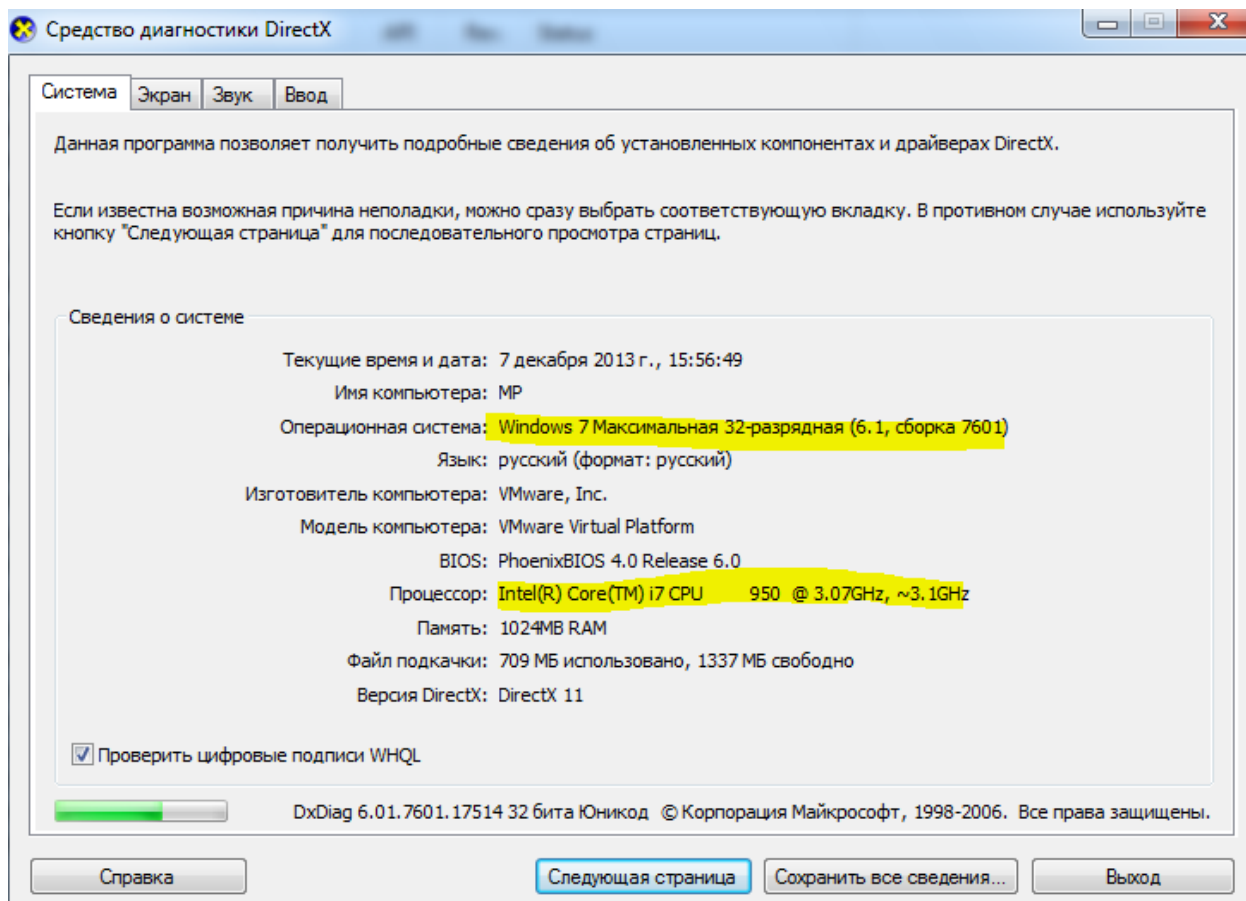
- Процессор Intel® с поддержкой инструкций VT-x, EM64T и Execute Disable (XD) Bit
- Не менее 1 ГБ оперативной памяти

Поддерживаемые операционные системы

- Microsoft Windows 7 (32/64-bit)
- Microsoft Windows Vista (32/64-bit)
- Microsoft Windows XP SP2 (32-bit only)

Примечание: Intel HAXM работает только с теми процессорами Intel, которые в точности соответствуют приведенным выше требованиям. Процессоры других производителей не поддерживаются.

Как можно проверить – подходит ли это все вам? Необходимо узнать модель процессора. Сделать это можно, нажав сочетание клавиш *Win+R*. У вас должно всплыть окно, далее вбиваем в него: *dxdiag*. Появиться вот такое окно (вначале может спросить про подписи – жмем что угодно):



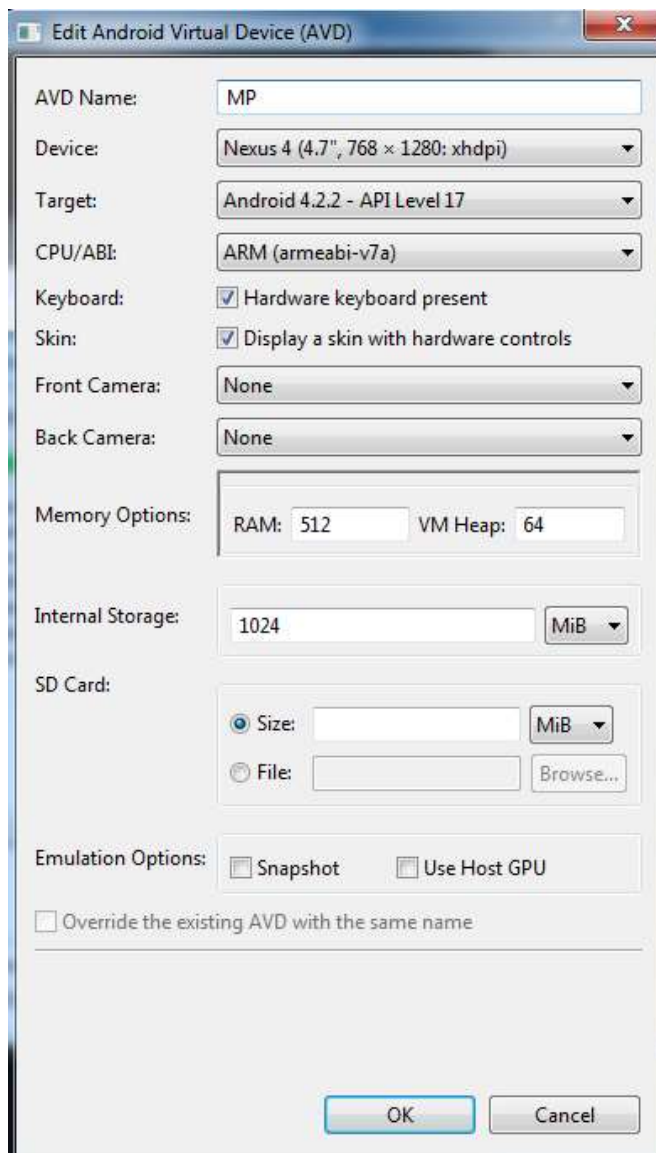
Операционная система – подходит. А вот от процессора нам нужна модель. В моем случае это **i7 950**. Захожу на сайт <http://ark.intel.com/> и справа вверху, в окно поиска, вношу свою модель **i7 950**. Спускаемся вниз и смотрим:

Advanced Technologies	
Intel® Turbo Boost Technology ‡	Yes
Intel® Hyper-Threading Technology ‡	Yes
Intel® Virtualization Technology (VT-x) ‡	Yes
Intel® VT-x with Extended Page Tables (EPT) ‡	Yes
Intel® 64 ‡	Yes
Idle States	Yes
Enhanced Intel SpeedStep® Technology	Yes
Intel® Demand Based Switching	No
Thermal Monitoring Technologies	No
Intel® Data Protection Technology	
AES New Instructions	No
Intel® Platform Protection Technology	
Trusted Execution Technology ‡	No
Execute Disable Bit ‡	Yes

Мой процессор, выпущенный в 2009 году, удовлетворяет условиям. Поэтому я ставлю ту галочку. Дополнительно про это написано на сайте <http://software.intel.com/ru-ru/articles/intel-hardware-accelerated-execution-manager>.

Теперь жмем кнопку *Install*. Далее принимаем лицензионные соглашения и начнется установка. Вся установка будет идти из интернета. Так что учтите это. После установки – закрываем все. И запускаем AVD – эмулятор Андроида, для этого открываем файл `C:\MobilePI\Soft\sdk\AVD Manager.exe`.

Нажимаем *New* и заполняем поля:



Если в двух словах, то мы по сути эмулируем некое устройство. Мы указываем какую версию Андроид использовать, какую архитектуру процессора, оперативную память, память для одной виртуальной java-машины (приложение), внутреннюю память и т.д.

Нажимаем *OK*. И после этого жмем кнопку *Start*. У нас должна запуститься виртуальная машина на Андроиде.

Установка приложения в Андроид

На текущий момент вы должны уже были определиться с устройством, на котором вы будете запускать мобильную платформу. Рассмотрим два варианта установки приложения.

Для примера, возьмем платформу для разработчиков. Напоминаю, она находится в архиве, который мы скачали ранее с сайта 1С, и называется *1cem-arm.apk* и *1cem-x86.apk*. Так как у меня устройство на arm-архитектуре, то я буду устанавливать файл *1cem-arm.apk*.

Установка приложения на телефоне

Для установки – скопируйте его на телефон, и запустите. Перед этим убедитесь, что у вас разрешено устанавливать приложение не только с доверенных источников. Эта галочка ставится в *Настройки > Безопасность*. Для передачи файлов на телефон я использую программу *DropBox*. Т.е. я устанавливаю ее на компьютер и на телефон, потом просто скидываю в папку нужный мне файл и забираю его с телефона. Этот подход требует установки дополнительного софта, и выхода в интернет.

Установка приложения на AVD

Тут все немного сложнее. Интерактивного установщика приложений нет. Поэтому придётся устанавливать через командную строку. Но чтобы не мучатся сильно, мы сделаем *cmd*-файл.

Для этого создадим новый текстовый документ и нажмем *Файл > Сохранить как > имя файла – «Установка apk.cmd» > Тип файла – «Все файлы»* и жмем *Сохранить*. Далее вставим в него такой текст:

```
C:\MobilePI\Soft\sdk\platform-tools\adb.exe install
```

```
C:\MobilePI\Soft\8.3.4.9\Android\1cem-arm.apk
```

Сохраним и запустим. После того, как выполнится этот скрипт – будет установлена мобильная платформа.

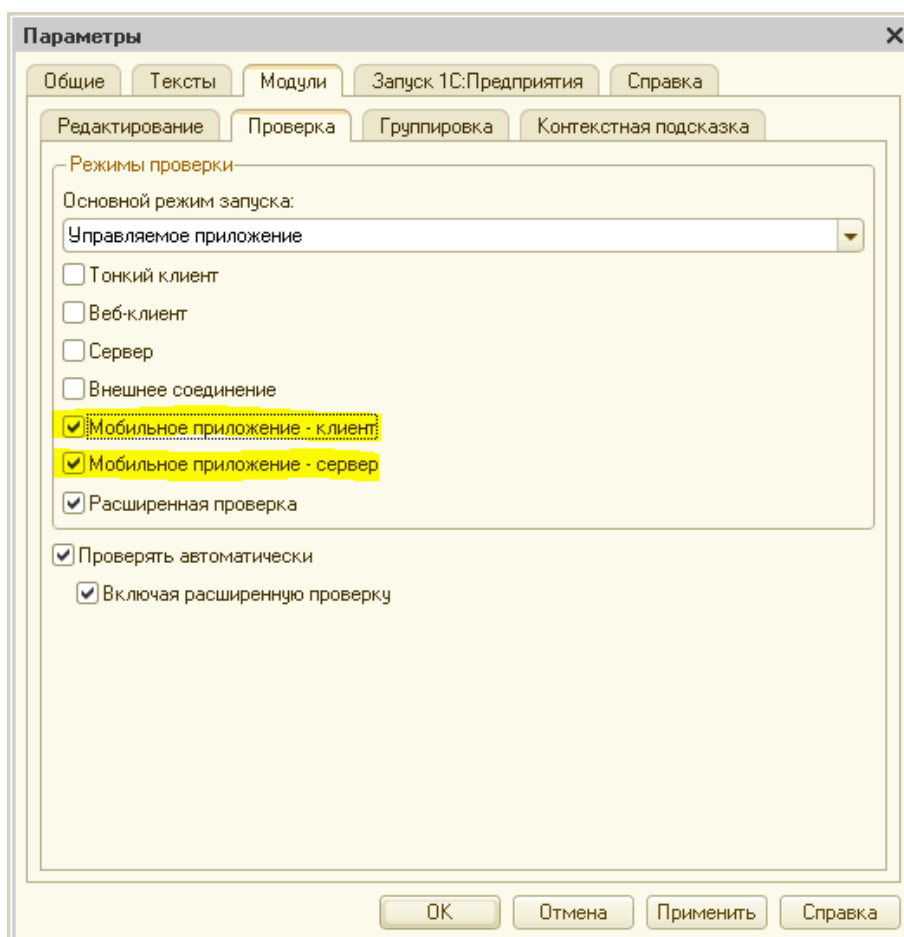
По умолчанию в AVD стоит андроид с локализацией для Америки. Поэтому 1С будет там на английском языке. Для работы с русской версией – зайдите в настройки локализации и выберите нужную вам локализацию.

Пишем и переносим простейшую конфигурацию на мобильное устройство

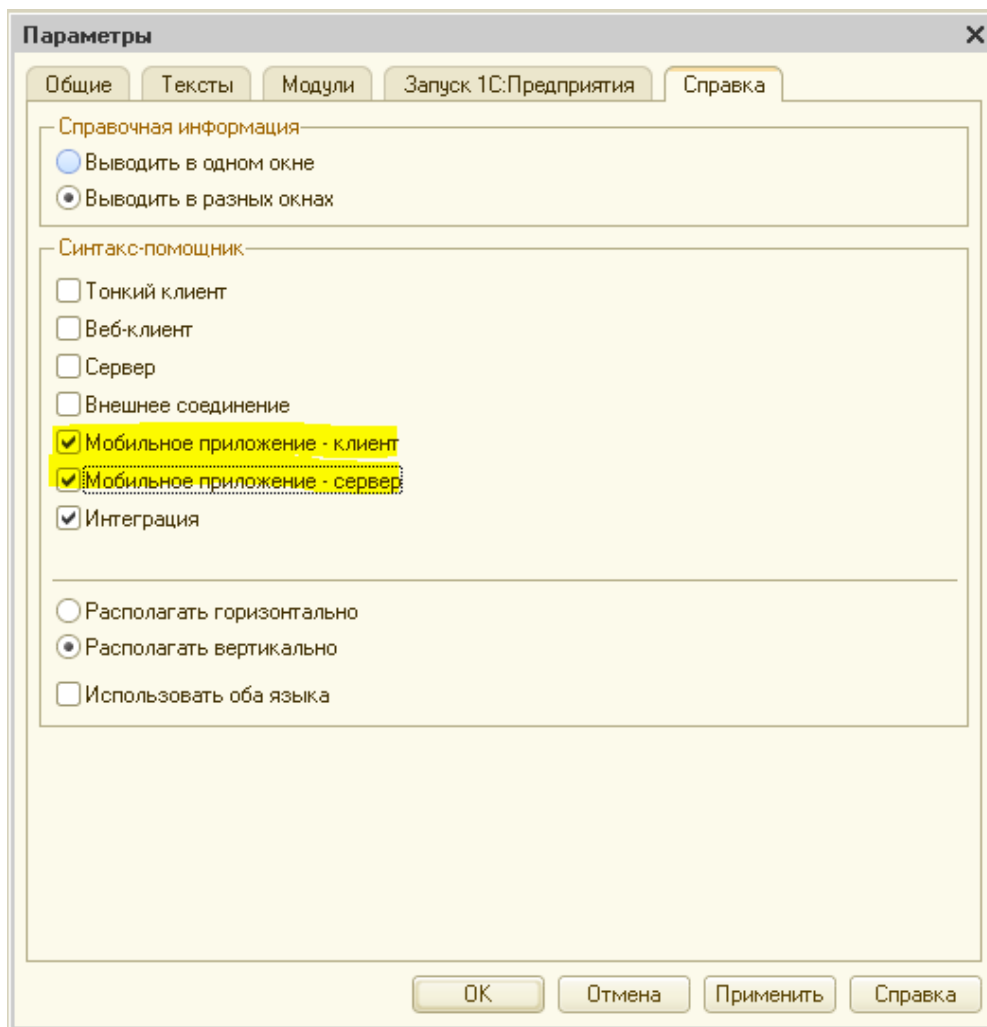
Прежде чем мы приступим к написанию конфигурации, я хотел бы рассказать об ограничениях МП. МП поддерживает не все виды метаданных, которые доступны на стационарном компьютере, но, кроме этого, она еще поддерживает не все общие функции.

Так как целью данного параграфа будет написание конфигурации только под мобильную платформу, то я предлагаю убрать пару галочек из настроек конфигуратора. А именно, мы с вами уберем из контекстной подсказки те функции, которые не поддерживает МП, и из справки – ту информацию, которая не доступна на МП.

Начнем. Давайте создадим для начала новую базу данных. Тут нет ничего необычного, создаем как обычную базу данных. Входим в конфигуратор и первое что мы делаем – заходим в *Сервис > Параметры*:



Тут снимаем галочки напротив тех опций, которые нам будут только мешать.

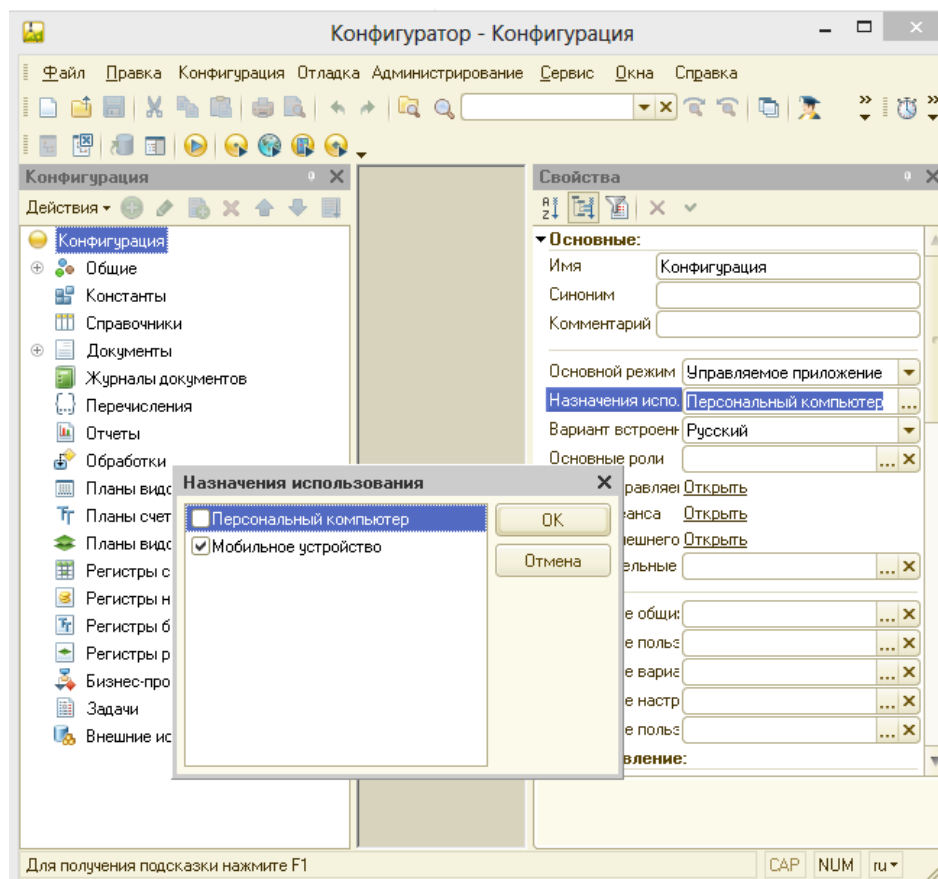


Тут снимаем галочки с тех опций, которые мы не хотим видеть в справке.

Зачем это надо? Все очень просто – если вы оставите эти галочки, то вы всюду будете наткаться на функции, которые не поддерживаются мобильной платформой и наоборот, если вы используете те функции, которые поддерживает только МП – конфигуратор будет ругаться, что их не поддерживают остальные.

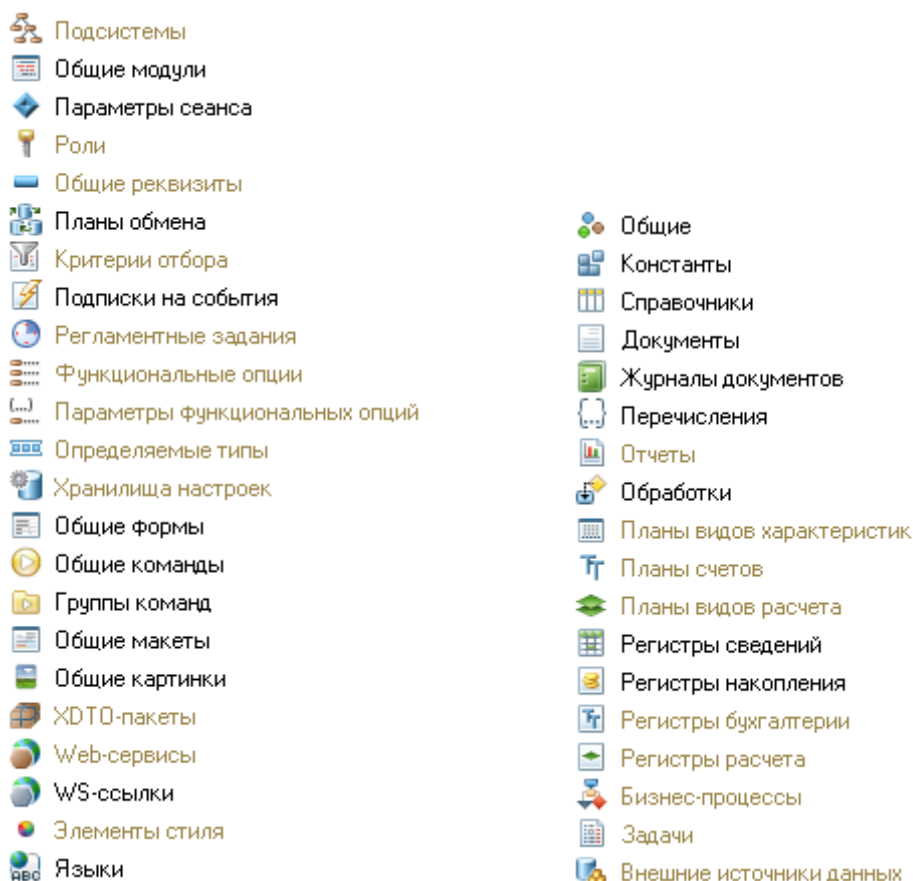
После этого нам надо указать конфигуратору, что данная конфигурация разрабатывается только для МП.

Правой кнопкой на дереве конфигурации > *Свойства* > *Назначение использования* > Ставим галочку *Мобильное устройство*, и снимаем галочку с *Персональный компьютер*.



После нажатия **OK** – дерево метаданных изменит свой вид. В нем появятся затемненные объекты. Это объекты, с которыми МП работать не может.

Выглядит это так:



Тут четко видно объекты, которые не доступны. Среди этих объектов, есть объекты, программирование на мобильные устройства без которых является весьма нелегкой задачей. По моему мнению это:

1. Конечно же **Отчеты!**

Отчеты – это одна из самых главных функций, которые просто обязаны быть доступны на мобильнике. Я даже с ребятами как-то написал приложение, которое эмулировало СКД конструктор (1С + Android (конструктор отчетов) v1.5.1). Однако тут доступны Обработки, с помощью которых можно выводить отчет. Так же 1С, на примере *демо конфигурации Заказы*, подключается к *web-сервису* на сервере, передает параметры, на сервере строится отчет, сериализуется, сжимается и отправляется на телефон, телефон его десериализует и выводит пользователю. Минусы очевидны – только в режиме онлайн. Но и плюсы есть – можно любой сложности отчет строить, вот только поместиться ли он в 5 дюймов и ограничение связанные с использованием SOAP протокола?

Однако этот же механизм использовали и мы, для синхронизации нашего приложения, поэтому я вам позже расскажу про подводные камни такого подхода конкретней.

2. **Общие реквизиты.** Ну тут скорее желание удобства. Просто в случае программирования конфигурации с нуля, начинаешь понимать, как они полезны.

3. **Регламенты,** ну тут и так все ясно, и этого хотят все, например, запускать в фоне службу 1С, которая синхронизирует задачи, остатки и т.д.

4. **Функциональные опции.** Так как ролей нету, да они тут и не нужны, но в одном предприятии может быть несколько вариантов мобильного приложения, раз нельзя сделать видимость по Ролям, то хотя бы по опциям делать, дабы не держать для каждого свою отдельную конфигурацию. Но следует отметить, что мы несколько конфигураций можем уместить в одно приложение, как это сделать – мы обсудим позже.

Ну и самое печальное, на текущий момент – это то, что нет печати. Печать на мобильных устройствах – это весьма нетривиальная задача, однако мы даже не можем сохранить, например, в DgorBox документ в экселе, что бы потом распечатать. Т.е. это все придется самому делать.

Как один из предложенных вариантов был – вы делаете запрос на сервер, сервер принимает документ, сохраняет его в эксель и отправляет менеджеру на почту, но это как бы не правильно, так как опять-таки – нужен сервер и интернет. Можно конечно генерировать этот же эксель, и передать его в ответ на запрос, а потом сохранить на СД карту, но это уже дело такое, так как вопрос печати с мобильного все же остается.

С другой стороны, все эти ограничения принесли весьма положительный опыт, так как первопроходцы не решались на телефон отправлять 100 000 товаров и 25 типов цен, потому что работать с ними, с учетом текущих ограничений, было бы весьма сложно.

Самое главное – запомните один девиз:

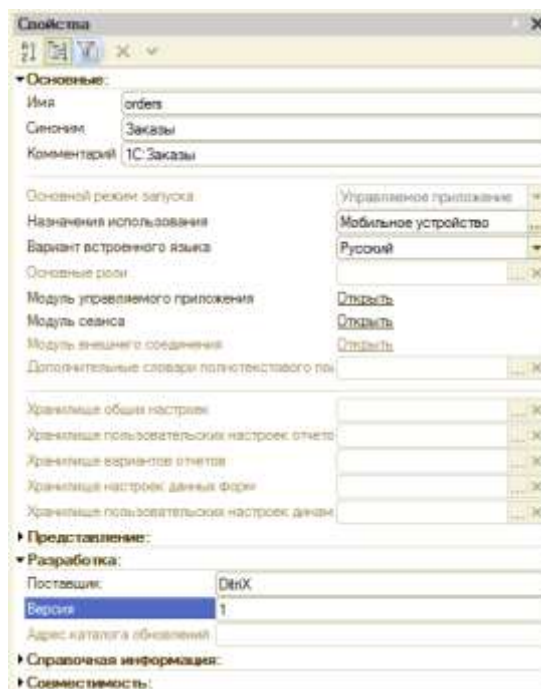
Самое сложное при программировании под мобильную платформу – понять, что это МОБИЛЬНАЯ платформа.

Тут надо четко понимать, что мобильное приложение – это не обычная версия 1С. Так не получится вывести 20 кнопок и 30 показателей.

Там не надо вести партионный учет, там должен быть минимум информации, с которой пользователь может комфортно работать. Каждый лишний реквизит в номенклатуре – добавляет вес приложению и свободу действий оператору. Давайте условимся, что если менеджеру надо видеть себестоимость товара по партиям, то он просто каждое утро будет получать срез этих данных, ну, а если они ему нужны постоянно, то он будет работать в режиме онлайн, иначе это будет костыль на костыле.

После того, как мы определились с ограничениями и уложили это все у себя в голове – приступим к настройке программы.

Основные свойства, которые надо заполнить у конфигурации



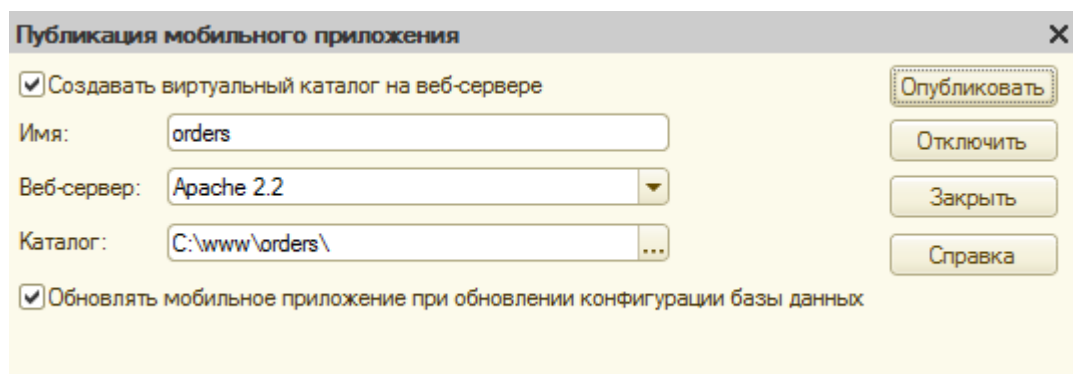
Возвращаемся в нашу конфигурацию и до заполним свойства:

- *Имя* – его желательно указывать латиницей, дальше я объясню почему;
- *Синоним* – тут пишем полное название базы, она так и будет называться и на мобильном телефоне или в виде отдельного приложения, или в составе конфигураций;
- *Поставщик* – укажите себя;
- *Версия* – начинаем с версии 1;

Платформа для разработчика

Данный вариант предполагает наличие доступа к компьютеру из мобильного приложения через сеть. Суть заключается в следующем – мы публикуем конфигурацию приложения (это по сути своей web-сервис), мобильное приложение подключается к нему и забирает конфигурацию.

Давайте опубликуем конфигурацию для того, чтобы получить ее на телефоне. Для этого заходим в *Конфигурация – Мобильное Приложение – Опубликовать* – заполняем данные – *Опубликовать*.



Имя и путь к каталогу – не могут содержать кириллицу!

Давайте чуть поподробней остановимся на некоторых полях.

Имя – не пытайтесь в него вложить много смысла, оно должно быть простое, так как вы не раз его будете вбивать вручную. Имя должно отличаться от имени публикации базы для тонкого клиента.

Обновлять мобильное приложение – эта галочка отвечает за очень удобную опцию. Ее имеет смысл использовать только в произвольной конфигурации, суть ее в том, что при каждом обновлении конфигурации автоматически будет обновлена конфигурация на устройстве (если на нем тоже стоит галочка, об этом позже). Кроме этого, обновлять можно и вручную.

Если, в момент нажатия кнопки Опубликовать, у вас появилось сообщение о том, что 1С не может записать данные в файл, то это у вас сработал UAC. Это значит, что вы запустили 1С не от имени администратора. Решить эту проблему можно следующим путем:

Каждый раз запускать 1С от имени администратора. Правой кнопкой на ярлыке 1С > Запустить от имени администратора.

Отключить UAC (не безопасно!).

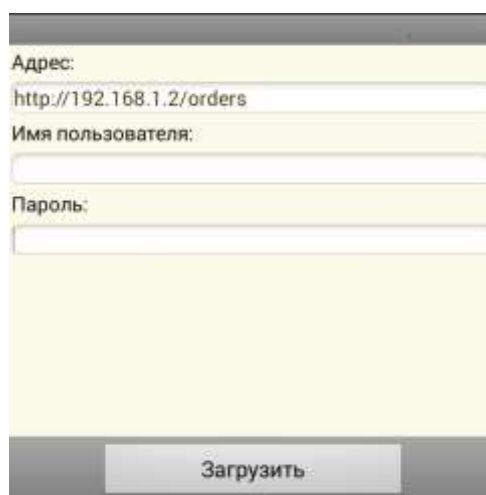
Зайти в свойства ярлыка 1С, и поставить галочку Выполнять эту программу от имени администратора.

У каждого из этих вариантов есть свои плюсы и минусы. Какой подойдет вам – решайте сами.

Если у вас все получилось, то давайте убедимся в том, что наша конфигурация была правильно опубликована. Для этого перейдем по ссылке <http://127.0.0.1/orders/>. Мы должны увидеть html-файл. Это и есть структура нашей конфигурации.

Теперь заходим в ранее установленное приложение, и нажимаем кнопку *Добавить*.

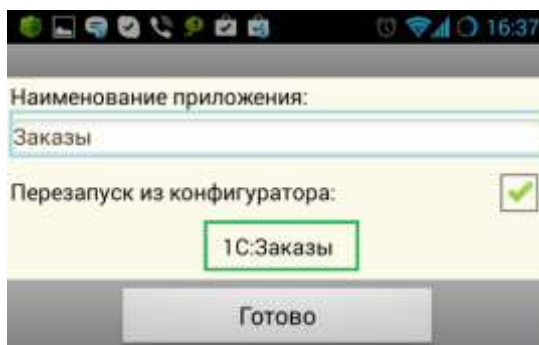
Видим вот такую картинку:



В поле адрес вбиваем IP компьютера, на котором опубликована конфигурация. Если надо, то можно указать другой порт, например, <http://192.168.1.2:54215/orders>.

Учтите, просто вбив адрес без <http://> у вас может ничего не сработать. Как узнать IP адрес вашего компьютера, и что должен быть открыт нужный порт и т.д. – я тут рассказывать не буду.

Далее жмем кнопку *Загрузить*.



Увидим вот такую картинку. Давайте остановимся не на долго. Помните мы заполняли свойства конфигурации? Ну вот тут можете сопоставить одно к другому и использовать в своих целях.

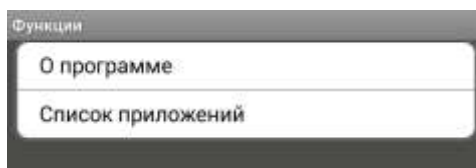
Обязательно поставьте галочку *Перезапуск из конфигуратора*.

Нажимаем *Готово*. Теперь откроется окно выбора конфигурации. Если зажать на конфигурации палец и из всплывающего меню выбрать *Изменить*, то станет доступным такое окно:

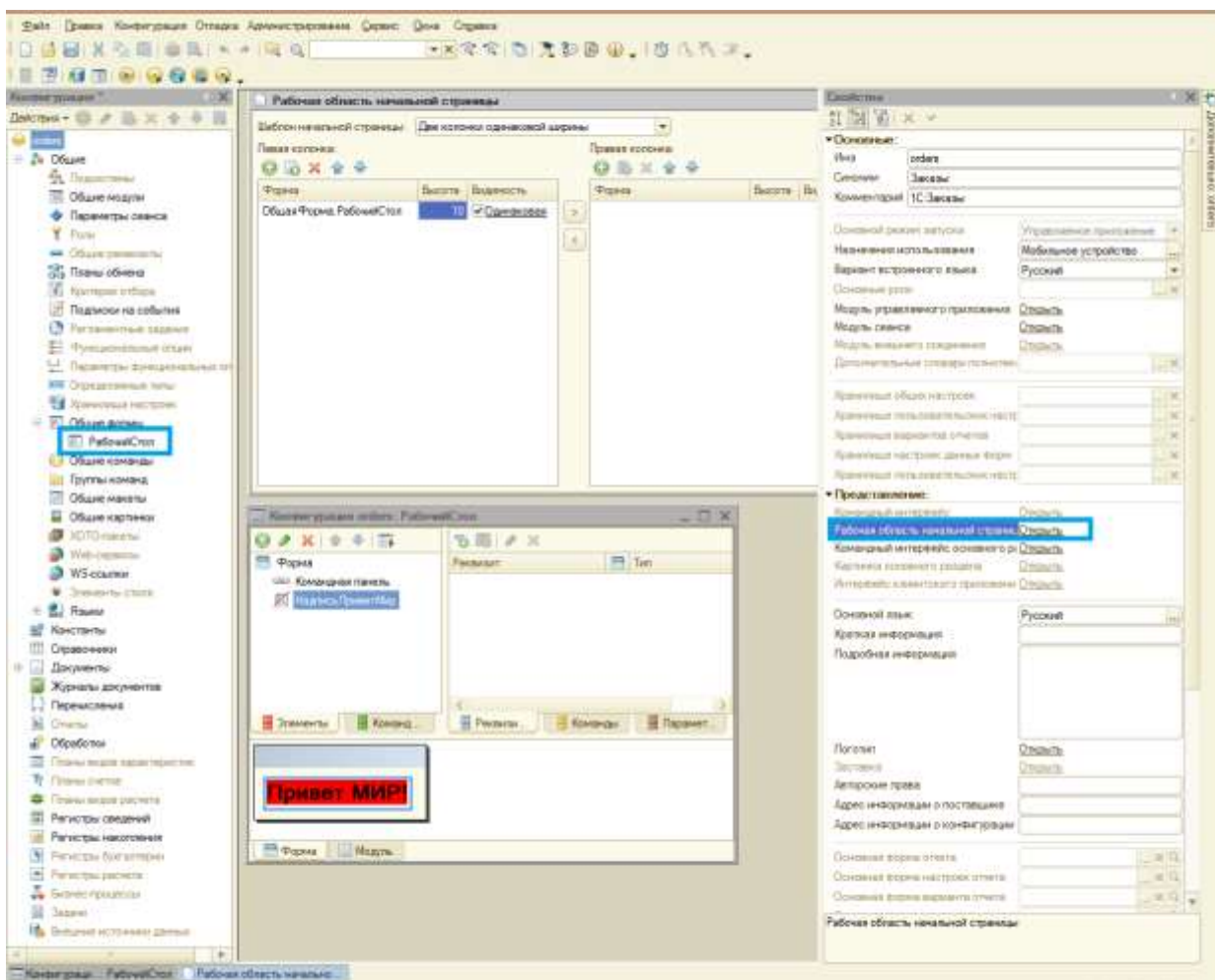


Т.е. вы все можете изменить со временем. Кроме этого, если не стоит галочка *Перезапуск из конфигуратора*, то вам придется постоянно выходить и заходить в конфигурацию самостоятельно, чтобы применились изменения.

Давайте зайдём в конфигурацию, и посмотрим, что у нас получилось.



Так как мы изначально не определили никакой формы для рабочего стола, то программа нам просто выводит меню функций. Давайте исправим это. Добавим новую форму в конфигурацию и выведем ее на рабочий стол.

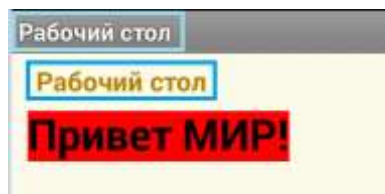


Для начала создадим общую форму, сделаем надпись, вспомним что мы можем использовать форматированную строку, потом в свойствах конфигурации открываем *Рабочую область начальной страницы* и добавляем туда эту форму.

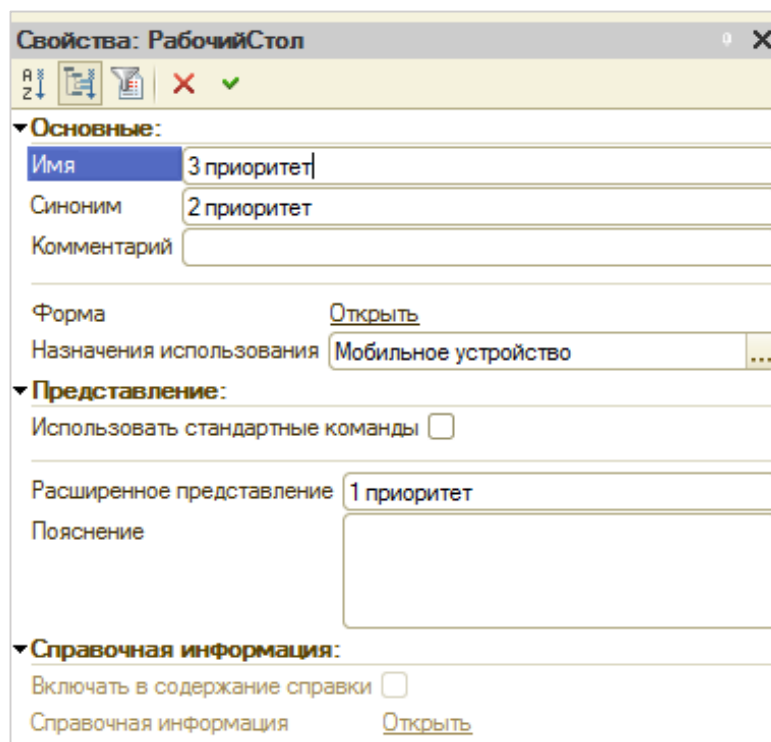
А теперь нажмем кнопку *Обновить*. И смотрим на телефон – приложение могло обновиться, а могло и нет. (*почему? Раньше же обновляло.*) Если не обновилось, тогда выводим кнопочку:



И нажимаем на нее, смотрим на открытую конфигурацию в телефоне. Через пару секунд, телефон должен перезапустить конфигурацию, и мы увидим вот такую форму:



Я специально выделил заголовки в рамку, и хочу сейчас вернуться в свойство формы:



В Андроиде есть разграничение между заголовком основной программы и заголовками формы которые показывает программа. 1С тоже так сделали, но, тут видать ошибка, потому что эти два заголовка идентичные, и ни один из них скрыть, для освобождения пространства – вы не можете.

Заголовок берется из свойств формы согласно приоритетам как я расписал в картинке, т.е. вначале программа берет Расширенное представление и т.д. Поведение идентично стационарной платформе, однако иногда про это забывают.

Для того, чтобы попрактиковаться – предлагаю скопировать нашу тестовую базу и переименовать ее в *warehouse* (далее по статье поймете почему). И в каждой базе – изменим надпись с «Привет МИР!» на «Заказы» и «Склад». Обязательно подключите эту базу в платформу для разработчиков и попробуйте поочередно обновлять базы, зайдите и выйдите. Цель – понять, как работают обновления базы с компьютера.

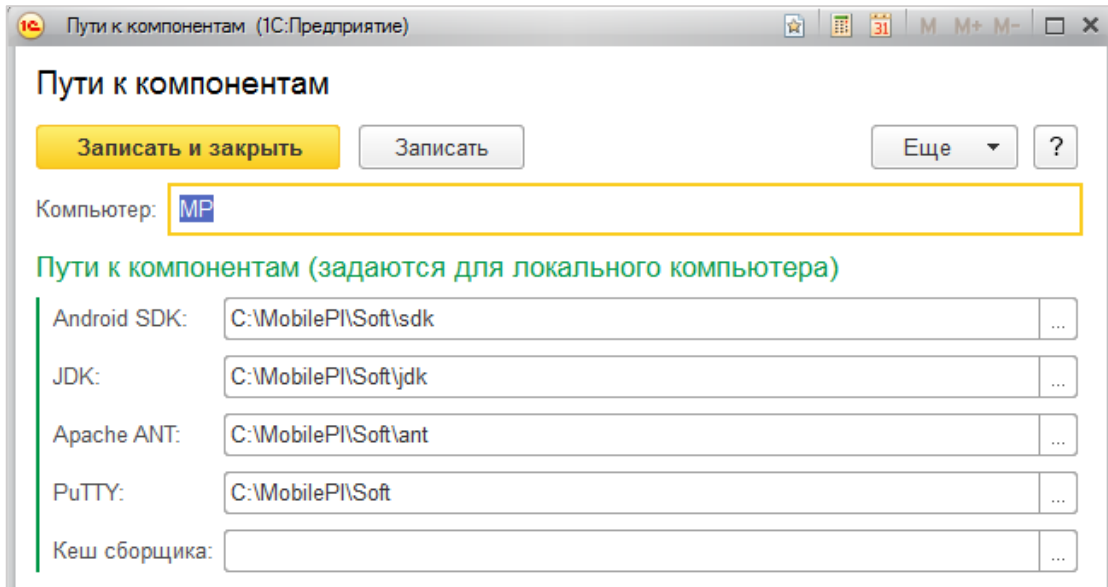
Поздравляю – теперь вы можете начать разработку своего собственного приложения или просто присмотреться к мобильной платформе.

Конфигурация для компиляции приложения

Помните в архиве, скаченном с сайта 1С, у нас была конфигурация? Давайте теперь разберемся с ней. Для этого нам понадобятся обе базы. Начнем с того, что после установки вам надо вначале зайти в конфигуратор и создать пользователя с админскими правами. После этого – заходим в базу.

Сразу заходим в *Сервис > Настройки* приложения и заполняем все, что нам надо. Что и зачем – написано в справке, основная идея в том, что если работаете вы с разных компьютеров, то надо что бы на каждом из них были установлены все пакеты и прописаны пути.

1С, кроме прочего – требует еще утилиту PuTTY, она нужна при компиляции под iOS. К сожалению, обойти это не получается, т.е. разработчики уже сразу считают, что вы будете компилировать под iOS. Поэтому я просто выбираю папку *Soft* и все.



Тут в наименование берется имя компьютера, поэтому не меняйте его.

Далее идем в *Сервис > Редактирование поставщика*.

Придумываем себе префикс. Это очень важный шаг. Андроид устроен так, что он контролирует версию сборки и имя пакета. Т.е. если вы переименуете пакет, то обновить существующую базу – у вас не выйдет. Поэтому давайте придумаем себе префикс. Обычно имя пакета в андроиде выглядит вот так:

com.example.helloworld.

Где **com** – обычно либо окончание сайта, либо просто локаль автора (например – ru, ua и т.д.).

Example – либо продолжение сайта, либо ваше ФИО. Т.е. по сути, нечто уникальное, которое идентифицирует вас.

Helloworld – имя приложения. В нашем случае это может быть orders.

Я выбрал себе такое имя пакета: **ua.dnv + имя приложения**

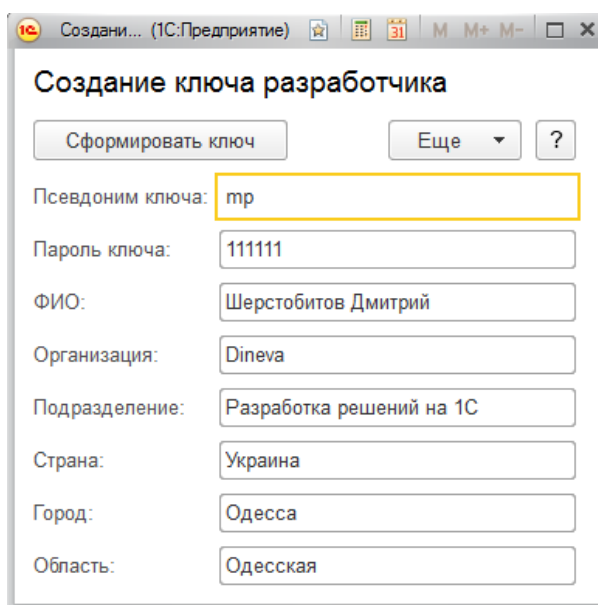
Внимание! Имя должно содержать только латиницу и цифры, ну и часть спец символов. Каждая часть названия – подчиняется правил объявления переменной, т.е. нельзя называть вот так rfë.1cd.dd#

Далее нам необходимо создать ключ поставщика. Это очень важный шаг. Например, если андроид нашел приложение с таким же именем пакета, то он проверяет подпись –

если подпись не совпадает, то андроид не обновит это приложение. Соответственно, если вы потеряете ключ, то обновить существующие приложения у клиентов не выйдет.

А если вы распространяете свое приложение про помощи площадки Google Play, то вы не сможете обновить там приложение. Придется создавать новое при этом теряя все скачки, голоса и т.д.

Итак, нажимаем на ссылку *Создать ключ разработчика*. Все поля, кроме пароля и псевдонима, можно заполнять на каком угодно языке.



Я заполнил вот так. Жмем теперь *Сформировать ключ*. Дальше программа заполнит все сама.

Поставщик мобильных решений (Поставщики) (1С:Предприятие)

Поставщик мобильных решений (Поставщики)

Главное Сертификаты на компьютерах Apple

Записать и закрыть Записать

Еще ?

Код: 00000001

Наименование: Поставщик мобильных решений

Префикс идентификатора приложения: ua.dnv

Новые прикладные решения данного поставщика собирать

Для ОС iOS

Для ОС Android

Параметры ключа разработчика для ОС Android

Пароль для ключа разработчика: *****

Псевдоним ключа разработчика для ОС Android: mp

Хеш SHA1 ключа разработчика: D3:B7:F1:EA:9D:87:BE:5E:9E:90:E2:9A:A8:E5:36:78:03:DD:4C:70

[Установить ключ разработчика](#) [Создать ключ разработчика](#) [Получить хеш ключа разработчика](#)

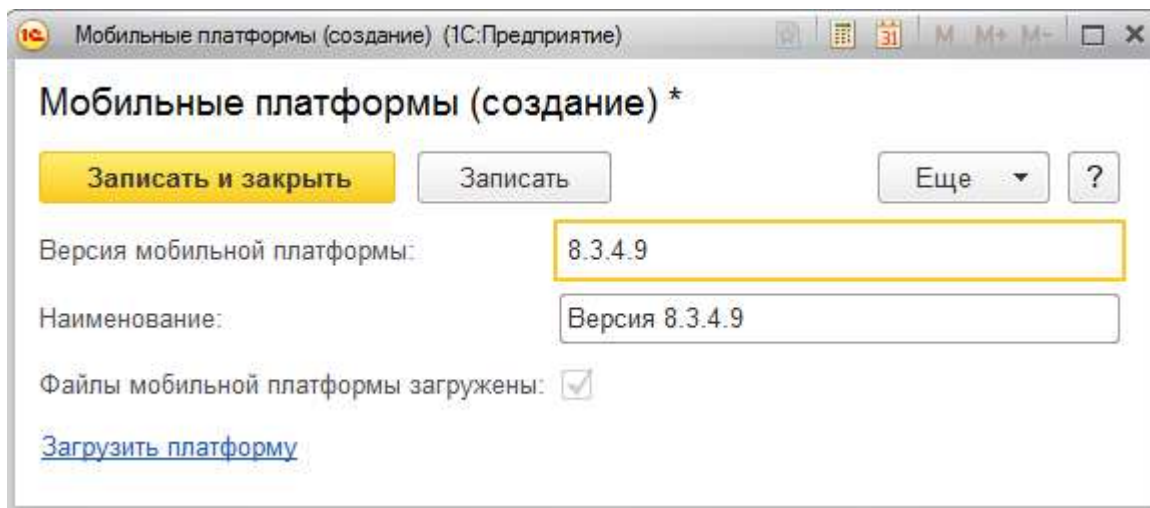
Ставим галочку “Для ОС Android” и записываем элемент.

Далее нам надо подключить архив мобильной платформы, чтобы программа могла скомпилировать все. Для этого сверху откроем справочник *Мобильные платформы*. Тут надо обратить внимание на такой момент – вся эта информация будет храниться в базе, т.е. если мы каждую платформу будем вставлять в базу в полном объеме – у нас очень быстро вырастит база, а нам это не надо. Поэтому мы немного урезаем архив. Из справки:

При загрузке система ожидает, что мобильная платформа будет представлена zip-файлом со следующей структурой (регистр имени файла и каталога – важен!), остальные файлы в архиве игнорируются:

- *Android\prjandroid-arm.zip* – мобильная платформа для ОС Android на платформе ARM
- *Android\prjandroid-x86.zip* – мобильная платформа для ОС Android на платформе x86
- *iOS\prjios.zip* – мобильная платформа для iOS (мульти язычная)
- *iOS\prjios_en_ru.zip* – мобильная платформа для iOS (только русский и английский языки)

Так как я не компилирую под iOS, но выбора у меня нет, то мне придется и те файлы оставить. Убедимся, что при входе в архив – у нас вначале появляется папка Android и iOS, а в них уже те файлы, что надо. Т.е. в архиве должна быть только 1 папка.

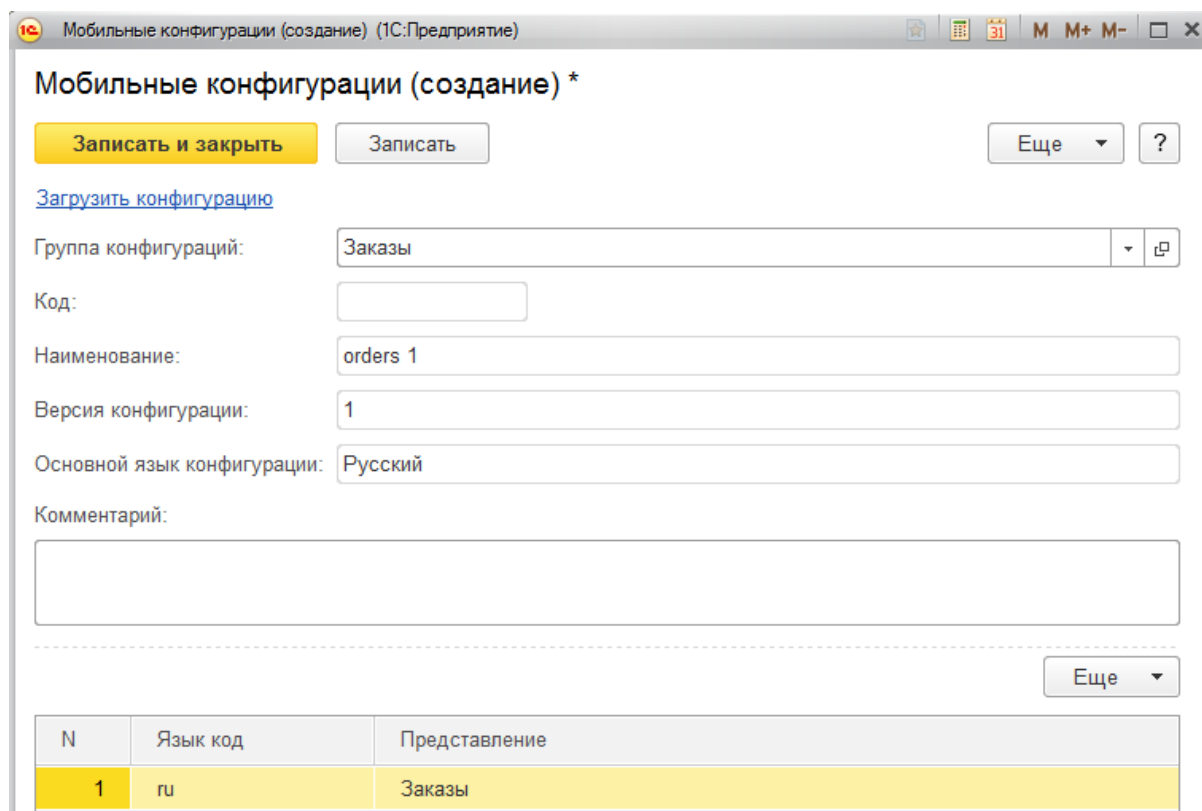


Пишем версию мобильной платформы, причем до последней цифры. Иначе потом запутаетесь в сборках.

Если бы сюда еще информация о найденных и исправленных багов добавлялась – этой конфигурации бы цены не было :).

Далее нам необходимо вставить конфигурации, разработанные ранее. Для этого мы заходим в конфигуратор каждой базы и нажимаем *Конфигурация > Мобильное приложение > Записать в файл*.

Назовем файл как-то по-человечески, например, *orders* и *warehouse* соответственно. Это *xml*-представление конфигурации. В конфигурации для сборки открываем справочник *Мобильные конфигурации* и для каждой конфигурации создаем независимые группы первого уровня – *Заказы* и *Склад*. Заходим в группу и добавляем элемент. Слева вверху нажимаем *Загрузить конфигурацию* и выбираем соответствующий файл.



N	Язык код	Представление
1	ru	Заказы

Далее программа подхватывает имя конфигурации и ее версию. Тоже самое сделаем для конфигурации *Склад*.

А теперь пора бы уже создать и приложение, но придётся заполнить еще одну форму. Для этого открываем справочник *Мобильные приложения* и создаем группу, без группы нельзя. В группе мы указываем все, что нам может пригодиться.

The screenshot shows a web-based interface for creating a mobile application order. The title is "Заказы (Мобильные приложения) *". There are buttons for "Записать и закрыть" (highlighted in yellow), "Записать", and "Еще".

Fields include:

- Код: 00000001
- Наименование: Заказы
- Поставщик: Поставщик мобильных решений

Section: Мобильное приложение собирать

- Для ОС iOS:
- Для ОС Android:
- Мобильная платформа: Версия 8.3.4.9
- Графические ресурсы: (empty)
- Идентификатор решения: orders | ua.dmv.orders (with a "Проверить уникальность идентификатора" link)
- Сертификат для сборки на компьютере Apple: (empty)

Section: Ключ для карт Google

- Параметр получения ключа для работы с картами Google: D3:B7:F1:EA:9D:87:BE:5E:9E:90:E2:9A:A8:E5:36:78:03:DD:4C:70;ua.dmv.orders
- Ключ для работы с картами Google: (empty)

Version and Build fields:

- Версия приложения: 1
- Номер сборки: 1 (highlighted in yellow)

Buttons: "Добавить" and "Еще".

N	Источник
1	Заказы

Называем группу, в нашем случае – это *Заказы*.

Ставим галочку *Для ОС Android*.

Выбираем версию мобильной платформы, на которой будем компилировать.

Пишем идентификатор решений – *orders*. Справа покажется полный идентификатор приложения, еще правее есть кнопка *Проверить уникальность идентификатора*. Она проверяет идентификаторы в текущей базе. Идеально было бы, что бы программа проверяла уникальность на площадке, где вы собираетесь разместить приложение в будущем. Но пока и так сойдет.

Указываем версию приложения и номер сборки.

Тут остановимся подробнее. По номеру сборки – программа Android проверяет версию программы, т.е. если у вас установлена программа с номером сборки 22, то программу с номером сборки 20 вы уже не поставите. Этот номер не имеет дробной части, это целое число. На моей практике – доходило до нескольких тысяч. Это не значит, что если вы подняли номер сборки – то «обязаны» опубликовать приложение.

А версия приложения – это представления для пользователя. Как все любят, что-то вроде 1.21.44.

И внизу – вы добавляете папку с конфигурациями, которые вы нынче будете создавать.

Теперь давайте-таки создадим само приложение. Заходим в созданную группу и создаем новый элемент:

N	Конфигурация	Представление	Версия конфигурации
1	orders 1	Заказы	1

Программа заполнит все сама. Нажимаем на заветную кнопку *Собрать приложение*. Ждем. Это займет некоторое время. Периодически будут мигать черные окна консоли – не закрывайте их.

После сборки приложений, если у вас высветились ошибки доступа к файлам, то перезагрузите компьютер, запустите 1С под администратором и попробуйте еще раз.

Теперь нам надо забрать эти файлы. Забрать их можно на вкладке *Результаты сборки*.

Заказы; версия 1; сборка 1 (Мобильные приложения)

Записать и закрыть Записать Собрать приложение Еще ▾ ?

Параметры сборки Результаты сборки

Получить результаты сборки... Показать журнал... Получить приложение... Еще ▾

N	Приложение	Журнал сборки	Приложение
1	✓	✓	Мобильное приложение для ОС Android (ARM)
2	✓	✓	Мобильное приложение для ОС Android (x86)

Получить файл

Открыть или сохранить файл?
ua.dnv.orders-arm.apk

Открыть Сохранить Отмена

Выбираем нужную нам строку и жмем *Получить приложение*. Откроется диалог сохранения файла. Сохраняем файл в нашу папку, теперь устанавливаем его на эмулятор или на телефон по примеру, который описан был выше.

Повторите тоже самое для конфигурации «Склад». И тоже установите на телефон. И так, нам надо работать с двумя конфигурациями, а это дико не удобно, когда они являются разными приложениями.

Давайте сделаем так, чтобы эти две конфигурации у нас были в одном приложении. Для этого создадим новую группу в справочнике *Мобильные приложения*. И заполним ее вот так:

Мобильные приложения (создание группы) *

Записать и закрыть Записать Еще ▾ ?

Наименование:

Поставщик:

Мобильное приложение собирать

Для ОС iOS:

Для ОС Android:

Мобильная платформа: - ⓘ

Графические ресурсы: - ⓘ

Идентификатор решения: [Проверить уникальность идентификатора](#)

Сертификат для сборки на компьютере Apple: - ⓘ

Ключ для карт Google

Параметр получения ключа для работы с картами Google:

Ключ для работы с картами Google:

Версия приложения: Номер сборки:

Добавить Еще ▾

N	Источник
1	Заказы
2	Склад

Отличие в том, что в табличной части, где указываются ресурсы конфигураций – я указал и *Заказы* и *Склад*. Теперь создаем элемент справочника в этой группе:

Мобильные приложения (создание)

[Записать и закрыть](#)
[Записать](#)
[Собрать приложение](#)
[Еще ▾](#)
[?](#)

[Параметры сборки](#)
[Результаты сборки](#)

Наименование:

Поставщик:

Мобильное приложение собирать

Для ОС iOS:

Для ОС Android:

Полный идентификатор приложения:

Ключ для работы с картами Google:

Сертификат для сборки на компьютере Apple:

Мобильная платформа:

Графические ресурсы:

Язык представления мобильного приложения

Язык: Представление:

Версия приложения: Номер сборки:

[Конфигурации мобильного приложения](#)
[Представления мобильного приложения](#)

[Добавить](#)
[Еще ▾](#)

N	Конфигурация	Представление	Версия конфигура...
1	orders 1	Заказы	1
2	warehouse 1	Склад	1

На вкладке *Представление мобильного приложения* мы можем указать название нашего приложения, оно будет показываться в приложениях:

[Конфигурации мобильного приложения](#)
[Представления мобильного приложения](#)

N	Представление языка	Представление конфигурации
1	Русский (ru)	Заказы+Склад

Но давайте разберемся попутно с еще одним моментом – это иконки и заставка. Для этого откроем справочник *Графические ресурсы*. И создадим там новый элемент, который назовем *Заказы+Склад*.

1С нам подсказывает структуру архива:

При загрузке графической информации системы ожидает, что графическая информация будет представлена *zip*-файлом со следующей структурой (регистр имени файла и каталога – важен!):

- *Android\icon-36x36.png*
- *Android\icon-48x48.png*
- *Android\icon-72x72.png*
- *Android\icon-96x96.png*
- *Android\splash-320x480.png*
- *Android\splash-480x854.png*
- *Android\splash-640x960.png*
- *Android\splash-768x1024.png*
- *Android\splash-800x1280.png*
- *Android\splash-854x480.png*
- *Android\splash-1024x768.png*
- *Android\splash-1280x800.png*
- *Android\splash-1536x2048.png*
- *Android\splash-2048x1536.png*

Где *icon* – иконка соответствующего разрешения для приложения, а *splash* – заставка. Давайте сделаем такой архив. Для этого зайдём, например, на сайт <http://www.iconsearch.ru> справа вверху выберем фильтр для иконок 128x128. И найдем то, что нам по нраву. Не забываем про лицензии на приложения. Мне, например, понравилась вот эта иконка <http://www.iconsearch.ru/detailed/19501/6/>.

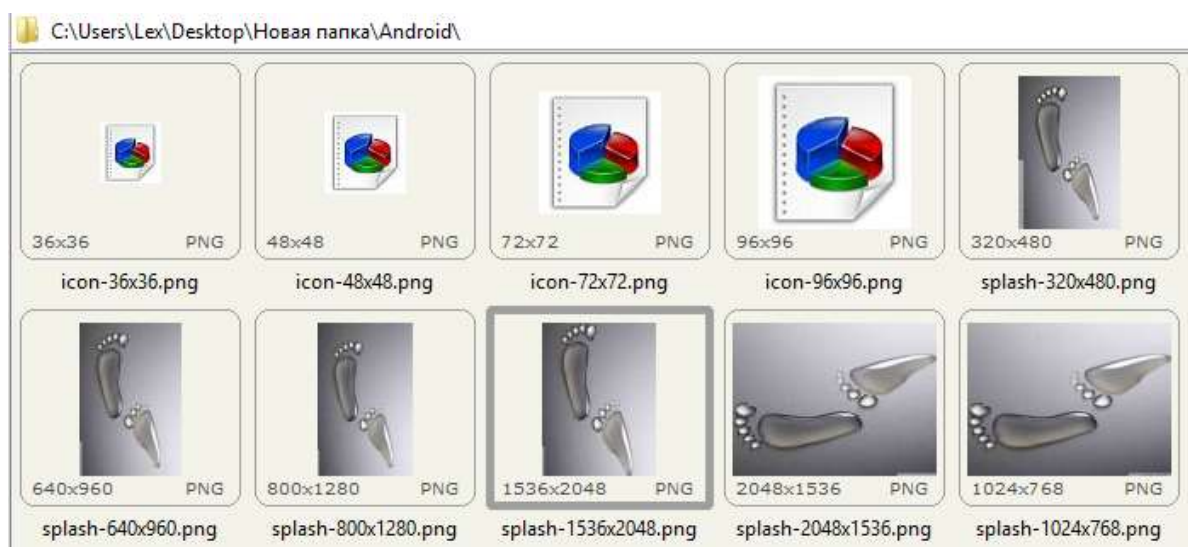
Тут нужного мне формата нету, поэтому я просто качаю эту иконку и пережимаю под меньшие размеры. Потом иду на другой сайт, например, на этот <http://wallpaperswide.com>, и ищу то что мне понравилось.

Я думаю, что все понимают, что изображение должно нормально выглядеть и под разные форматы экрана (4:3, 16:9) и под разные разрешения экрана, ну и, конечно же, под разную ориентацию.

Мне, например, понравилась вот эта картинка:
http://wallpaperswide.com/footprints_2-wallpapers.html.

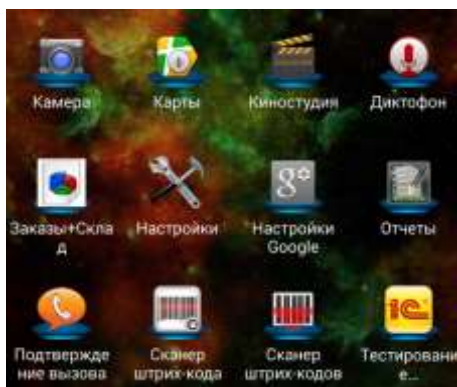
Там же я качаю нужные мне форматы. Однако обратите внимание – все придется пережать в *png*, и повернуть на 90 градусов, так как в имени – разрешение не зря указано именно в такой последовательности.

У меня получился вот такой набор:



Теперь я это все архивирую и вставляю в конфигурацию, выбираю этот графический набор и компилирую приложение.

Вот так теперь оно выглядит среди прочих приложений:



Вот так оно выглядит среди остальных приложений. Кстати обратите внимание на то, что при пережатии при помощи *Paint* – иконка потеряла прозрачность. Так что учтите это. Вот так теперь выглядит окно запуска:



И при входе – мы увидим пустой список баз, нажимаем *Добавить*. И выбираем нужный нам шаблон.



Почему шаблон? Потому что у вас может быть много баз с одинаковыми конфигурациями. При обновлении – вы обновляете шаблон, и, как следствие, все базы, которые были на основе него сделаны.

Мобильная платформа, конфигурация, обновление

Каким образом можно разрабатывать под мобильную платформу

К текущему моменту вы уже должны были бы определиться с вопросом о том, подходит вам мобильная платформа или нет. В независимости от вашего ответа мы пойдем дальше.

Итак, как же все-таки работать с мобильной платформой? Как передать в нее конфигурацию?

Давайте начнем с того, каким образом можно получить то, что мы хотим на мобильном устройстве?

Напомню, на текущий момент у нас существуют два варианта:

1. Установить платформу для разработчика и добавить в нее конфигурацию через публикацию на компе.
2. Скомпилировать нашу конфигурацию до арк-файла (это файл установки программы на андроиде) и запустить его на нашем телефоне.

Обновляемая конфигурация

Обновляемая конфигурация позволяет вам работать в одном мобильном приложении с несколькими разными базами, а точнее – их конфигурациями, а также обновлять конфигурации через свой сервер.

Под нее программировать довольно просто: сконфигурировали — опубликовали в web (в конфигураторе: Конфигурация — Мобильное приложение — Опубликовать; указываем имя публикации, отличное от имени публикации базы данных для веб доступа, если таковое имеется), затем нажали кнопку «Обновить» на телефоне.

Или поставили галочку «Обновлять мобильные приложения» в конфигурации: тогда при входе пользователя в базу на телефоне (при наличии доступа к серверу конфигураций) или если он находится в текущий момент в этой конфигурации, то мобильное приложение само обновит конфигурацию.

Не нужно ничего компилировать и обновлять вручную. Правда в этом случае вы не сможете обновить мобильную платформу, не скинув ее отдельным файлом, или без помощи других сервисов.

Фиксированная конфигурация

Это вариант скомпилированной до арк-файла конфигурации.

Соответственно, если вы изменили что-то в конфигурации, вам необходимо всем пользователям отправить этот файл (вес — около 40 Мб, стоит отметить, что с версии 8.3.3.42 — файл весит чуть более 20 Мб, но теперь надо самостоятельно контролировать архитектуру процессора и ставить соответствующую версию), и все должны переустановить приложение на телефоне. В этом случае обновится и платформа, и конфигурация. Данные должны остаться.

Если вы разработчик, который планирует написать конфигурацию для учета звезд во Вселенной и поделиться ею со всеми живыми, и не очень, существами, то вам однозначно необходим вариант фиксированной конфигурации. Выбрав этот вариант, вы можете расположить свой арк-файл в интернет-магазинах типа Google Play, где приложение будет продаваться или распространяться бесплатно (к слову, 1С еще не определилась с лицензиями до конца, но разрешает использовать без её ведома до 50 мобильных устройств).

Если конфигурация обновляется, вы размещаете новый арк-файл в интернет-магазине, и вскоре он обновится у всех купивших приложение.

Если вы штатный программист, которому дали задание роботизировать группу торговых представителей, то ваш выбор однозначно должен пасть на вариант без компиляции. В этом случае вам достаточно установить на телефон мобильное приложение, которое вы скачали, и добавить в него новую базу, прописав путь к опубликованной конфигурации. Этот вариант подходит также для отладки конфигурации до момента конечной компиляции до арк-файла.

Дмитрий Шерстобитов, г. Одесса

Спикер конференции Infostart Event Revolution 2013 (IER2013)

Модератор секции “Мобильная платформа + Интеграция” IER2013