

Section 3. The Conversion

Step 1. Create the New 'ABCX' Project

1. Bring up B4A
2. Click on File/New/B4XPages
3. Enter the Project Name as 'ABCX' and click 'Ok'.
4. Click on Project/Modify Configurations and change the Package name to 'ABCX'
5. Click on Project/Choose Icon and browse to the source of your icon. Select it.
6. Save the Project.
7. 'X' off B4A or bring up another project so as to get out of the new ABCX project.
8. Note on KeyStore. We found that all of our conversions (and all of our apps) pointed to a keystore in our original App. We did not change this and all are working fine. However, you should confirm that the new app is pointing to the same keystore location as the original. See 'Tools'/Private Sign Key'.
9. Note on Configuration. We found that the 'X' app did not respond to the google billing routines. We think that the ABCX.apk name was sent to Google and it did not recognize the app. Once we converted back to the original name, the billing routines worked. See also Step 12, Item 9.

Step 2. Copy your existing infrastructure into the new structure

1. Copy the .bas from C:\B4A\ABC into C:\B4A\ABCX.
2. Copy all the files from C:\B4A\ABC\Files into C:\B4A\ABCX\B4A\Files.
3. Copy all the files except .bal files into C:\B4A\ABCX\SharedFiles

Note: Why not the .Bal files? We tried it and when we made a change to the .Bal file from the IDE, it apparently only updated the file in the C:\B4A\ABCX\B4A\Files directory. When the next auto-copy occurred it scrapped over the file we updated. We also are not sure the bal files for B4I are identical so for now we are leaving them out of the shared files.

Step 3. Modify 'Main'

Bring up the new project 'ABCX' and modify 'Main' to your current project. You might consider bumping the #VersionCode up 1 higher than your current version so the load to the Play Store will be 'happy' and bump the #VersionName up to a whole new version to celebrate the conversion to B4XPages.

```
#ApplicationLabel: ABCX  
#VersionCode: 22  
#VersionName: V2.0
```

Make sure #BridgeLogger: True is not commented out.

At this point the 'Main' template will have all the current infrastructure to handle the left arrow on each title page as well as the 3 dot menu features that will be discussed below. At this point your original Keystore should be loaded.

Step 4. Add the 'Files' to the App

1. Click on the 'Files' Option in the 'Log' region of the IDE
2. Click on 'Add Files'.
3. Browse to C:\B4A\ABCX\B4A\Files
4. Highlight all files and click on 'Open'.
5. At this point most of the files are 'unused' and you will get a purple error in the log for each of them.
6. **Caution** on updating .txt/.jpg/.png/etc. files. Always update them in the 'Shared Files' directory as they are copied into the C:\B4A\ABCX\B4A\Files directory every time you compile.

Step 5. Add your libraries

1. Return to your original App and write down all the libraries you are using.
2. Return to the new App and add those libraries.

Step 6. Replace your Manifest

1. Return to your ABC App and bring up the manifest editor
2. Copy the entire manifest
3. Return to the ABCX App and bring up the manifest editor.
4. Paste the contents over the top of everything in the manifest.
5. Save it.

Step 7. Load Starter and Other Classes

Starter is already a class. Our experience, so far, has indicated that we can load Starter as is. We use Starter for all SQL operations, building our database at start up, loading initial setup entries into the database, and reading the setup entries every time the app starts. We also have common routines in Starter that are called from Activities (now classes).

1. Click on Project/Add Existing Module and browse to B4A\ABCX to find Starter and click on it to bring it in. Be sure to enter the option "Link-Relative Path". Click Ok and then save the Project.
2. Load any other classes you use in your App in the same way.

Step 8. Modify 'B4XMainPage'

At this point 'B4XMainPage' is a minimal template. This step will bring in all the logic of your current 'Main'.

In the Ide, bring up 'B4XMainPage'. It will look like this:

```
#Region Shared Files
#CustomBuildAction: folders ready, %WINDIR%\System32\Robocopy.exe,"..\..\Shared Files" "..\Files"
'Ctrl + click to sync files:
ide://run?file=%WINDIR%\System32\Robocopy.exe&args=..\..\Shared+Files&args=..\Files&FilesSync=True
#End Region
'Ctrl + click to export as zip: ide://run?File=%B4X%\Zipper.jar&Args=Project.zip
Sub Class_Globals
    Private Root As B4XView
    Private xui As XUI
End Sub
Public Sub Initialize
'    B4XPages.GetManager.LogEvents = True
End Sub
'This event will be called once, before the page becomes visible.
Private Sub B4XPage_Created (Root1 As B4XView)
    Root = Root1
    Root.LoadLayout("MainPage")
End Sub
'You can see the list of page related events in the B4XPagesManager object. The event name is B4XPage.
Private Sub Button1_Click
    xui.MsgboxAsync("Hello world!", "B4X")
End Sub
```

1. Un-comment the `B4XPages.GetManager.LogEvents= True`.
2. Save the new App. Bring up your original App and copy the contents of 'Main' from the 'Sub class Globals' down.
3. Bring up the new App and B4XMainPage. Highlight from the top of the `Private Sub Button1_Click` routine on down. Ctrl/V to paste over it.
4. Move all of your Globals and Process Globals up into the `Class_Globals` above.

5. Drop the now empty **Sub Class_Globals** and **Sub Process_Globals**.
6. Modify the **Root.LoadLayout** (“MainPage”) to your main page .bal file name.
7. Remove the **Sub Activity_Create(FirstTime As Boolean)** line.
8. Modify the following as shown:

Activity_Resume Activity.color Activity.Top/Left/width/height StartActivity ("xxxx") will when you convert those Activities as described below. Activity.Title="xxx"	to Private Sub B4XPage_Appear to Root.Color=. to Root.xxx. to B4XPages.ShowPage ("xxxx"). These won't work yet, but to B4XPages.SetTitle (Me, "xxx").
--	--

Step 9. Compile the Project and load any database.

At this time, you should have a clean log. If not, review the steps above. You may have left an “Activity” or something else un-changed.

1. Compile the App to your device. It should bring up your App Main Menu. It may crash if something is needed that is not yet loaded. This is ok as we just want to load the database. Following that, it may start up.
2. If you create databases using the format (rp.getsafeDirDefaultExternal("ABC"),"ABC.DB",True) then you will have created your initial database that a new user would have. Since it is ‘External’, you can test your app with what a new user would ‘see’ and later you can:
 - a. Unload your App.
 - b. Find your existing database on your device and copy it into ABCX. Leave it in directory ABC and leave the name ABC.DB. You will now have your full database for the ABCX App and will be able to test with your full database as you add the remaining Activities.
 - c. Restart your App.

Step 10. Load any converted ‘Shared Modules’

If you have already converted modules in the ‘Shared Modules’ folder to B4XPages, load them as ‘Existing Modules’ and check the box for ‘link-relative path. Any unconverted ‘Shared Modules’ must be converted as in Step 11.

My own Modules-These are specific to my conversions. If another user has specific instructions, they could replace these with their own.

Before starting, move the color definitions and settings from B4XMainPage to Starter.
 This will break a number of color settings in B4XMainPage. Mod them to starter.xxcolor

PageFS01P-Load it from ‘Shared Modules’ and make the following changes.

- a. In starter, set up public vars for bPswOk, bPswQuit.
- b. Bring up GoShop and copy the logic in B4XPageAppear related to the Optin and Password into the new app PageAppear.
- c. Pull the pswd panel out of FS01.bal/kill the psw panel and views in B4XMainPage.

PageFS92-Load it from ‘Shared Modules’ and make the following changes.

- a. In starter, add public variables AppName, DBName,DBDir
- b. In Starter, set the values specific to the new App
- c. Copy file HelpEMail.txt from Goshop into the new Apps Help files and into the ‘shared’ files. Add it to the Apps ‘files’.

PageFS94-Load it from ‘Shared Modules’

PageFS95-Load it from ‘Shared Modules’

PageFS96-Complex steps as the old FS96 held the ‘About’ display.

- a. Load your original FS96 as an activity.
- b. Copy PageFS99.bas from GoShop into your new App
- c. Load the new PageFS99
- d. Copy the “About” portion of your old FS96 into the PageFS99.
- e. Unload FS96.
- f. Load the new PageFS96 from ‘Shared Modules’.
- g. In the B4XMainPage, modify the ‘Help About’ entry in the 3dot logic to start ‘FS99’.

PageFS97-Load it from ‘Shared Modules’

For each of these modules:

- a. Copy the .bal from GoShop into the 'SharedFiles' and 'Add File' to the App.
- b. enable the Pages in B4XMainPage as described in 8-9 of the next step.

Step 11. Modify Activities to Classes

We now need to modify all of our remaining Activities to Classes. There are two choices for this:

1. Make minimal changes to the Activity using notepad and then make the rest of the mods using the IDE.
2. Do most of the changes in notepad.
3. Furthermore, you have the choice of modifying all Activities at once or do one at a time.

We suggest doing one Activity at a time, in an order progressing through the App from the Main Menu deeper into the App. We also suggest doing the minimal changes and then doing the remaining changes in the IDE as the IDE will highlight errors that you can quickly address.

Change the Name of the module to 'Pagexxxx'.

For all modules in the new ABCX directory, change the names to add "Page" in front of the old name.

For 'Shared Modules'

If you have located common modules in C:\B4A\SharedModules you cannot simply rename the module as that will leave your un-converted Apps without that Module at compile time.

Instead, ctrl/c and ctrl/v each shared module. Then, rename each copy to its 'Page' version. You will then have an 'FS96' and a 'PageFS96'. Then, follow the instructions below for the conversion. Happily, you will only have to convert these once as your other Apps will then use them.

Minimal Changes Using Notepad: Bring up an activity (PageFS20) in notepad. It will look something like this:

1. Modify the entries as shown in blue:

B4A=true	
Group=Default Group	
ModulesStructureVersion=1	
Type=Activity	Type=Class
Version=8.5	Version=11.0
@EndOfDesignText@	
#Region Activity Type=Activity Attributes	Drop
#FullScreen: false	Drop
#IncludeTitle: True	Drop
#End Region	Drop
Sub Process_Globals	Sub Class_Globals
Xxx	
Xxx	
End Sub	Drop
Sub Class_Globals	Drop
Xxx	
xxx	

2. Save it.

Remaining Changes using B4A.

3. Click on 'Project'/'Add Existing Module' and browse to this module in C:\B4A\ABCX. Select it and then click the box for 'link-Relative path' to load it.
4. In the new class module, make the changes below. Note that in most cases you can copy these new entries in from B4XMainPage or from one of the previously converted modules.
5. Under the top Sub Class_Globals add

- ```

Private Root As B4XView 'ignore
Private xui As XUI 'ignore

```
- Add the following before the Sub Activity\_Create.

```

Public Sub Initialize
 Return Me
End Sub

```

(this is required or the entry for FS20.initialize in B4XMainPage will show an error).
  - Make the following changes as shown:

|                                                                                                |                                                                                                                                                                                                                                                                                                                        |
|------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> Sub <b>Activity_Create</b>(FirstTime As Boolean)     Activity.LoadLayout ("FS20") </pre> | <pre> Private Sub <b>B4XPage_Created</b> (Root1 As B4XView)     Drop     Add Root = Root1     Add Root.LoadLayout("FS20")     to Private Sub <b>B4XPage_Appear</b>     to Root.Color=.     to Root.xxx.     to B4XPages.ShowPage ("xxxx").     to B4XPages.SetTitle (me, "xxx").     to B4XPages.ClosePage(Me). </pre> |
|------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

  

|                                                                                                                                               |  |
|-----------------------------------------------------------------------------------------------------------------------------------------------|--|
| <pre> <b>Activity_Resume</b> Activity.color Activity.Top/Left/width/height StartActivity ("xxxx") Activity.Title="xxx" Activity_Finish </pre> |  |
|-----------------------------------------------------------------------------------------------------------------------------------------------|--|

**Enable the new Page in B4XMainPage**

- Add to Sub Class\_Globals

```

Public FS20 As PageFS20 ' be careful to change both of these!!

```
- Add to Private Sub **B4XPage\_Created** (Root1 As B4XView)

```

FS20.Initialize
B4XPages.AddPageAndCreate("FS20", FS20)

```

**Compile the project after each module is added and test the new module.**