BlackBerry Developer Summit

Launcher Integration

Table of Contents

Launcher Integration

1. Integrate the launcher into a BlackBerry Dynamics app

The Launcher can be integrated into BlackBerry Dynamics projects. Download the BlackBerry Launcher library from here:
https://developers.blackberry.com/us/en/resources/downloads.html

View the development guide for the Launcher library:
http://help.blackberry.com/en/blackberry-dynamics-sdk-launcher-library/2.7/

An example of such integration is shown in project AppKinetics_Launcher app available in the download page of the BlackBerry developer event.

www.blackberrydevsummit.com

Here are the steps required to integrate the Launcher with your BlackBerry Dynamics application on iOS:

1. Make a copy of the AppKinetics sample from the folder below.

    ```
    ~/Library/Application Support
    /BlackBerry/Good.platform/iOS/Examples/objective-c/
    ```

2. Open the copied AppKinetics project in Xcode. You will be intergrating the Launcher library into this application project.

3. If you are using XCode10. There are two ways to make existing apps compatible with XCode 10.

    1. Change path in the xcconfig file. Update the existing path for LDPLUSPLUS and LD from
        LDPLUSPLUS=~/Library/Application Support/BlackBerry/Good.platform/iOS/FIPS_module/$FIPS_PACKAGE/bin/gd_fipsld
        LD=~/Library/Application Support/BlackBerry/Good.platform/iOS/FIPS_module/$FIPS_PACKAGE/bin/gd_fipsld
        to
        LDPLUSPLUS=$(HOME)/Library/Application Support/BlackBerry/Good.platform/iOS/FIPS_module/$FIPS_PACKAGE/bin/gd_fipsld
        LD=$(HOME)/Library/Application Support/BlackBerry/Good.platform/iOS/FIPS_module/$FIPS_PACKAGE/bin/gd_fipsld

   Note the change from ~ to $(HOME).

    2.   Switch to Legacy Build System. You can switch to Legacy Build System in XCode by going to File>Project Settings>Build System

4. Unzip the Launcher distribution to a folder inside your application project, adding it to the Xcode project and make sure the *Launcher.framework* and *LauncherBundle.bundle* are added to the project as shown below.
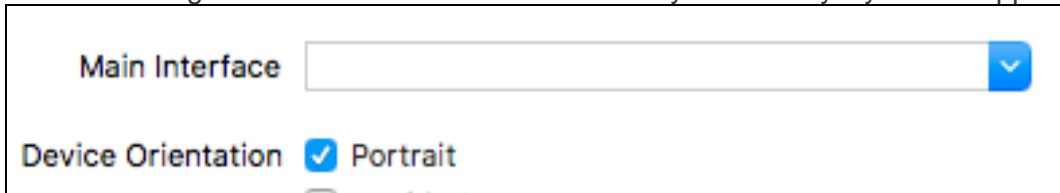
5. If necessary add Launcher.framework to *Link Binary with Libraries* phase:

6. If necessary add LauncherBundle.bundle to *Copy Bundle Resources* phase:
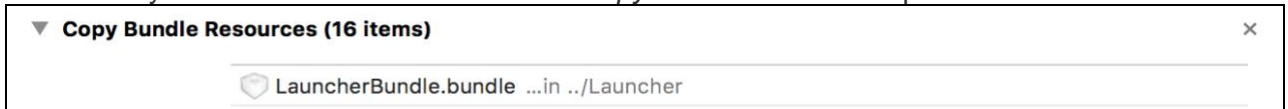
7. Ensure that the Main Interface setting for both iPhone and iPad under the target's General settings is cleared as recommended for any BlackBerry Dynamics application

8. If needed add *GDFetchResources* boolean value to Info.plist with value *YES* to ensure icons are downloaded:

9. In *Other Linker Flags* add -ObjC if needed

10. The Launcher lib must be initialised. The initialisation comprises 2 steps: configure the UI and start the service. Both must be done once after the application has been authorised.

   - To initialise the UI, an instance of `GTLauncherViewController` must be set as root view controller of your application. The view hierarchy of your app is preserved initialising the `GTLauncherViewController` with the view controller you would have set as root view controller.

   - To initialise the service the method `startServicesWithOptions` must be called. The options are defined in `GTLauncherServicesStartupOptions` and one must be used depending on whether your app makes use of any of the following BlackBerry Dynamics features:

o Push Connection management, the GDPushConnection class.
o Authentication Token, the GDUtility class.

The selection of the option value is dependent on whether the application registers for either GDPushConnection or GDUtility delegate callbacks.

- If the application uses neither API then use:
  GTLInternalGDAuthTokenAndPushConnectionManagement

- If the application registers for GDPushConnection events use:
  GTLHostGDPushConnectionManagement and pass status messages from
  `[GDPushConnectionDelegate onStatus]` to
  `[GTLauncherViewController setGDPushConnectionStatus]`

- If the application registers for GDUtility events use:
  GTLHostGDAuthTokenManagement and implement the
  `didRequestGDAuthTokenForServerName` delegate callback

Consult the technical documentation for advanced usage of the Launcher lib. This include details about how to customise the settings UI as well as the interactions between the UI controlled by the Launcher and by the application itself.

2. Your Application's Visibility in the Launcher

How can your application be made to appear in the Launcher of other applications?
Your application will appear in the Launcher of other applications, if it provides at least one AppKinetics service, for example:
- Transfer File (com.good.gdservice.transfer-file)
- Presence Service (com.good.gdservice.enterprise.presence)

What if your application doesn't provide any AppKinetics services?
There is a dummy service that you can register as providing:
- Launch (com.good.gdservice.launch)

There is no need to write any code to handle service requests for the Launch service.
Details of all shared services are accessible through the *developers.blackberry.com* site, or via this shortcut: https://apps.good.com/#/services

To add an app kinetics service to your app, view your app in UEM and add the service with the '+' button under 'Version', as seen in the screenshot below:

APP Dev Summit App 312

| Settings | Assigned to 358 users | Assigned to 2 groups |

BlackBerry Dynamics app

BlackBerry Dynamics app ID
com.bbdevsummit.devapp312

| BlackBerry Dynamics | iOS | macOS | Android | Windows |

Override BlackBerry Dynamics profile
- Select -

Override compliance profile
- Select -

App configuration Upload a template

Name	Ranking	+
None assigned		

Server configuration payload
Add

Version

Version	Release status	Service bindings	Note	+
1.0.0.0	PROD	Launch Service 1.0.0.0, Launchable Service 1.0.0.0		×

User certificates
☐ Allow BlackBerry Dynamics apps to use user certificates and user credential profiles