



#UACP security level English Customer
SAP BusinessObjects Mobile (for iOS)
Document Version: 6.8.1 – 2018-02-23

Developer's Guide

Content

- 1 Document History. 4**
- 2 About This Document. 6**
- 3 Building the Application with your Certificate. 7**
- 4 Customizing the Visual Features of the Application. 8**
 - 4.1 Customizing the Name of the Installed Application. 8
 - 4.2 Customizing the Launch Icon and the Launch Screen. 8
 - 4.3 Customizing the Application Password Screen. 11
 - 4.4 Customizing Getting Started. 13
 - 4.5 Customizing the Appearance of the Home Screen. 16
 - 4.6 Customizing the Browse Screen (Left Panel) 20
 - 4.7 Customizing Icons Within the Application. 21
- 5 Customizing the Default Behavior of the Application. 25**
 - 5.1 Customizing the Onboarding Features. 26
 - 5.2 Customizing the Application Functionality. 33
 - 5.3 Customizing the Protocols. 50
 - 5.4 Customizing the Sample BI Documents. 52
- 6 Extending the Feature Capabilities of the Application. 54**
 - 6.1 Implementing Extensions. 54
- 7 Embedding SAP BI Analytics. 60**
 - 7.1 Modifying Your Project for Use with SAP BI Libraries. 60
 - 7.2 Using SAP BI Libraries to Perform Tasks. 63
 - 7.3 Integrating with Third Party Security Solutions. 64
 - Wrapping the Application with Citrix Xenmobile (iOS). 64
 - Blackberry Dynamics SDK. 67
 - Airwatch SDK. 71
 - MobileIron AppConnect SDK. 74
 - Microsoft Intune Application Configuration SDK. 76
 - Configuring Microsoft Authentication Library Cross app-SSO with SAP Business Objects Mobile Application. 91
- 8 Customizing Mobile Dashboards. 96**
- 9 Configuring Application's Behavior Using the SAP BusinessObjects Mobile Server. 97**
 - 9.1 Configuring the Behavior of the Application by Setting Properties on the Server. 97

| | | |
|-----------|--|------------|
| 9.2 | Configuring the Session Timeout Duration of the Application. | 98 |
| 9.3 | Configuring the Thumbnail Background Image of BI Documents. | 99 |
| 9.4 | Configuring Document-Specific Help. | 100 |
| 10 | Mobile Server and Property File Locations Matrix. | 102 |
| 11 | Using the CMC to Define Properties on BI Platform 4.1 or 4.2. | 103 |

1 Document History

The following table provides an overview of the most important document changes.

| Version | Date | Changes |
|--|-----------------|---|
| SAP BusinessObjects Mobile for iOS 6.8.16 | July 28, 2020 | Added a new section Configuring Microsoft Authentication Library Cross app-SSO with SAP Business Objects Mobile Application Integrating with Third Party Security Solutions [page 64] |
| SAP BusinessObjects Mobile for iOS 6.8.10 | September, 2019 | Added the section Microsoft Intune Application Configuration SDK under Integrating with Third Party Security Solutions [page 64] |
| SAP BusinessObjects Mobile for iOS 6.8.9 | May, 2019 | Updated the section Integrating the Application with the Blackberry Dynamics SDK [page 68] |
| SAP BusinessObjects Mobile for iOS 6.8.1 | February, 2018 | <ul style="list-style-type: none">• Added information on the <i>Prefer Cache</i> option in the Fiori theme under Customizing the Application Functionality [page 33]• Added the section MobileIron Application Configuration [page 75]• Added a note in the section Integrating the Application with the Blackberry Dynamics SDK [page 68] |
| SAP BusinessObjects Mobile for iOS 6.7.6 | December, 2017 | <ul style="list-style-type: none">• Added information on Consuming Client Certificates via Airwatch console<ul style="list-style-type: none">◦ Configuring the Application Profile for Certificate Push [page 72]◦ Setting up Airwatch for Integrated Authentication [page 73]• Added the section Wrapping the application with MobileIron App-Connect [page 74]. |

| Version | Date | Changes |
|--|----------------|--|
| SAP BusinessObjects Mobile for iOS 6.6 | June, 2017 | <ul style="list-style-type: none"> • Added information on Wrapping the Application with Citrix Xenmobile (iOS) [page 64] • Added information on Prerequisites for Integrating the Application with Blackberry Dynamics SDK [page 67] • Added information on Integrating the Application with the Blackberry Dynamics SDK [page 68] • Added information on Airwatch Application Configuration [page 73] • Added information on Airwatch Application Wrapping [page 71] • Updated table in the section Mobile Server and Property File Locations Matrix [page 102] |
| SAP BusinessObjects Mobile for iOS 6.5 | December, 2016 | <ul style="list-style-type: none"> • Updated relevant sections with latest xcode version. • Updated the sections: <ul style="list-style-type: none"> ◦ Modifying Your Project for Use with SAP BI Libraries [page 60] ◦ Using SAP BI Libraries to Perform Tasks [page 63] |

2 About This Document

This document is meant for developers and administrators of the SAP BusinessObjects Mobile (SAP BI) application for iOS, who host the application in their enterprise app store, customize the behavior of the mobile application features, or extend the application functionality based on their specific requirements. Most tasks explained in this document involve the use of the Xcode project bundle of the application that is provided at the SAP Service Marketplace (SMP).

The SAP BusinessObjects Mobile application SDK provides you with the following capabilities:

1. Ability to **customize** the application using a **metadata-driven approach**: In this approach, you can control the application's visual features, behavior and functionality using the values of features specified in the **.plist files** in the Xcode project.
2. Ability to **extend** the application's capabilities by **writing code using the classes provided within the application SDK** in Xcode environment.
3. Ability to **embed** Web Intelligence reports and Dashboards in the mobile applications that you have developed.

i Note

This document also covers information on how you can customize the standard SAP BusinessObjects Mobile application shipped by SAP (through the Apple App Store) by configuring **properties** on the Mobile BI platform server without using the SDK.

3 Building the Application with your Certificate

Prerequisite

1. You have downloaded the application project file `MOBSBOPIOSSDK00P_XXXXXXXXX.ZIP` from the SAP Service Marketplace.
(<https://support.sap.com/software/installations.html> > *Installation and Upgrades* > *A-Z Alphabetical list of my Products* > *M* > *SAP Mobile Solutions* > *SAP Business Objects Mobile on IOS Software Development Kit* > *MOB SBOP IOS SDK*).
2. You have the appropriate Xcode version to open and compile the project.
 - Use Xcode 8.x to open and compile SAP BusinessObjects Mobile application for iOS 6.5 SDK.
 - Use Xcode 7.x to open and compile SAP BusinessObjects Mobile application for iOS 6.4.x SDK.

Building the Application

To prepare the application for use by your local app store, perform the following steps:

1. Unzip the application project file to open the SAP BusinessObjects Mobile project.
A folder named `SAPBusinessObjectsMobile-<app_version_number>-xcodeproj-with-deps` appears.
2. Rename the folder `SAPBusinessObjectsMobile-<app_version_number>-xcodeproj-with-deps` to a shorter name, and ensure the name does not contain spaces or special characters.

i Note

In this example, the folder above is renamed to `MOBISDK`.

3. Access `MOBISDK/src/xcode/` and choose `SAPBI_SDK.xcodeproj` to start the project.
4. (Optional) Customize the application by making the required changes to the files contained in the `MOBISDK` folder, as described in subsequent chapters.
5. Build and run the project using your certificate.
Users can now use the generated application on iOS devices.

i Note

For more information on building the application, refer to <https://developer.apple.com/support/technical/code-signing> 🐾

4 Customizing the Visual Features of the Application

4.1 Customizing the Name of the Installed Application

Once installed, the SAP BusinessObjects Mobile application appears on your device screen with the name *SAP BI*. To change the application name based on your requirements, perform the procedure below:

1. Open the `SAPBusinessObjectsMobile-Info.plist` file available within `SAPBI_SDK.xcodeproj`.
2. Edit the parameter: `Bundle display name`.
3. Save the file, run the project, and validate your changes.

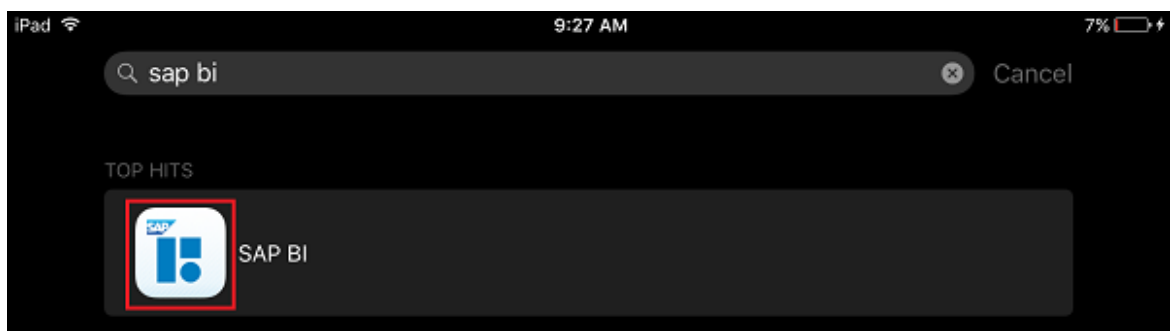
iNote

Do not change the Bundle version value in the `SAPBusinessObjectsMobile-Info.plist` file because changing the application version here may lead to issues during future upgrades of the application.

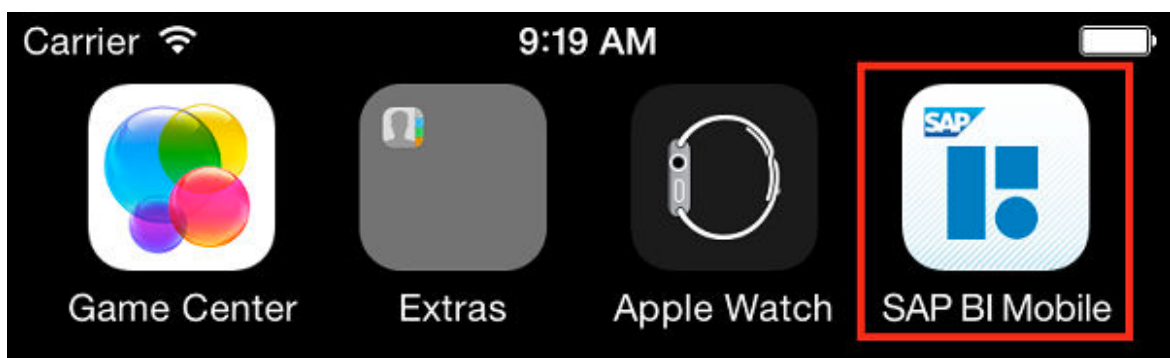
4.2 Customizing the Launch Icon and the Launch Screen

The application launch icon appears in the following places on an iOS device:

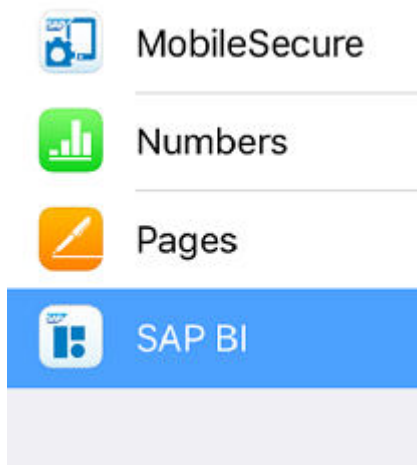
1. Spotlight:



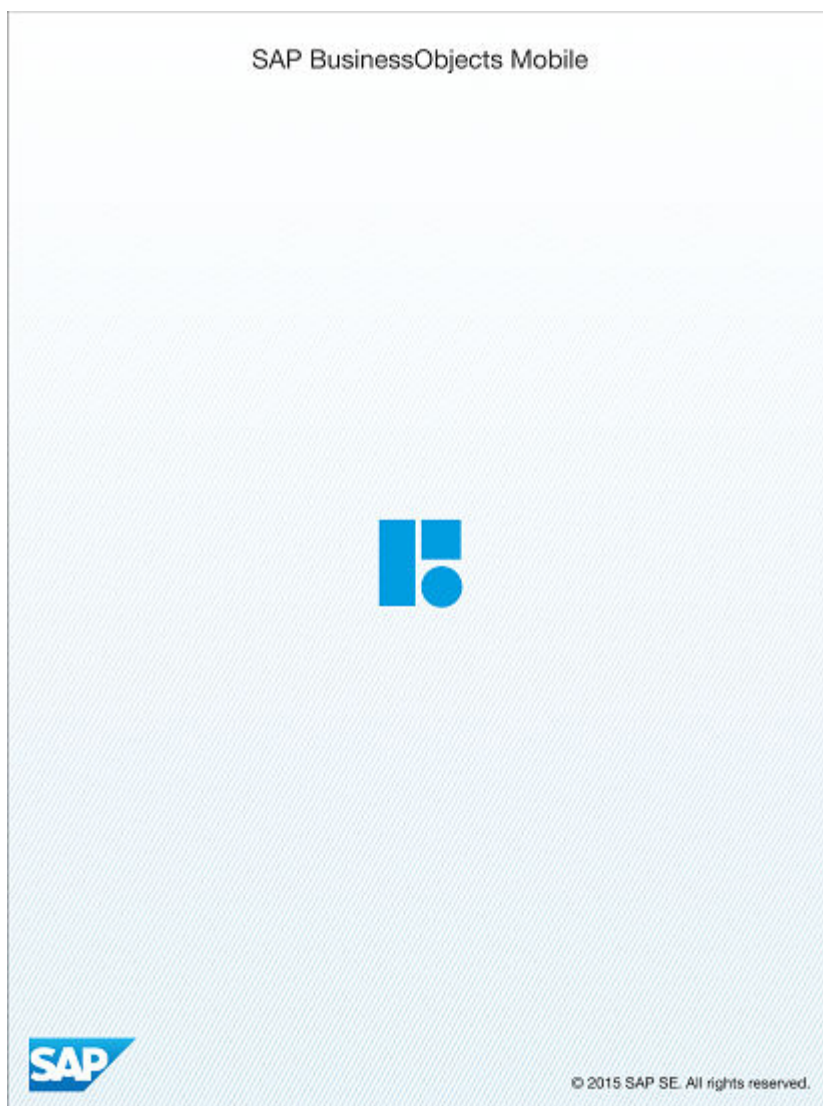
2. On the device home screen as shown below:



3. On the device *Settings* screen as shown below:



The application launch screen appears when the application is launched on the device:



The launch icons for the product are organized as an Asset Catalog.

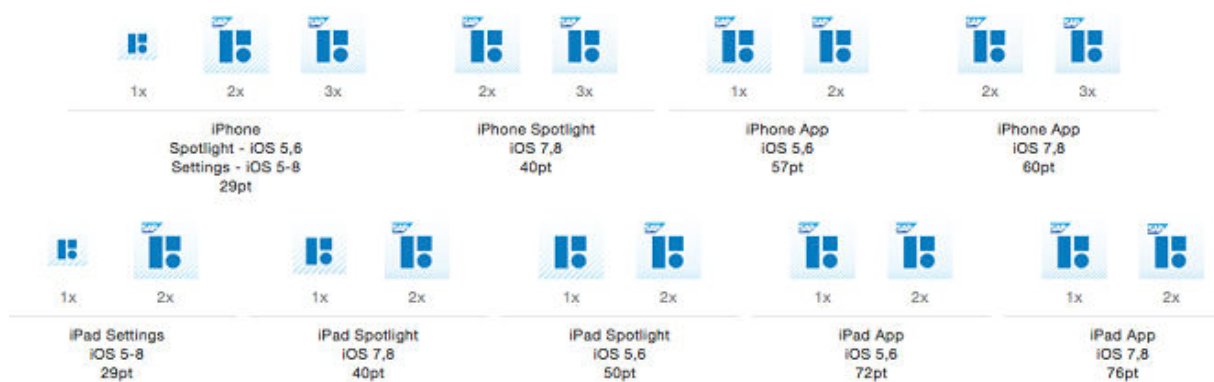
i Note

To learn more about Asset Catalogs, refer to https://developer.apple.com/LIBRARY/IOS/recipes/xcode_help-image_catalog-1.0/_index.html

To view or edit the Asset Catalog for launch icons in the SAP BI application, access **Xcode** > **Resources** > **Bundles** > **AppLaunch.xcassets** and select:

- *AppIcon* to customize the application icon.
- *LaunchImage* to customize the launch screens.

The following figure displays how the application icons appear in the Asset Catalog set viewer:



To replace an image with your own custom image, select the image (using Finder) and drag it to the appropriate image in the set viewer. You can see the defined size of each image at the bottom of the image in the set viewer.

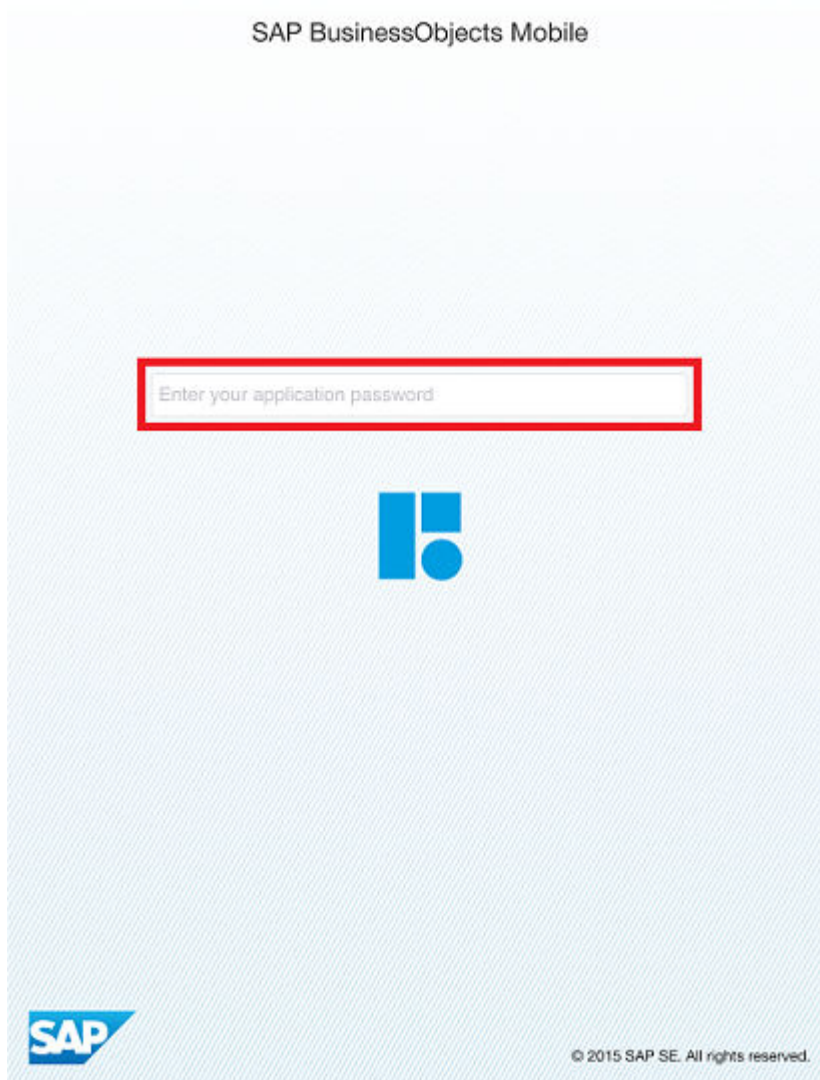
i Note

If you want to replace the application icon or launch screen images quickly, you can locate the files (using the Finder) in the following folder:

- `MOBISDK/src/xcode/Resources/Bundles/AppLaunch.xcassets/AppIcon.appiconset` (for application icons)
- `MOBISDK/src/xcode/Resources/Bundles/AppLaunch.xcassets/LaunchImage.launchimage` (for launch screen images)

4.3 Customizing the Application Password Screen

If the user has enabled the application password, the password screen appears every time the application is launched or brought to the foreground:



To customize the application password screen image, perform the procedure below:

1. Access `MOBISDK/src/xcode/Resources/Images/AppPwdBackground` using the Finder.
2. Choose the image and replace it with your custom image.

i Note

The name and dimensions of your custom image should be the same as those of the default image, and they are listed in the table below.

| Image file name | Image dimensions (pixels-width*pixels-height) | Image purpose |
|--|--|---|
| bg_SAPBI_splash_iPad_Landscape.png | 1024*768 | Application password background splash image for the landscape mode of iPad devices |
| bg_SAPBI_splash_iPad_Landscape@2x.png | 2048*1536 | Application password background splash image for the landscape mode of iPad devices with Retina displays |
| bg_SAPBI_splash_iPad_Portrait.png | 768*1024 | Application password background splash image for the portrait mode of iPad devices |
| bg_SAPBI_splash_iPad_Portrait@2x.png | 1356*2048 | Application password background splash image for the portrait mode of iPad devices with Retina displays |
| bg_SAPBI_splash_iPhone.png | 640*960 | Application password background splash image for the portrait mode of iPhone devices |
| bg_SAPBI_splash_iPhone@2x.png | 640*1136 | Application password background splash image for the portrait mode of iPhone devices with Retina displays |
| SAPBI_LaunchScreeniPhone6@2x.png | 750*1134 | Application password background image for iPhone 6 |
| SAPBI_LaunchScreeniPhone6Plus@3x.png | 1242*2208 | Application password background image for iPhone 6 Plus |
| SAPBI_LaunchScreen-568h@2x.png | 640*1136 | Application password background image for iPhone 5 |
| SAPBI_LaunchScreen-Landscape@2x~ipad.png | 2048*1536 | Application password background image for the landscape mode of iPad devices with Retina displays |
| SAPBI_LaunchScreen-Landscape~ipad.png | 1024*768 | Application password background image for the landscape mode of iPad devices (iPad 1 and iPad 2) |
| SAPBI_LaunchScreen-Portrait@2x~ipad.png | 1536*2048 | Application password background image for the portrait mode of iPad devices with Retina displays |
| SAPBI_LaunchScreen-Portrait~ipad.png | 768*1024 | Application password background image for the portrait mode of iPad devices (iPad 1 and iPad 2) |
| SAPBI_LaunchScreen@2x.png | 640*960 | Application password background image for iPhone devices with Retina displays |

| Image file name | Image dimensions (pixels-width*pixels-height) | Image purpose |
|-----------------|---|--|
| | | i Note On iPhone devices, the application is only launched in the portrait mode. |

3. Run the project and validate your changes in the application.

i Note

The image specifications are in accordance with the guidelines specified in the iOS developer library (Reference: https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/IconMatrix.html#//apple_ref/doc/uid/TP40006556-CH27-SW1).

4.4 Customizing Getting Started

Customizing the Getting Started Slides for Image, Title, and Sub-Titles

For iPhone devices

To customize Getting Started for the application on iPhone devices, follow the procedure below:

1. Using the Finder, access the `SAPBI_GettingStarted.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Right click on `SAPBI_GettingStarted.bundle` and choose *Show Package Contents*. All the images belonging to the bundle appear on the screen.
3. Select the image that you want to replace and replace it with your custom image.

i Note

- The name and dimensions of your custom image should be the same as those of the default image.
- You can identify the purpose of each image icon by looking at its name.
- `<ImageName>@2x.png` has the same purpose as `<ImageName>.png`, but is used for devices with **Retina** displays.

4. Open the `HelpSliderPageResourceFile-iPhone.plist` file in the bundle, and specify (or modify) the Help image name, title and sub-title as an array of items for each slide of Getting Started.

| Key | Type | Value |
|--------------------------------|------------|--|
| ▼ Root | Dictionary | (6 items) |
| backgroundImage | String | getting_started_bg.png |
| titleTextColor | String | 333333 |
| textColor | String | 666666 |
| pageIndicatorColor | String | 7F7F7F |
| pageIndicatorSelectedPageColor | String | 000000 |
| ▼ data | Array | (2 items) |
| ▼ Item 0 | Dictionary | (2 items) |
| info | String | These slides are used in case of new installation. Sequence of items in each slide (image name, title key, subtitle key) |
| ▼ slides | Array | (5 items) |
| ▼ Item 0 | Array | (3 items) |
| Item 0 | String | MOBI_Help_0 |
| Item 1 | String | helpPage.0.Title |
| Item 2 | String | helpPage.0.SubTitle |
| ▼ Item 1 | Array | (3 items) |
| Item 0 | String | MOBI_Help_1 |
| Item 1 | String | helpPage.1.Title |
| Item 2 | String | helpPage.1.SubTitle |
| ▼ Item 2 | Array | (3 items) |
| Item 0 | String | MOBI_Help_2 |
| Item 1 | String | helpPage.2.Title |
| Item 2 | String | helpPage.2.SubTitle |
| ▶ Item 3 | Array | (3 items) |
| ▶ Item 4 | Array | (3 items) |
| ▶ Item 1 | Dictionary | (2 items) |

5. Save your changes, run the project, and validate your changes in the simulator.

For iPad devices

To customize Getting Started for the application on iPad devices, perform the procedure below:

1. Using the Finder, access the `SAPBI_GettingStarted.bundle` file located at `: MOBISDK/src/xcode/Resources/Bundles/`.
2. Right click on `SAPBI_GettingStarted.bundle` and choose *Show Package Contents*. All the images belonging to the bundle appear on the screen.
3. Select the image that you want to replace, and replace it with your custom image.

i Note

- The name and dimensions of your custom image should be the same as those of the default image.
- You can identify the purpose of each image icon with the name of the image.
- `<ImageName>@2x.png` has the same purpose as `<ImageName>.png`, but is used for devices with **Retina** displays.

4. Open the `HelpSliderPageResourceFile-iPad.plist` file available in the bundle.
5. As shown in the figure below, specify (or modify) the Help image name, title and sub-title as an array of items for each slide of Getting Started.

| Key | Type | Value |
|--------------------------------|------------|--|
| ▼ Root | Dictionary | (6 items) |
| backgroundImage | String | getting_started_bg.png |
| titleTextColor | String | 333333 |
| textColor | String | 666666 |
| pageIndicatorColor | String | 7F7F7F |
| pageIndicatorSelectedPageColor | String | 000000 |
| ▼ data | Array | (2 items) |
| ▼ Item 0 | Dictionary | (2 items) |
| info | String | These slides are used in case of new installation. Sequence of items in each slide (image name, title key, subtitle key) |
| ▼ slides | Array | (6 items) |
| ▼ Item 0 | Array | (3 items) |
| Item 0 | String | MOBI_Help_0 |
| Item 1 | String | helpPage.0.Title |
| Item 2 | String | helpPage.0.SubTitle |
| ▼ Item 1 | Array | (3 items) |
| Item 0 | String | MOBI_Help_1 |
| Item 1 | String | helpPage.1.Title |
| Item 2 | String | helpPage.1.SubTitle |
| ▼ Item 2 | Array | (3 items) |
| Item 0 | String | MOBI_Help_2 |
| Item 1 | String | helpPage.2.Title |
| Item 2 | String | helpPage.2.SubTitle |
| ▼ Item 3 | Array | (3 items) |
| Item 0 | String | MOBI_Help_3 |
| Item 1 | String | helpPage.3.Title |
| Item 2 | String | helpPage.3.SubTitle |
| ▶ Item 4 | Array | (3 items) |
| ▶ Item 5 | Array | (3 items) |
| ▶ Item 1 | Dictionary | (2 items) |

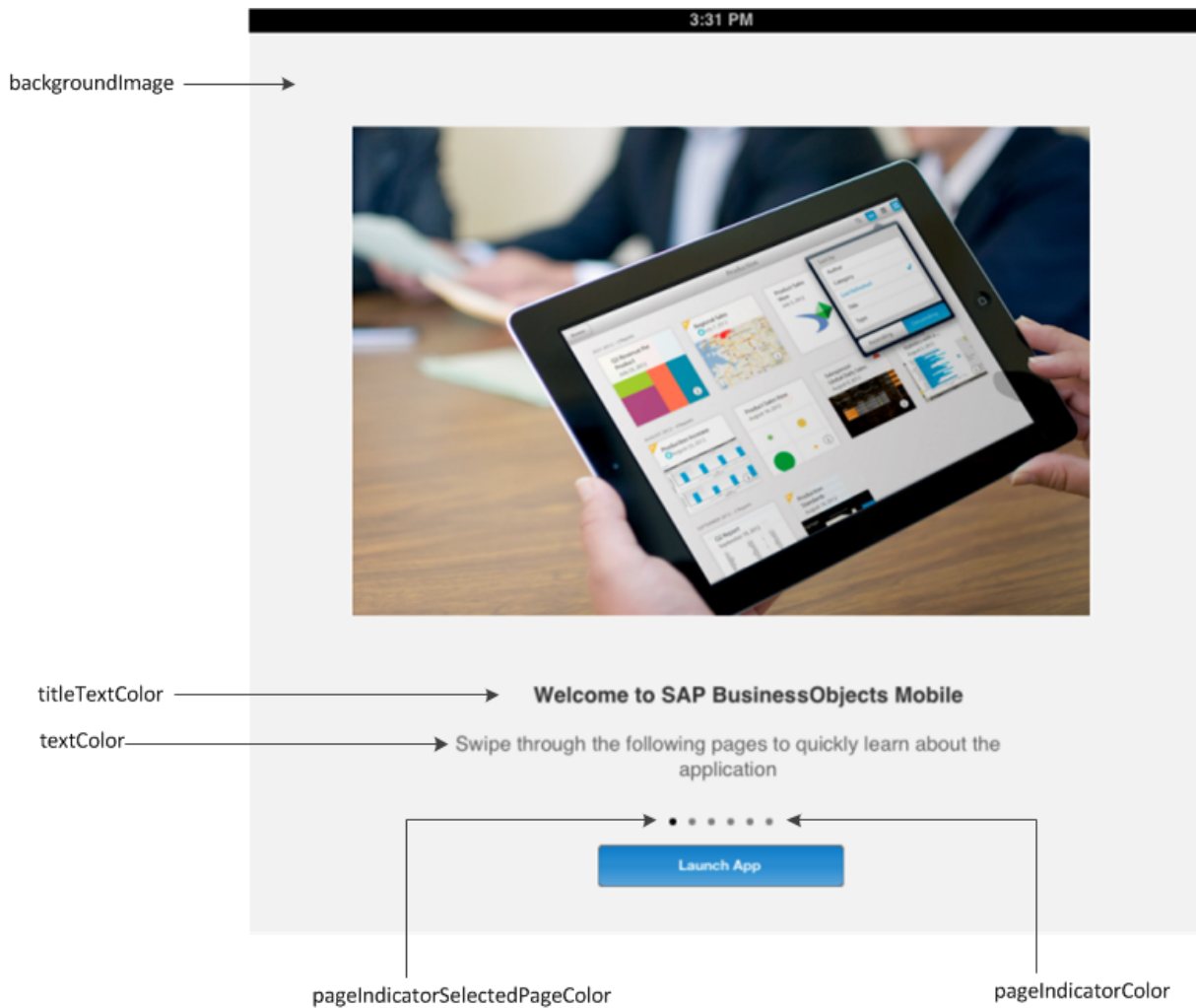
6. Save the changes, run the project, and validate your changes in the application build.

Customizing the Getting Started Slides for Appearance

You can customize the overall appearance of Getting Started by modifying values of some features defined in the `HelpSliderPageResourceFile-iPad.plist` and `HelpSliderPageResourceFile-iPhone.plist` files. The figure below displays those features:

| Key | Type | Value |
|--------------------------------|------------|------------------------|
| ▼ Root | Dictionary | (6 items) |
| backgroundImage | String | getting_started_bg.png |
| titleTextColor | String | 333333 |
| textColor | String | 666666 |
| pageIndicatorColor | String | 7F7F7F |
| pageIndicatorSelectedPageColor | String | 000000 |

The figure below illustrates how each feature above impacts the Getting Started appearance:



After modifying the value of any feature, save the changes, run the project, and validate your changes.

4.5 Customizing the Appearance of the Home Screen

Customizing the Text Colors and Background Colors of the Home Screen

To customize the Home screen (that appears with sample documents when you install and launch the application), follow the procedure below:

1. Open the `ApplicationAppearance.plist` file available at the following location in your project:
▶ `Xcode` ▶ `Resources` ▶ `Bundles` ▶ `SAPBI_Appearance.bundle` ▶

An example snapshot of the file is shown in the figure below:

| Key | Type | Value |
|---|------------|------------|
| ▼ Root | Dictionary | (12 items) |
| navbarTextColor | String | 666666 |
| homeListViewSectionHeaderPrimaryTextColor | String | 666666 |
| homeListViewSectionHeaderSecondaryTextColor | String | 666666 |
| homeGridViewSectionHeaderPrimaryTextColor | String | 666666 |
| homeGridViewSectionHeaderSecondaryTextColor | String | 666666 |
| homeInfoBoxTextColor | String | 666666 |
| homeInfoBoxBGColor | String | FBFBFB |
| homeListViewCellDocFieldTextColor | String | 333333 |
| homeListViewCellDateFieldTextColor | String | 666666 |
| homeListViewCellPersonFieldTextColor | String | 666666 |
| homeListViewCellBGColor | String | FBFBFB |
| homeListViewCellSeparatorColor | String | E6E6E6 |

2. Update the value of features with color codes (in 6 digit hexadecimal value format) based on your custom requirements.
3. Save your changes, run the project, and validate your changes in the simulator.

i Note

The figures below display how each feature in the `ApplicationAppearance.plist` file impacts the Home screen in the **Grid** and **List** views, respectively:





Customizing the Background Images of the Home Screen

To customize the background images for the application's Home screen, follow the procedure below:

1. Using the Finder, access the `SAPBI_Appearance.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Right click on `SAPBI_Appearance.bundle` and choose `Show Package Contents`. All the images belonging to the bundle appear on the screen.
3. Select the image that you want to replace, and replace it with your custom image.

i Note

The name and dimensions of your custom image should be the same as those of the default image, and they are listed in the table below.

| Image name | Image size | Image purpose |
|-----------------------------|---------------|--|
| <code>home_bg.png</code> | *not specific | Image tile used for the background image of the Home screen. (See the figure provided at the bottom for more clarity.) |
| <code>home_bg@2x.png</code> | *not specific | Image tile used for the background image of the Home screen for devices with Retina displays. |

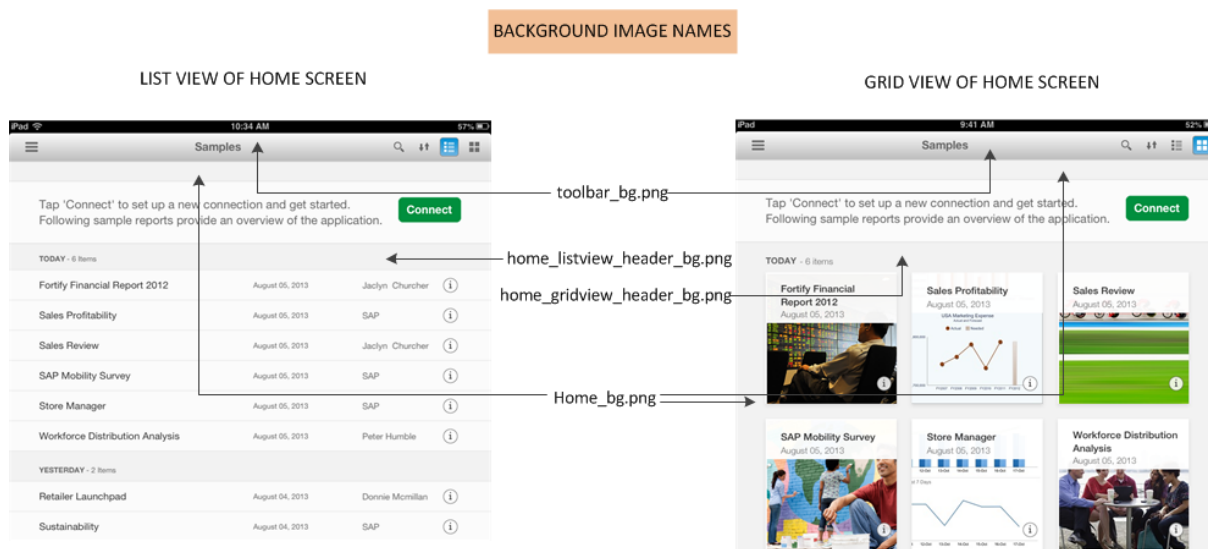
| Image name | Image size | Image purpose |
|-----------------------------|---|---|
| toolbar_bg.png | Height=44 pixels; width can be any value (recommended to keep small) | Image tile used for the background of the Home screen toolbar. |
| toolbar_bg@2x.png | Height=88 pixels; width can be any value (recommended to keep small) | Image tile used for the background of the Home screen toolbar for devices with Retina displays. |
| home_listview_header_bg.png | *not specific | Image tile used for the background of group headers in List view of the Home screen. |
| home_gridview_header_bg.png | *not specific | Image tile used for the background of group headers in Grid view of the Home screen. |

i Note

The toolbar background image must be created without alpha channel (no transparency). Otherwise, the toolbar would overlap the report content on iOS 7 devices. This may also cause the secondary toolbar to not appear completely.

* If the size of the image is smaller than the iPad size, the images get tiled on the device screen. The image should be small in size (e.g. 256*256 for normal screens and 512*512 for Retina displays), so that the Home screen looks good in both landscape and portrait orientations. Large images may saturate the device memory and make the application experience slower.

The figure below illustrates the Home screen sections to which each background image corresponds:



4.6 Customizing the Browse Screen (Left Panel)

You can customize the appearance of the left navigation (Browse) panel of the application including the background images and the font color.

Customizing the Background Images of the Browse Panel (Body) and Toolbar

To customize an image, follow the steps below:

1. Using the Finder, access the `SAPBI_Appearance.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Right click on `SAPBI_Appearance.bundle` and choose *Show Package Contents*. All the images belonging to the bundle appear on the screen.
3. Select the image that you want to replace and replace it with your custom image.

i Note

The name and dimensions of your custom image should be the same as those of the default image, and they are listed in the table below.

Customizing the Font Color of Labels in the Browse Panel

1. Using the Finder, access the `SAPBI_Appearance.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Open the `ApplicationAppearance.plist` file available at the following location in your project.
3. Update the value of the `homeLeftPanelTextColor` property with the required color code (in 6 digit hexadecimal value format).
4. Save your changes, run the project, and validate your changes in the simulator.

Customizing the Font Color of Current Connection in the Header of the Browse Panel

1. Using the Finder, access the `SAPBI_Appearance.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Open the `ApplicationAppearance.plist` file available at the following location in your project.
3. Update the value of the `homeLeftConnNameTextColor` property with the required color code (in 6 digit hexadecimal value format).
4. Save your changes, run the project, and validate your changes in the simulator.

4.7 Customizing Icons Within the Application

This chapter explains how to customize the various icons used within the application (including the toolbar on the Home screen, the Browse panel and open reports).

Customizing the Icons of the Home Screen (Used in Toolbar, Body, BI Document Tiles) and the Browse Panel

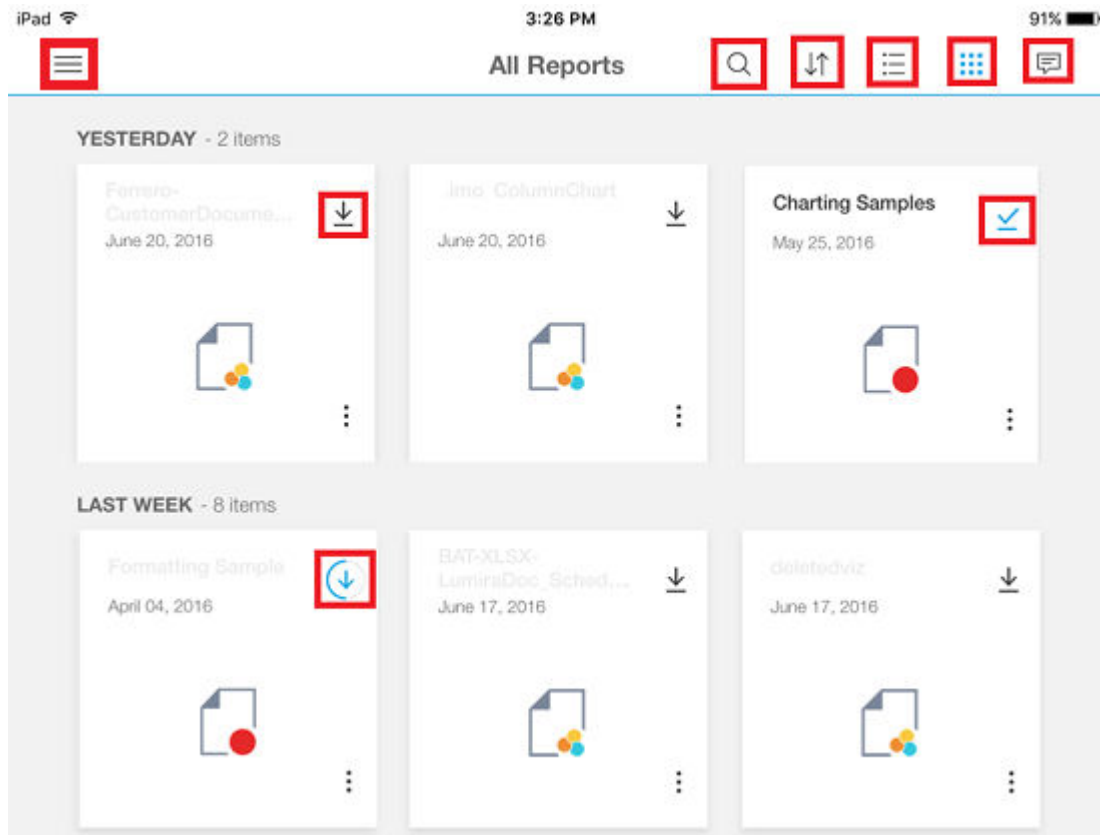
To customize an icon, perform the procedure below:

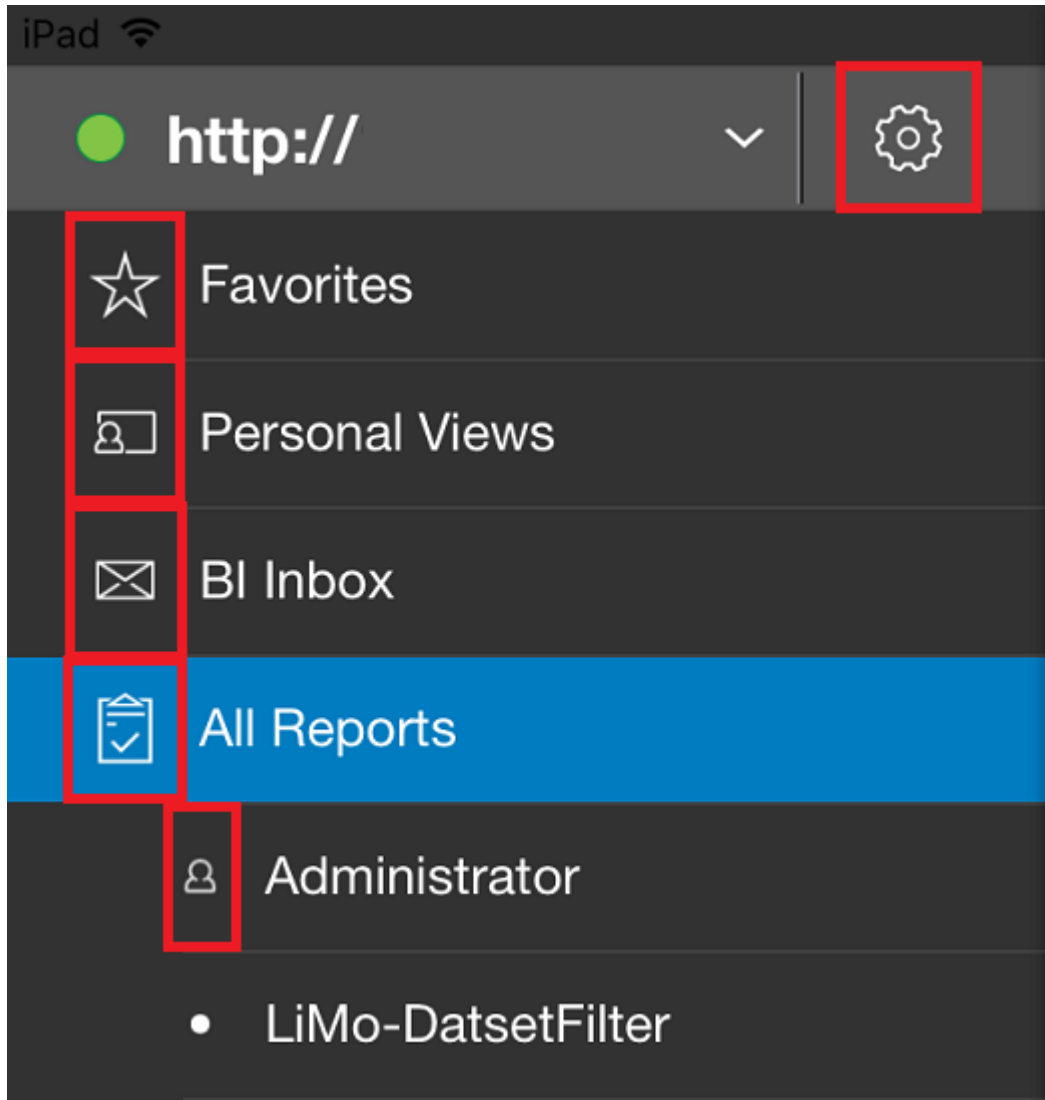
1. Using the Finder, access the `SAPBI_Icons_HomeScreen.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Right click on `SAPBI_Icons_HomeScreen.bundle` and choose *Show Package Contents*. All the images belonging to the bundle appear on the screen.
3. Select the image that you want to replace and replace it with your custom image.

i Note

- The name and dimensions of your custom image should be the same as those of the default image.
- You can identify the purpose of each image icon by looking at its name. For example, `download_list_normal.png` is the image name for a normal Download icon that appears next to a listed BI document in the List View of the Home screen. Similarly, `download_thumb_normal.png` is the unselected download icon appearing over a thumbnail of BI document in Grid View of the screen, `download_thumb_progress.png` is the name of the image that should appear over the tile once the download is in progress, and `download_thumb_completed.png` is the name of the image that should appear once the document is completely downloaded.
- `<ImageName>@2x.png` has the same purpose as `<ImageName>.png`, but is used for devices with **Retina** displays.

The following images display some examples of the icons that you can customize using the images in the `SAPBI_Icons_HomeScreen.bundle`:





Customizing the Icons of BI Reports (Used in Primary and Secondary Toolbars and the Report Body)

To customize an icon, perform the procedure below:

1. Using the Finder, access the `SAPBI_Icons_HomeScreen.bundle` file located at `:MOBISDK/src/xcode/Resources/Bundles/`.
2. Right click on `SAPBI_Icons.bundle` and choose *Show Package Contents*. All the images belonging to the bundle appear on the screen.
3. Select the image that you want to replace, and replace it with your custom image.

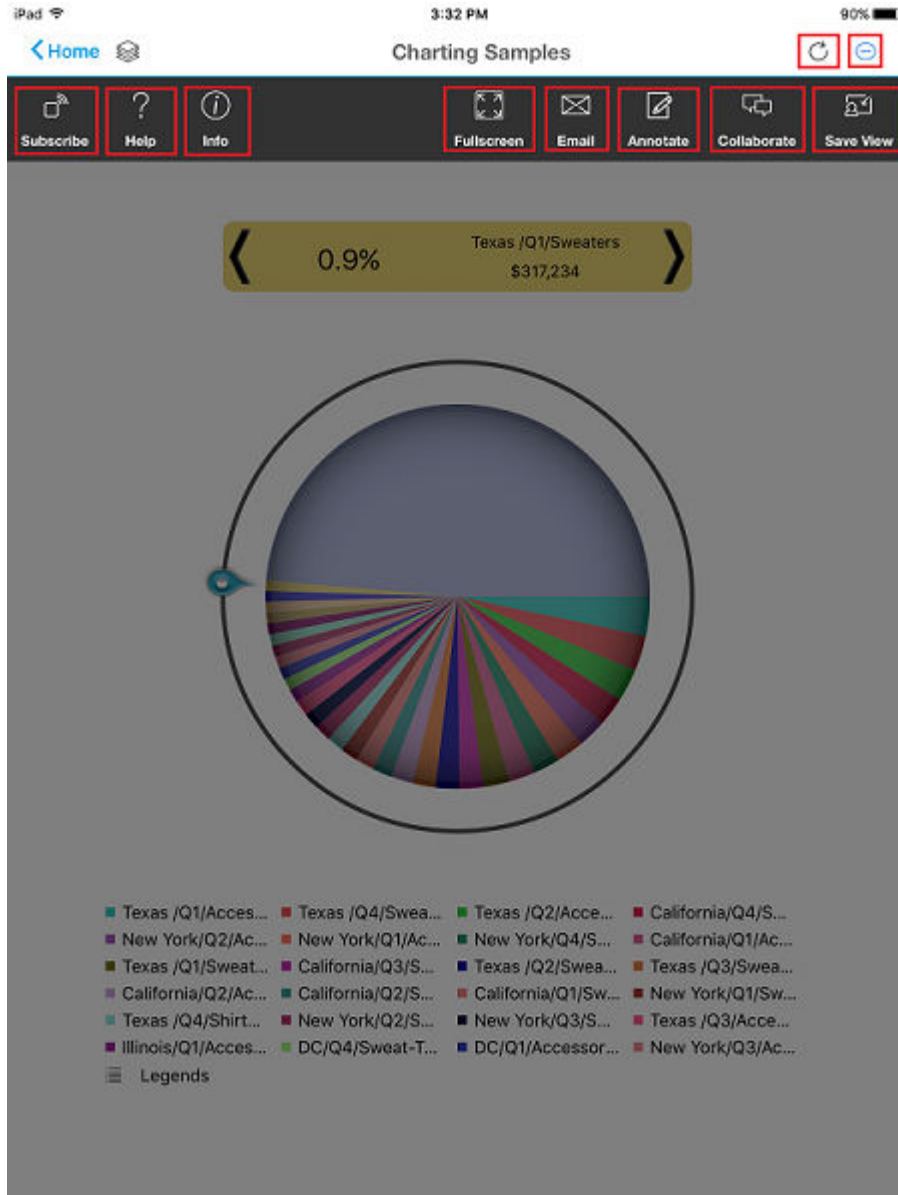
i Note

- The name and dimensions of your custom image should be the same as those of the default image.
- You can identify the purpose of each image icon by looking at its name.

For example, `annotate_crop_normal.png` is the image name for a normal Crop annotation icon that appears in the toolbar of a BI report that the user has locked for annotation. Similarly, `annotate_crop_selected.png` is the Crop icon as it appears when the user has selected it in the toolbar.

- `<ImageName>@2x.png` has the same purpose as `<ImageName>.png`, but is used for devices with **Retina** displays.

The following figure displays some examples of the icons you can customize using the `SAPBI_Icons.bundle`:



5 Customizing the Default Behavior of the Application

The default behavior of the application is defined by various features in the `DefaultSettings.plist` file available at the following location:

► [Xcode](#) ► [Resources](#) ►

An example snapshot of the file is displayed in the figure below:

| Key | Type | Value |
|--|------------|--|
| ▼ Customizations | Dictionary | (29 items) |
| feature.predefinedconnections.enabled.default | String | true |
| ▼ feature.predefinedconnections.list.default | Array | (1 item) |
| Item 0 | String | sapbi://addconnection?ConnectionType=BOEConnection&connection_name=SAP%20Analytics |
| feature.help.enabled.default | String | true |
| feature.help.url.default | String | http://help.sap.com/bomobileios |
| feature.help.video.enabled | String | true |
| feature.help.video.playlistfeed.default | String | http://gdata.youtube.com/feeds/api/playlists/PLs5htBlwERYXvNwU6uYyYCMtRgFbBRMGc |
| feature.get.started.enabled.default | String | true |
| feature.get.started.slides.plist.iphone.default | String | HelpSliderPageResourceFile-iPhone |
| feature.get.started.slides.plist.ipad.default | String | HelpSliderPageResourceFile-iPad |
| feature.addconnection.enabled.default | String | true |
| feature.addconnection.BOE.enabled.default | String | true |
| feature.addconnection.HANABIServer.enabled.default | String | true |
| feature.addconnection.HANABICloud.enabled.default | String | true |
| feature.addconnection.SUP.enabled.default | String | true |
| feature.addconnection.BIOD.enabled.default | String | true |
| feature.SUPconnection.application.id.default | String | MobiApp |
| feature.email.enabled.default | String | true |
| feature.annotation.enabled.default | String | true |
| feature.ipad.hs.tile.date.default | String | modified |
| feature.dashboard.sup.enabled.default | String | false |
| feature.recommend.app.enabled.default | String | true |
| feature.recommend.app.mail.subject.default | String | SAP BusinessObjects Mobile |
| feature.recommend.app.mail.body.default | String | |
| feature.collaboration.enabled.default | String | true |
| feature.streamwork.enabled.default | String | true |
| feature.jam.enabled.default | String | false |
| feature.collaboration.jam.callback.urlverb.default | String | jamcallback |
| feature.home.view.default | String | list |
| ► WebIntelligence | Dictionary | (5 items) |
| ▼ Extenstions | Dictionary | (1 item) |
| ▼ WebIntelligence | Dictionary | (2 items) |
| ▼ Tables | Array | (1 item) |
| ▼ Item 0 | Dictionary | (2 items) |
| blockNamePrefix | String | myprefix_ |
| viewProvider | String | SampleView |
| ▼ Charts | Array | (1 item) |
| ▼ Item 0 | Dictionary | (2 items) |
| viewType | String | chart:sample:sample |
| viewProvider | String | SampleView |

Each feature in the file above impacts the behavior of the application on the device and you can change its default value based on your requirements.

i Note

Some of the names of the features shown in the file above were changed in the current release based on feedback received after the release of previous versions of the SAP BusinessObjects Mobile application.

The sub-sections of this chapter explain how you can specify values for these features to customize various aspects of the application.

5.1 Customizing the Onboarding Features

You can customize the following onboarding features of the application through a metadata-driven approach:

Predefining Connections

- Predefine the following types of connections in the application, so that they appear as readily available connections immediately after the application is installed on the device:
 - Enterprise/ BI Platform
 - SAP Market Place
 - BI OnDemand
- Predefine multiple connections for the application.
- Predefine a default BI document to be opened when the user logs in to the connection.

Disabling Connection Creation

- Disable the creation of any connection in the application.
- Disable the creation of specific connection types in the application:
 - Enterprise/ BI Platform
 - SAP Market Place
 - BI OnDemand

To customize any feature from the list above, follow the procedure below:

1. Open the `DefaultSettings.plist` file from the following location in your Xcode project bundle:
[▶ Xcode ▶ Resources ▶](#)
2. Navigate to the [Customizations](#) section of the `DefaultSettings.plist` file and modify the value of the required feature based on its description provided in the table below this procedure.
3. Save your changes, run the project, and validate your customization in the simulator.

| Feature | Description | Default value |
|---|---|---------------|
| feature.predefinedconnections.enabled.default | <ul style="list-style-type: none">• If the value is "true", the connections listed in feature.predefinedconnections.list.default are created. | true |

| Feature | Description | Default value |
|--|---|---|
| | <ul style="list-style-type: none"> If the value is "false", you cannot have predefined connections for the application. | |
| <i>*feature.predefinedconnections.list.default</i> | <p>You can define a single predefined connection or multiple predefined connections by specifying values within this parameter. It is of the type "Array" and each <i>Item</i> in the array is of the type "String". The value of each item corresponds to a connection URL.</p> <div style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 10px; margin-top: 10px;"> <p>Note</p> <ol style="list-style-type: none"> See the bottom of this chapter for more information on creating predefined connections. This property is meant only for the first installation of the application. Hence if you change the value of any parameter in the URL, you need to remove and install the application again to see the changes reflected in the application. However, if you are using a simulator to test, then you can choose Reset Content and Settings... on the simulator to see your changes reflected in the application. </div> | <p>1 item (connection URL) of type <code><String></code> in the array: <code>sapbi://addconnection?ConnectionType=BOEConnection&connection_name=SAP%20Analytics%20Gallery&server_url=&cms=</code></p> |
| <i>feature.addconnection.enabled.default</i> | <ul style="list-style-type: none"> If the value is 'true', users can add connections in the application in the following ways: <ul style="list-style-type: none"> Importing from the server (using Application Settings) Creating manually (using Application Settings) Using the <code>sapbi://</code> URL provided by the administrator If the value is 'false', the users cannot add connections in the application. | <p>true</p> |
| <i>feature.addconnection.BOE.enabled.default</i> | <ul style="list-style-type: none"> If the value is 'true', users can add BI platform/ Enterprise connections in the application in the following ways: | <p>true</p> |

| Feature | Description | Default value |
|--|---|---------------|
| | <ul style="list-style-type: none"> ○ Importing from the Configuration server (using Application Settings) ○ Manually creating connections of the type <i>SAP BusinessObjects BI Platform</i> (using Application Settings) ○ Using the sapbi:// URL provided by the administrator ● If the value is 'false', the users cannot add BI platform connections in the application. | |
| | <p>i Note</p> <ol style="list-style-type: none"> 1. This feature does not impact the application if <i>feature.addconnection.enabled.default</i> is set to 'false'. 2. This feature does not impact the list of connections provided in <i>feature.predefinedconnections.list.default</i> | |
| <i>feature.addconnection.SUP.enabled.default</i> | <ul style="list-style-type: none"> ● If the value is 'true', users can add SUP server connections in the application in the following ways: <ul style="list-style-type: none"> ○ Manually creating connections of the type <i>Sybase Unwired Platform</i> (using Application Settings) ○ Importing from the Sybase Afaria server ○ Using the sapbi:// URL provided by the administrator ● If the value is 'false', the users cannot add SUP connections in the application. | true |
| | <p>i Note</p> <ol style="list-style-type: none"> 1. This feature does not impact the application if <i>feature.addconnection.enabled.default</i> is set to 'false'. 2. This feature does not impact the list of connections provided in <i>feature.predefinedconnections.list.default</i> | |

| Feature | Description | Default value |
|--|---|---------------|
| <i>feature.application.pwd.minlen.default</i> | This property defines the minimum length of the application password string for the user. | 8 |
| <div style="background-color: #f0f0f0; padding: 10px;"> <p>i Note</p> <ul style="list-style-type: none"> In the current release, this property is used only when the user creates or modifies the application password. If an application is previously installed with a certain value for this property, and you reduce the value in the SDK, it does not have any impact on the password's minimum length. </div> | | |

| | | |
|--|---|-------|
| <i>feature.connection.trustallservers.default</i> | <ul style="list-style-type: none"> If the value is 'true', the application trusts all the HTTPS servers in the network. If the value is 'false' (default), the application only allows connections to the HTTPS servers that have a valid server certificate. | false |
| <div style="background-color: #f0f0f0; padding: 10px;"> <p>i Note</p> <p>We recommend retaining the default value of this parameter.</p> </div> | | |

i Note

To avoid unpredictable results, ensure accuracy of the value that you specify for each feature in the `DefaultSettings.plist` file.

Information for Specifying Values of the [*feature.predefinedconnections.list.default*](#) Feature

Description of Common Parameters Used Across All Predefined Connection Types

| Parameter name | Description |
|------------------------------|-----------------------------|
| <code>connection_name</code> | The name of the connection. |
| <code>connectionType</code> | The type of the connection. |

| Parameter name | Description |
|----------------|--|
| | <p>i Note</p> <p>Below is the list of valid <ConnectionType> names corresponding to each connection type:</p> <ul style="list-style-type: none"> • BOEConnection for a BI platform connection • SUPConnection for an SUP connection • BIOD for a BI OnDemand connection |
| usr | User name for the connection. |
| pwd | Password for the connection. |
| savePwd | Boolean parameter (with 'true'/'false' value) that determines if the option to <i>Save Password</i> appears in the application on the device. |
| | <p>i Note</p> <p>This property is ignored if <code>pwd</code> has not been supplied and <code>editable=false</code></p> |
| editable | Boolean parameter (with 'true'/'false' value) that determines whether the predefined connection is editable on the device. |
| | <p>i Note</p> <p>The following constraints apply if <code>editable=false</code></p> <ul style="list-style-type: none"> • Users cannot view the details of the connection in the application. • When users share a document from the connection, the email draft that the application creates does not contain any links. • Users cannot assign documents to personal categories on the Home screen of the application. • The connection password is retained even when data is wiped off using the <i>Application Settings</i>. |

Creating a Predefined BI platform/ BusinessObjects Enterprise Connection

Syntax:

sapbi://addconnection?<query string parameters>

Mandatory Parameters:

- Connection name
- Connection type
- Server URL

- CMS
- Authentication type

Optional Parameters

- editable
- default
- user credentials

Description and Example

- `server_url`- mobile server address
- `CMS`- CMS address
- `authType`- one of `secEnterprise|secWinAD|secLDAP|secSAPR3`

`sapbi://addconnection?`

```
ConnectionType=BOEConnection&connection_name=myBOEConnection&server_url=11.22.33.44
%3A8080&cms=11.22.33.44&authType=secEnterprise&usr=myuser&pwd=myspassword&savePwd=true&editable=true
```

Creating a Predefined Sybase Unwired Platform Connection

Syntax:

`sapbi://addconnection?<query string parameters>`

Mandatory Parameters:

- Connection name
- Connection type
- Server
- Port
- Farm ID
- Sec Config

Optional Parameters

- editable
- default
- user credentials

Description and Example

- `server_url`- SUP server address/ Relay server address
- `SUP_port`- SUP port/ relay server port
- `Sec_Config`- Security Configuration name
- `SUP_Farm_ID`- Specify if relay server is not used

```
sapbi://addconnection?ConnectionType=SUPConnection&connection_name=SUP
%20Connection&server_url=55.66.77.88&SUP_Port=5001&SUP_Farm_ID=0&Sec_Config=MobiTes
t&Application_ID=MobiApp&usr=myuser&pwd=myspassword&savePwd=true&editable=true
```

Creating a Predefined Connection Specifying a Document that Opens When the Connection is Chosen on the Device

Syntax:

sapbi://setDefaultDoc?<query string parameters>

Mandatory Parameters:

- iDocId (Document CUID)
- Type
- Parameters as required by the appropriate connection type

i Note

The supported document types are as follows:

- PDF
- Web Intelligence
- Zen
- Crystal Reports
- Xcelsius
- Hyperlink

Optional Parameters

Parameters as required by the appropriate connection type

i Note

- The Parameters are case-sensitive.
- It is mandated for you to add the Mandatory Parameters and pass valid values to the Mandatory Parameters. Failing to do which may cause an application error.

Description and Example

- Use an `addconnection` URL as appropriate
- Replace `addconnection` verb with `setDefaultDoc`
- Introduce the `iDocId` parameter that specifies the CUID of the document that should open when the connection is chosen on the device

sapbi://**setDefaultDoc?**

```
ConnectionType=BOEConnection&connection_name=myBOEConnection&server_url=11.22.33.44%3A8080&cms=11.22.33.44&authType=secEnterprise&iDocId=123456789&usr=myuser&pwd=mypassword&savePwd=true&editable=true&type=webi
```

i Note

If you are specifying this URL, you do not need to use `addconnection` explicitly.

Related Information

[Configuring the Behavior of the Application by Setting Properties on the Server \[page 97\]](#)

5.2 Customizing the Application Functionality

You can customize the following aspects of application functionality through a metadata-driven approach:

Customizing Help, Video and Other Features in the Application

- Disable Help
- Customize the application Help and BI content specific contextual Help to custom HTML pages
- Disable Video Tutorials
- Modify the video tutorials URL to your chosen Youtube playlist
- Disable Getting Started
- Customize the Getting Started content/screens
- Customize the first home screen view type (**List/Grid**) that the users see after installing the application on the device
- Enable or disable the + (Add to device) option that appears for BI documents in the application

Customizing Usability and Security Features

- Enable or disable the display of BI document samples on the Home screen of the application
- Enable or disable the option to generate BI document samples (for Web Intelligence and Dashboards) in the application
- Enable or disable the appearance of the BI document's author name on the following screens of the application:
 - Home screen (where BI documents are listed)
 - BI document Information screen
 - Sort pop up (invoked from the application toolbar)
- Customize the email ID on which you receive crash reports from the application
- Enable or disable the QR code scanning in the application
- Specify the number of days between the first prompt to change the BI connection password and the expiration of the password.
- Customize the required minimum length of the application password

Customize Collaboration


- Disable all collaboration options
- Specify a Web-based collaboration tool (*SAP JAM*)
- Disable *Email* and *Annotation*



Customizing the *Recommend This Application* Feature

- Disable *Recommend This Application* which appears in **Settings > Additional Settings** of the application
- Customize the subject and body of the email draft generated by tapping the *Recommend This Application* option

To customize any feature from the list above, follow the procedure below:

1. Open the `DefaultSettings.plist` file from the following location in your Xcode project bundle:
Xcode > Resources
2. Navigate to the *Customizations* section of the file and modify the value of the required feature based on its description provided in the table below this procedure.
3. Save your changes, run the project, and validate your changes in the simulator.

| Feature | Description | Default value |
|---|---|---|
| <code>feature.help.enabled.default</code> | <ul style="list-style-type: none"> • If the value is 'true', the <i>Help</i> is enabled in the application. • If the value is 'false', the <i>Help</i> does not appear in the application on the device. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>i Note</p> <p>If the value is 'true', you can further use the device settings (<i>SAP BI</i> settings -> <i>Show Help</i>) to govern the visibility of Help.</p> </div> | true |
| <code>feature.help.url.default</code> | <p>The value of this feature determines the generic application Help page that opens on tapping  (Settings) > Help in the Browse panel on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <pre>http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_iPadGe nHelp_[LANG]/ index.html</pre> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>Example</p> <ul style="list-style-type: none"> • Enter [LANG- </div> |

| Feature | Description | Default value |
|--|---|--|
| <i>feature.help.url.iPhone.default</i> | <p>The value of this feature determines the generic application Help page that opens on tapping  (Settings) > Help > in the Browse panel on the iPhone device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <pre>COUNTRY]=en-US • Enter [LANG]= en</pre> <hr/> <pre>http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPhone/ mobiOS_iPhone GenHelp_[LANG]/index.html</pre> |
| <i>feature.help.url.webi.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings) > Help > in an open Web Intelligence report on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <div data-bbox="1195 927 1394 1263" style="border: 1px solid #0070C0; padding: 5px;"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US • Enter [LANG]=en </div> <hr/> <pre>http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_WebIiP adHelp_[LANG] / 3d33b18d647c4 373836ec5ff5b 07d817.html</pre> <div data-bbox="1195 1724 1394 1926" style="border: 1px solid #0070C0; padding: 5px;"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US </div> |

| Feature | Description | Default value |
|---|---|---|
| <i>feature.help.url.webi.iPhone.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings)  Help  in an open Web Intelligence report on the iPhone device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <div data-bbox="1198 353 1394 483" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <ul style="list-style-type: none"> • Enter [LANG]=en </div> <pre>http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPhone/ 61_mobiOS_Web IiPhoneHelp_[LANG]/ 3d33b18d647c4 373836ec5ff5b 07d817.html</pre> <div data-bbox="1198 949 1394 1279" style="border: 1px solid #ccc; padding: 5px;"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US • Enter [LANG]=en </div> |
| <i>feature.help.url.exp.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings)  Help  in an open Explorer document on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <pre>http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_Explor eriPadHelp_[L ANG]/ 19bf70d94de04 4e49cf4a2bd8c 3f8770.html</pre> <div data-bbox="1198 1756 1394 1951" style="border: 1px solid #ccc; padding: 5px;"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US </div> |

| Feature | Description | Default value |
|--|---|--|
| <i>feature.help.url.exp.iPhone.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings)  in an open Explorer document on the iPhone device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <div data-bbox="1198 353 1394 483"> <ul style="list-style-type: none"> • Enter [LANG]=en </div> <pre data-bbox="1195 506 1394 925">http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPhone/ Explorer_Mobi _iPhone_[LANG]/ 19bf70d94de04 4e49cf4a2bd8c 3f8770.html</pre> <div data-bbox="1198 949 1394 1281"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US • Enter [LANG]=en </div> |
| <i>feature.help.url.cr.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings)  in an open Crystal Reports document on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <pre data-bbox="1195 1308 1394 1693">http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_CRiPad Help_[LANG]/ 07c58dde36084 9519059660d81 4e1177.html</pre> <div data-bbox="1198 1718 1394 1915"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US </div> |


| Feature | Description | Default value |
|--|--|---|
| <i>feature.help.url.lumira.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings) > Help > in an open Lumira visualization on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <div data-bbox="1198 353 1394 483"> <ul style="list-style-type: none"> • Enter [LANG]=en </div> <pre data-bbox="1195 506 1394 920">http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_Lumira iPadHelp_[LAN G]/ 462bf122b5974 e6087d45a4441 36ea32.html</pre> <div data-bbox="1198 949 1394 1279"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US • Enter [LANG]=en </div> |
| <i>feature.help.url.db.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings) > Help > in an open Dashboards document on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <pre data-bbox="1195 1308 1394 1722">http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_Dashbo ardsHelp_[LAN G]/ e498d39c88324 754bedf22c69f 9ce89a.html</pre> <div data-bbox="1198 1751 1394 1946"> <p>❖ Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US </div> |

| Feature | Description | Default value |
|---|--|---|
| <i>feature.help.url.analysis.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings) > Help > in an open Analysis document on the iPad device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <div data-bbox="1198 353 1394 483"> <ul style="list-style-type: none"> • Enter [LANG]=en </div> <pre data-bbox="1195 506 1394 925">http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPad/ mobiOS_Analys isiPadHelp_[L ANG]/ 38fb41a0fdf34 baf8f3472e738 bd5cfc.html</pre> <div data-bbox="1198 947 1394 1283"> <p>Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US • Enter [LANG]=en </div> |
| <i>feature.help.url.analysis.iphone.default</i> | <p>The value of this feature determines the Web Intelligence Help page that opens on tapping  (Report Settings) > Help > in an open Analysis document on the iPhone device. You can replace the default URL with the path of your custom DHTML Help page.</p> | <pre data-bbox="1195 1312 1394 1727">http:// help.sap.com/ download/ multimedia/ mobi/[LANG- COUNTRY]/ iPhone/ mobiOS_Analys isiPhoneHelp _[LANG]/ 38fb41a0fdf34 baf8f3472e738 bd5cfc.html</pre> <div data-bbox="1198 1749 1394 1951"> <p>Example</p> <ul style="list-style-type: none"> • Enter [LANG-COUNTRY]=en-US </div> |

| Feature | Description | Default value |
|---|--|---|
| | | <ul style="list-style-type: none"> Enter [LANG] = en |
| feature.help.video.enabled.default | <ul style="list-style-type: none"> If the value is 'true', the Video Tutorials option is enabled in the application. If the value is 'false', the Video Tutorials option does not appear in the application. | true |
| | <p>i Note</p> <p>The application supports videos only on iPad devices (not on the iPhone).</p> | |
| feature.help.video.playlistfeed.default | The value of this feature is a URL that provides a link to a Youtube playlist feed. | playlist feed URL |
| feature.get.started.enabled.default | <ul style="list-style-type: none"> If the value is 'true', the Getting Started screens appear on the device when the application is launched after installation. If the value is 'false', the Getting Started screens do not appear on the device when the application is launched after installation. | |
| feature.get.started.slides.plist.iphone.default | This feature helps you customize the Getting Started screens (for content, title or sub-title) for the iPhone device. See the related topic of this chapter for more information. You do not need to change the value of this feature. | HelpSliderPageResourceFile-iPhone.plist |
| feature.get.started.slides.plist.ipad.default | This feature helps you customize the Getting Started screens (for content, title or sub-title) for the iPad device. See the related topic of this chapter for more information. You do not need to change the value of this feature. | HelpSliderPageResourceFile-iPad.plist |
| feature.email.enabled.default | <ul style="list-style-type: none"> If the value is 'true', the Email option appears for the BI documents opened in the application. If the value is 'false', the Email option does not appear for the BI documents opened in the application. | true |
| feature.annotation.enabled.default | <ul style="list-style-type: none"> If the value is 'true', the Annotate option appears for the BI documents opened in the application. If the value is 'false', the Annotate option does not appear for the BI documents opened in the application. | true |
| | <p>i Note</p> <p>If you disable both feature.email.enabled.default and feature.collaboration.enabled.default, the application ignores the feature.annotation.enabled.default property.</p> | |
| feature.recommend.app.enabled.default | Using this feature, you can disable the Recommend This Application option in the additional settings of the application. | true |

| Feature | Description | Default value |
|--|--|----------------------------------|
| feature.recommend.mail.subject.default | The value specified for this feature determines the default <i>Subject</i> (that appears pre-populated) in the email draft generated by the application on tapping the <i>Recommend This Application</i> setting. | SAP BusinessObjects Mobile |
| feature.collaboration.enabled.default | <ul style="list-style-type: none"> If the value is 'true', the <i>Collaborate</i> option appears in the secondary report toolbar of a BI report opened on the device. If the value is 'false', the <i>Collaborate</i> option does not appear in the secondary report toolbar of a BI report opened on the device. <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p>i Note</p> <p>If you set the property listed below as 'false', the application ignores the collaboration property:</p> <ul style="list-style-type: none"> feature.jam.enabled.default </div> | true |
| feature.jam.enabled.default | <ul style="list-style-type: none"> If the value is 'true', <i>SAP JAM</i> popup appears asking for user credentials when users choose <i>Collaborate</i> in the secondary report toolbar of a BI report opened on the device If the value is 'false', <i>SAP JAM</i> does not appear as the Web-based collaboration tool when users choose <i>Collaborate</i> in the secondary report toolbar of a BI report opened on the device. <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p>i Note</p> <ul style="list-style-type: none"> If feature.collaboration.enabled.default is set to 'false', feature.jam.enabled.default is not considered in the application. </div> | false |
| feature.jam.url.default | JAM URL | <none specified> |
| | <div style="background-color: #f0f0f0; padding: 10px;"> <p>i Note</p> <p>The value of this feature must be specified to make SAP JAM work in the application.</p> </div> | |
| feature.jam.consumer.key.default | JAM consumer key | <none specified> |
| | <div style="background-color: #f0f0f0; padding: 10px;"> <p>i Note</p> <p>The value of this feature must be specified to make SAP JAM work in the application.</p> </div> | |
| feature.jam.consumer.secret.default | JAM consumer secret | <none specified> |
| | <div style="background-color: #f0f0f0; padding: 10px;"> <p>i Note</p> <p>The value of this feature must be specified to make SAP JAM work in the application.</p> </div> | |

| Feature | Description | Default value |
|--|---|---------------|
| <i>feature.jam.callback.urlverb.default</i> | JAM Callback URL Verb | jamcallback |
| | <p>i Note</p> <p>The value of this feature must be specified to make SAP JAM work in the application.</p> | |
| <i>feature.home.view.default</i> | <p>The value of this feature determines the first home screen view that the users see after installing the application on the device. Afterward, the view type last used is persisted in the application.</p> <p><i>list</i> and <i>grid</i> are the two valid values for this feature.</p> | list |
| <i>feature.home.view.types.default</i> | <p>The value of this feature determines the default view types first the home screen view that the users see after installing the application on the device. Afterward, the view type last used is persisted in the application.</p> <p><i>list</i> and <i>grid</i> are the two valid values for this feature.</p> <ul style="list-style-type: none"> • If you choose <i>list</i> value, application displays only list mode. • If you choose <i>grid</i> value, application displays only grid mode. | list |
| <i>feature.application.pwd.minlen.default</i> | This property defines the required minimum character length of the application password that users have to fulfill while creating the password in the application. | 8 |
| <i>feature.connection.trustallservers.default</i> | | true |
| <i>feature.background.task.execution.allowed.default</i> | This property is used for internal purposes only. | true |
| <i>feature.documents.add.to.device.enabled.default</i> | <ul style="list-style-type: none"> • If the value is 'true', the + (<i>Add to device</i>) option appears for BI documents on following the screens of the application: <ul style="list-style-type: none"> ○ Application Home screen (List view and Grid View) ○ Secondary toolbar in open reports ○ Information (>) screen of a BI document • If the value is 'false', the <i>Add to device</i> option does not appear in the application. | true |
| <i>feature.home.sort.option.default</i> | <p>Use this property to define the sort parameter according to which the BI documents appear on the application's Home screen. The following values are valid for this property:</p> <ul style="list-style-type: none"> • author • date • title • type | date |

| Feature | Description | Default value |
|--|---|----------------------------|
| <i>feature.appurlscheme.credentials.enabled.default</i> | Enable this property to ensure that the user credentials (Username and Password) are automatically included in the connection URLs. | false |
| <i>feature.allow.sample.generation.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application provides the Generate Samples option on the Information(>) screen for Web Intelligence and Dashboards documents. If the value of this property is 'false', the application does not provide any option to generate samples for any BI document.. | false |
| <i>feature.show.samples.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application displays sample BI documents on the Home screen after the application is installed and accessed for the first time with no server connection. If the value of this property is 'false', the application does not display the sample BI documents on the Home screen.. | true |
| <i>feature.qrcode.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application displays the Scan QR code option within  (Settings). If the value of this property is 'false', the application does not provide the ability to scan QR codes. | true |
| <i>feature.boe.pwd.expiry.alert.before.default</i> | This property defines the number of days between the first prompt to change the BI connection password and the expiration of the password. | 5 |
| <i>feature.crash.report.mailid.default</i> | This property defines the email ID at which users receive the crash reports sent from the application. | SAP_mobilebi.crash@sap.com |
| <i>feature.home.doc.person.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application displays the Author name next to the BI documents on the list view of the Home screen, and also displays the Sort by: Author option in the Sort pop up (invoked from toolbar). If the value of this property is 'false', the application neither displays the Author name next to the BI documents on the list view of the Home screen, nor does it display the Sort by: Author option in the Sort pop up (invoked from toolbar). | true |
| <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>i Note</p> <p>If this property is disabled, the names of the Sender for BI Inbox Documents and of the Owner for Lumira documents also do not appear in the application.</p> </div> | | |
| <i>feature.info.doc.person.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application displays the Author name on the Information screen of the BI document . | true |

| Feature | Description | Default value |
|--|--|---------------|
| | <ul style="list-style-type: none"> If the value of this property is 'false', the application does not display the <i>Author</i> name on the Information screen of the BI document . | |
| <i>feature.home.bi.inbox.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application displays the <i>BI Inbox</i> in the left navigation panel of the application . If the value of this property is 'false', the application does not display the <i>BI Inbox</i> in the left navigation panel of the application . | true |
| <i>feature.home.favorites.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the application displays the <i>Favorites</i> in the left navigation panel of the application . If the value of this property is 'false', the application does not display the <i>Favorites</i> in the left navigation panel of the application . | true |
| <i>feature.notification.polling.interval.default</i> | <p>The polling interval is the time (in minutes) after which the application is automatically refreshed on the device.</p> <ul style="list-style-type: none"> If you set this property to zero or a negative value, auto-refresh is disabled in the application. The value of this setting can be in decimals. However, it is recommended that you do not specify a very small value for this setting as frequent auto-refresh of the Home screen puts load on the server when user traffic is high. | 15 |
| <i>feature.autoupdate.enabled.default</i> | <ul style="list-style-type: none"> If the value of this property is 'true', auto update (for BI documents) is enabled in the application by default. If the value of this property is 'false', auto update (for BI documents) is disabled in the application by default. Users can enable this setting using the SAP BusinessObjects Mobile application Settings on their iOS devices. <div style="border: 1px solid #0070c0; padding: 5px; margin-top: 10px;"> <p>i Note</p> <p>"Auto update" implies that the application automatically updates the downloaded BI documents by re-downloading them on the Home screen, when the Home screen is refreshed and document updates are available on the server.</p> </div> | true |
| <i>feature.dashboards.prefer.cache.enabled</i> | <ul style="list-style-type: none"> If the value of this property is 'true', users can enable/disable the <i>Prefer Cache</i> option using the "Browse" panel of the application on device. In the <i>Fiori</i> theme, you can access the <i>Prefer Cache</i> option in the <i>Dashboard</i> section by navigating to ▶ <i>Settings</i> ▶ <i>Additional Settings</i> ▶ <i>Document Settings</i> ▶ <i>Prefer Cache</i> ▶. | true |

| Feature | Description | Default value |
|---|--|---------------|
| | <p>i Note</p> <p>The Fiori theme is not supported in the SDK version of the application.</p> <p>When this option is enabled, the application loads the downloaded Dashboard documents from the device cache memory, even when the application is connected to a Mobile BI server.</p> <ul style="list-style-type: none"> If the value of this property is 'false', the <i>Prefer Cache</i> option is not active in the "Browse" panel of the application | |
| <i>feature.home.folder.enabled</i> | <ul style="list-style-type: none"> If the value of this property is 'true', the document folder information is displayed on tapping the <i>More Information</i> icon. | false |
| | <p>i Note</p> <p>This feature is not available by default. You can add it based on your needs.</p> | |
| <i>feature.disable.screenshot.capture</i> | <p>This feature is applicable for Discovery documents in the HTML mode and Designer documents.</p> <p>When you set the value to true, it disables screenshot capture.</p> <p>i Note</p> <p>This value is set to false by default.</p> | |

Customizing the Web Intelligence Features in the Application

You can customize the following Web Intelligence features of the application:

- Disable the report title display
- Disable *Refresh* for Web Intelligence documents and personal views of Web Intelligence reports
- Configure the colors to be displayed in tables and blank cells
- Control highlighting of linked elements in the table that acts as the source of report element linking
- Customize the default color palettes in Web Intelligence report parts
- Customize the default color theme of Web Intelligence documents on the device
- Enable or disable the report-by-report downloading (and viewing) of Web Intelligence documents on the device


To customize any feature from the list above, follow the procedure below:

1. Open the `DefaultSettings.plist` file from the following location in your Xcode project bundle:

► [Xcode](#) ► [Resources](#) ►

2. Navigate to the *Web Intelligence* section of the file and modify the value of the required features based on their description provided in the table below this procedure.
3. Save the changes in the file, run the project, and validate your changes in the application build on the simulator.

| Feature | Description | Default value |
|---|---|---------------|
| <i>feature.webi.refresh.enabled.default</i> | You can enable or disable <i>Refresh</i> in Web Intelligence reports on the device using this parameter. | true |
| <i>feature.webi.reportTitle.enabled.default</i> | When viewing Web Intelligence reports in the application, the report title is displayed in the bar below the document name. <ul style="list-style-type: none"> • If the value of this feature is "true", the report title appears below the document name in the application. • If the value of this feature is set as "false", the report title does not appear below the document name in the application, thereby providing you more screen area for display. | true |
| <i>feature.webi.rel.legacy.mode.default</i> | This feature impacts how "Report Element Linking" (REL) is visualized in tables in Web Intelligence documents viewed using the application. In previous versions of the application, when REL was enabled from a table to other report blocks, the source table displayed every cell in blue color. Also, each cell in the source table was tappable. <p>In the current release, only the cells in the column on which REL is enabled appear in blue and are tappable. If you need to use the previous behavior, set the value to "true".</p> | false |
| <i>feature.webi.default.palette.enabled.default</i> | When viewing charts in Web Intelligence documents, the application uses a Mobile-specific palette to display the colors if the Web Intelligence document is defined with 'default palette'. | false |

| Feature | Description | Default value |
|---|--|---|
| | <ul style="list-style-type: none"> If you set this value as "true", the Mobile-specific colors are ignored, and the values defined in Web Intelligence will be used even if it uses default palette. If you set this value as "false" (and the Web Intelligence document is defined with 'default palette'), the Mobile-specific colors are used to display charts and the values defined in Web Intelligence are ignored. | |
| feature.webi.theme.default | <p>This property defines the default color theme (Ivory/Graphite) for viewing Web Intelligence documents on the device. Valid values are <code>WhiteTheme</code> and <code>BlackTheme</code>. These correspond to the Ivory and Graphite themes, respectively.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>i Note</p> <p>User can use the Application Settings to override the color theme.</p> </div> | WhiteTheme |
| feature.webi.view.report.by.report.default | <p>This property is used to enable report-by-report downloading of Web Intelligence documents. If this property value is 'false', the entire Web Intelligence document is downloaded from the server when users tap</p> <div style="text-align: center; margin-top: 10px;">  <p>(Download).</p> </div> | false |
| feature.webi.personal.view.update.enabled.default | <p>This property defines if users can update their Personal Views (with data from the server) using the Update option in the view toolbar</p> | true |
| IgnoredColors/CellTextColors/Item (n) | <p>For this feature, you specify or modify the Web Intelligence tablecell text colors that should be ignored by the application. Since the application maps the</p> | <p>Item 0: 000000</p> <p>Item 1: 333333</p> |

| Feature | Description | Default value |
|--|--|--|
| | <p>cell text colors used in Web Intelligence to specific colors on the device screen, not ignoring these colors means that they appear in their original form on the device screen (without any conversion).</p> <p>Colors should be in the 6 digit RGB value format in hexadecimal.</p> | |
| <p><i>IgnoredColors/ FreecellTextColors/Item (n)</i></p> | <p>For this feature, you specify the blank (free) cell text colors that should be ignored by the application. Since the application maps the cell text colors used in Web Intelligence to specific colors on the device screen, not ignoring these colors means that they appear in their original form on the device screen (without any conversion).</p> <p>Colors should be in the 6 digit RGB value format in hexadecimal.</p> | <p>Item 0: 000000 Item 1: ffffff</p> |
| <p><i>IgnoredColors/ CellBackgroundColors/Item (n)</i></p> | <p>For this feature, you specify the table cell background colors that should be ignored by the application. Since the application maps the cell text colors used in Web Intelligence to specific colors on the device screen, not ignoring these colors means that they appear in their original form on the device screen (without any conversion).</p> <p>Colors should be in the 6 digit RGB value format in hexadecimal.</p> | <p>Item 0: f0f0f4 Item 1: ffffff</p> |
| <p><i>IgnoredColors/ HeaderCellBackgroundColors/Item (n)</i></p> | <p>For this feature, you specify the table header cell background colors that should be ignored by the application. Since the application maps the cell text colors used in Web Intelligence to specific colors on the device screen, not ignoring these</p> | <p>Item 0: 5175b9 Item 1: ffffff</p> |

| Feature | Description | Default value |
|---------|---|---------------|
| | colors means that they appear in their original form on the device screen (without any conversion). | |
| | Colors should be in the 6 digit RGB value format in hexadecimal. | |

i Note

1. If the following features have a value defined in the `client` settings on the Mobile server, it overrides the default value which you specify in `DefaultSettings.plist`:
 - `feature.email.enabled`
 - `feature.collaboration.enabled`
 - `feature.annotation.enabled`
 - `feature.jam.enabled`
 - `feature.jam.url`
 - `feature.jam.consumer.key`
 - `feature.jam.consumer.secret`
 - `feature.jam.callback.urlverb`
 - `feature.help.enabled`
 - `feature.help.url`
 - `feature.help.video.enabled`
 - `feature.help.video.playlistfeed`
 - `feature.webi.refresh.enabled`
 - `feature.webi.reportTitle.enabled`
 - `feature.rel.legacy.mode`
 - `feature.webi.default.palette.enabled`
 - `feature.dashboard.sup.enabled`
 - `feature.qrcode.enabled`
 - `feature.show.samples.enabled`
 - `feature.webi.personal.view.update.enabled`
 - `feature.home.doc.person.enabled`
 - `feature.info.doc.person.enabled`
 - `feature.dashboards.prefer.cache.enabled`
 - `feature.disable.screenshot.capture`

See the related topic of this chapter for more information.
2. To avoid unpredictable results, ensure accuracy of the value that you specify for each feature in the `DefaultSettings.plist` file.

Related Information

[Configuring the Behavior of the Application by Setting Properties on the Server \[page 97\]](#)

[Customizing Getting Started \[page 13\]](#)

[Configuring Document-Specific Help \[page 100\]](#)

[Customizing Getting Started \[page 13\]](#)

5.3 Customizing the Protocols

You can customize the following application protocols using a metadata-driven approach:

- Change the SUP application ID
- Enforce the data refresh in Dashboards to pass through the SUP infrastructure
- Customize the SAP BI URLs for connections and documents

i Note

You can change the `sapbi://` part of the SAP BI URLs to some other label (URL scheme) based on your specific enterprise requirements.

To customize the first two features in the list above, open the `DefaultSettings.plist` file from the following location in your Xcode project bundle:

► [Xcode](#) ► [Resources](#) ►

Then modify the value of the relevant feature from the list shown in the table below. After modifying values, save your changes and run the project.

| Feature | Description | Default value |
|---|--|---------------|
| Customizations/feature.SUPconnection.application.id | This feature is meant to override the SUP application name. (If you use SUP connections, the application name always used for SUP is "MobiApp".) | MobiApp |

i Note

If the application specifying a new application ID is installed before the user deregisters from a connection containing an old application ID, the users must kill the application after deregistering and before registering in the new application.

| Feature | Description | Default value |
|--|--|---------------|
| Customizations/feature.dashboard.sup.enabled | <p>This feature impacts how the data connectivity works for Dashboard documents, when viewing it over an SUP connection.</p> <ul style="list-style-type: none"> • If the value is "true", the data connectivity for Dashboards happens through the SUP server. • If the value is "false", the data connectivity for Dashboards does not happen through the SUP server; the application directly connects to the Web application server over HTTP/ HTTPS. | false |

i Note

This value must be set to 'true' only if your SUP server version is 2.2 SPO2 and above.

To customize the URL schemes, perform the steps below:

1. Open the `SAPBusinessObjectsMobile-Info.plist` file in Xcode.
2. Access *Information Property List/ URL Types/ Item(n)/URL Schemes* and set `Item(0)= <your custom URL scheme>`
{By default `Item(0)` is set as "sapbi" and `Item(1)= "sapbiafaria"}`
3. Save your changes and run the project.

i Note

1. If the following feature has a value defined in the `client settings` on the Mobile server, it overrides the default value which you specify in `DefaultSettings.plist`:
 - `feature.dashboard.sup.enabled`
See the related topic of this chapter for more information.
2. To avoid unpredictable results, ensure accuracy of the value that you specify for each feature in the `DefaultSettings.plist` file.

Related Information

[Configuring the Behavior of the Application by Setting Properties on the Server \[page 97\]](#)

5.4 Customizing the Sample BI Documents

This chapter explains how to customize the sample BI documents that appear on the Home screen when you first install the application.

Replacing Default Sample Documents with Your Custom Sample Documents

1. Launch Xcode and access
▶ [Xcode](#) ▶ [Resources](#) ▶ [DefaultSettings.plist](#) ▶
2. In the `DefaultSettings.plist` file, set `feature.allow.sample.generation.enabled.default = true`. This enables the option to generate BI document samples in the application.
3. Save and run the project.
4. In the application (on simulator), log on to a server connection, and click on the (>) [Information](#) icon that appears on the document.
You find the [Generate Samples](#) option on the Information screen.

i Note

- The option to generate samples appears only for Web Intelligence and Dashboards documents.
- The option to generate samples does not appear if the document is set as confidential (secure) or if the `offlinestorage` property is set to `false` on the Mobile server.

5. Click on [Generate Samples](#) on the Information screen of all documents that you want to use as samples. This downloads the documents to the location `/Users/<user-id>/Library/Developer/CoreSimulator/Devices/<device-ID>/data/Containers/Data/Application/<app-id>/Documents/Samples`
6. Access ▶ [Xcode](#) ▶ [Resources](#) ▶ [SAPBI_Samples.bundle](#) ▶ . ▶
7. Right click on `SAPBI_Samples.bundle`, choose [Show Package Contents](#) and remove all the existing default files.
8. Move all the samples from the download location (specified in step # 5) to [SAPBI_Samples.bundle](#).
9. Build and run to validate the new samples in the app on simulator (command + R).

i Note

If you do not want to upgrade the existing application, you may want to uninstall the existing SAP BI app before building, and clean the application (command + shift + K).

If you do not want sample BI documents to appear on the Home screen of the application, you can disable the display of samples using the `feature.show.samples.enabled.default` property in the `DefaultSettings.plist`. See the related topic of this chapter for more information.

Related Information

[Customizing the Application Functionality \[page 33\]](#)

6 Extending the Feature Capabilities of the Application

The following extension capabilities are provided by the SAP BusinessObjects Mobile application SDK:

- Implementing new Mobile-specific charts that are available in Web Intelligence, but are not yet available in the application.
- Implementing niche Mobile-specific visualizations on the tabular data returned from Web Intelligence.

i Note

Until the current release, the new Web Intelligence charts and tables which you implement in the application are only supported in the “Page” view of the report.

See the sub-section of this chapter for information on how to perform the extension tasks above.

6.1 Implementing Extensions

Prerequisites:

1. You have downloaded the application project file `MOBSBOPIOSSDK00P_XXXXXXXXX.ZIP` from the SAP Service Marketplace (SMP link).
2. You have the appropriate Xcode version to open and compile the project.
 - Use Xcode 8.x to open and compile SAP BusinessObjects Mobile application for iOS 6.5 SDK.
 - Use Xcode 7.x to open and compile SAP BusinessObjects Mobile application for iOS 6.4.x SDK.
3. You have iOS programming skills with good knowledge of Objective C.

Implementing New Web Intelligence Chart Types

To implement new Web Intelligence charts in the application, perform the following steps:

1. Create a class of the type `UIView` (for example `SampleView`).
2. Add the created class and the dependent libraries to the `SAPBI_SDK.xcodeproj` project.
3. Open the `DefaultSettings.plist` file available at the following location in your Xcode project bundle:
▶ [Xcode](#) ▶ [Resources](#) ▶
4. Navigate to the `Extensions` section of the file as shown in the figure below:

| | | |
|-------------------|------------|---------------------|
| ▼ Extensions | Dictionary | (1 item) |
| ▼ WebIntelligence | Dictionary | (2 items) |
| ▼ Tables | Array | (1 item) |
| ▼ Item 0 | Dictionary | (2 items) |
| blockNamePrefix | String | myprefix_ |
| viewProvider | String | SampleView |
| ▼ Charts | Array | (1 item) |
| ▼ Item 0 | Dictionary | (2 items) |
| viewType | String | chart:sample:sample |
| viewProvider | String | SampleView |

- Specify a class name for `<ViewProvider>`, for example SampleView.

i Note

`<ViewProvider>` is the physical class name that you create for the view of table.

- Specify the viewType of the Web Intelligence chart.

i Note

`<viewType>` represents the type of Web Intelligence chart that you want to implement. The `<viewType>` of charts that are supported in Web Intelligence but are not yet implemented in the application are provided in the table below:

| Chart Name | viewType (to be specified in DefaultSettings.plist) |
|---------------------|---|
| Box plot | chart.box:boxplot |
| Heat map | chart.map:heatmap |
| Tree map | chart.map:treemap |
| Polar scatter chart | chart.point:polarscatter |
| Tag cloud | chart.tag cloud:tagcloud |
| Radar chart | chart.radar:radar |

- Open the class `<ViewProvider>.h` and follow the sub-steps below:
 - Import two files: `BaseVizData` and `VizViewProtocol` as shown below:

```
import "BaseVizData.h"
import "VizViewProtocol.h"
```

i Note

- "BaseVizData" has a set of methods that fetch data from the application.
- "VizViewProtocol" is a protocol with a set of required and optional methods that define how your class should behave.

- Ensure that your created class file (assume `SampleView.h`) conforms to the following syntax:

```
@interface SampleView : UIView<VizViewProtocol>
{
    id<VizViewProtocol> delegate;
}
```

```
@property (nonatomic, assign) id<VizViewProtocol> delegate;
```

3. Open `SampleView.m` (implementation file) and write the `init()` method such that it assigns the sample view to a delegate. An example is shown in the code snippet below:

```
- (id) init
{
    self = [super init];
    if(self)
    {
        self.delegate =self;
    }
    return self;
}
```

4. Implement the methods provided by the class files. A description of the class files is provided at the bottom of this chapter.

Implementing New Web Intelligence Table Types

To implement new Web Intelligence tables in the application, perform the following steps:

1. Create a class of the type `UIView` (for example `SampleView`).
2. Add the created class and the dependent libraries to the [SAPBI_SDK.xcodeproj](#) project.
3. Open the `DefaultSettings.plist` file available at the following location in your Xcode project bundle:

► [Xcode](#) ► [Resources](#) ►

4. Navigate to the `Extensions` section of the file as shown in the figure below:

| | | |
|-------------------|------------|---------------------|
| ▼ Extensions | Dictionary | (1 item) |
| ▼ WebIntelligence | Dictionary | (2 items) |
| ▼ Tables | Array | (1 item) |
| ▼ Item 0 | Dictionary | (2 items) |
| blockNamePrefix | String | myprefix_ |
| viewProvider | String | SampleView |
| ▼ Charts | Array | (1 item) |
| ▼ Item 0 | Dictionary | (2 items) |
| viewType | String | chart:sample:sample |
| viewProvider | String | SampleView |

5. Specify a class name for `<ViewProvider>`, for example `SampleView`.

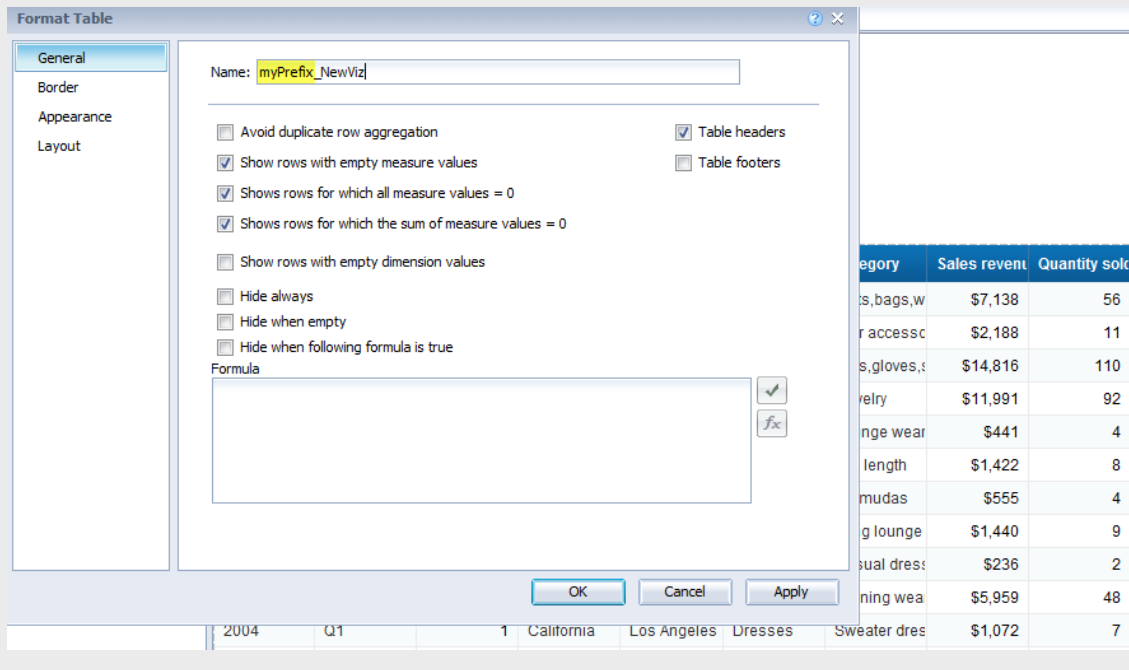
i Note

`<ViewProvider>` is the physical class name that you create for the view of table.

6. Specify a name for `<BlockNamePrefix>`.

i Note

This has to be the same as the prefix of the block name that you specify for the table-block in Web Intelligence, as shown in the figure below:



7. Open the class <ViewProvider>.h and follow the sub-steps below:
 1. Import two files: BaseVizData and VizViewProtocol as shown below:

```
import "BaseVizData.h"  
import "VizViewProtocol.h"
```

i Note

- o "BaseVizData" has a set of methods that fetch data from the application.
- o "VizViewProtocol" is a protocol with a set of required and optional methods that define how your class should behave.

2. Ensure that your created class file (assume SampleView.h) conforms to the following syntax:

```
@interface SampleView : UIView<VizViewProtocol>  
{  
    id<VizViewProtocol> delegate;  
}  
@property (nonatomic, assign) id<VizViewProtocol> delegate;
```

3. Open SampleView.m (implementation file) and write the init() method such that it assigns the sample view to a delegate. An example is shown in the code snippet below:

```
- (id) init  
{  
    self = [super init];  
    if(self)  
    {  
        self.delegate =self;  
    }  
    return self;  
}
```

```
}
```

4. Implement the methods provided by the class files. A description of the class files is provided below.

Description of the BaseVizData Class

This class gives you the information below:

- Overview information of the report part
- Access to the Database (To retrieve complete information of the report part).

The tables below provide a description of the data types and methods provided by the class:

| Data Type (with Syntax) | Description |
|--|---|
| NSDictionary *qualificationInfo; | This specifies if the table column is a measure or dimension. |
| NSString *reportName; | This returns the report part name. |
| NSMutableDictionary *reportPartBoundInformation; | This returns the reportPartBoundInformation which includes: <ul style="list-style-type: none">• reportBlockName• reportTableName• reportViewType• reportBlockid• reportSectionKey |

i Note
The <reportSectionKey> defines if the report part has sections or not.

| Method | Description |
|---|---|
| (sqlite3 *)getDatabase; | This returns the database instance. |
| (NSString *)getReportPartID:(NSString*)reportPartName; | This gives the reportpartid based on the reportPartName obtained from reportpartboundinformation. |
| (void)getAxisViewDetails:(NSInteger)reportId blockId:(NSInteger)blkId axisAttributes:(NSMutableArray *)axisAttrs; | This gives the axis view details by filling axisAttrs array. |
| (void)getReportBlockProperties:(NSInteger)reportId blockId:(NSInteger)blockId properties:(NSMutableDictionary *)properties; | This gives the report part block properties by filling array properties. |

Description of Methods in the VizViewProtocol Class

| Method | Description |
|---|--|
| (id)createViewWithData:(BaseVizData *)data; | This method is used to create a View and return the view back with the delegate of that class. When you implement this method in the .m file, it returns the data from the application. You use this data persistently to create your views. |
| (UIView *)getCustomView; | This method fetches the original view with the help of delegate. |
| return self; | This method returns your created view to the class. |
| (void)reLayoutSubviews; | This method is used to set frames (for revised coordinates) of the report parts when the orientation of the device is changed (portrait to landscape or viceversa). |
| (void)setFrameForView:(CGRect)frame; | This method sets the frame for your view. |
| (void)gotoZoomViewforNewCharts; | This method zooms the charts. |

i Note
 This method is currently not supported in the application.

7 Embedding SAP BI Analytics

You can now embed analytic views (Web Intelligence reports and Dashboards) supported by the SAP BI application inside your own custom mobile applications. The embedded BI information is managed by the SAP BusinessObjects BI platform. This feature enhances the context and usability of business data.

For example, your enterprise may have a leave approval application that managers can access on their iOS devices. Before approving leaves, a manager may want to look at the leave history of the employee before making a decision. For the employee leave history, you can embed a Web Intelligence report or a Dashboard (supported by the SAP BusinessObjects Mobile application) as part of your mobile application.

Similarly, you may have an expense approval application, where managers want to look at the history of expenses before approving a new one. You can embed analytical reports or report parts from the SAP BI application to enrich your expense approval application with insights.

7.1 Modifying Your Project for Use with SAP BI Libraries

Prerequisites:

1. You have downloaded the application project file `MOBSBOPIOSSDK00P_XXXXXXXXX.ZIP` from the SAP Service Marketplace (SMP link).
2. You have the appropriate Xcode version to open and compile the project.
 - Use Xcode 8.x to open and compile SAP BusinessObjects Mobile application for iOS 6.5 SDK
 - Use Xcode 7.x to open and compile SAP BusinessObjects Mobile application for iOS 6.4.x SDK.
3. You have iOS programming skills with good knowledge of Objective C.

Modifying Your Application

To prepare your application for use with SAP BI libraries, modify it by following the procedure below:

1. Unzip the application project file to open the SAP BusinessObjects Mobile project.
A folder named `SAPBusinessObjectsMobile-<app_version_number>-xcodeproj-with-deps` appears.
2. Rename the folder `SAPBusinessObjectsMobile-<app_version_number>-xcodeproj-with-deps` to a shorter name and ensure the name does not contain spaces or special characters.

i Note

In this example, the folder above is renamed to `MOBISDK`.

3. Launch Xcode, select your project, and perform the following steps:

1. Access **Build Phases** > **Link Binary With Libraries** and include the following frameworks as a library in your project.
 - For running on a device:
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/Logger
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/MobiCharts
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/MOBILumiraOffline
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/SAPBusinessObjectsLimo
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/SAPBusinessObjectsMobileSDK
 - For running on a simulator:
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/Logger
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/MobiCharts
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/MOBILumiraOffline
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/SAPBusinessObjectsLimo
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/SAPBusinessObjectsMobileSDK

2. Access **Build Settings** > **Framework Search Paths** and include the following folders as a framework search path in your project:
 - For running on a device:
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/Logger
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/MobiCharts
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/MOBILumiraOffline
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/SAPBusinessObjectsLimo
 - MOBISDK/target/xcode-deps/frameworks/Release-iphones/com.sap.mobile/SAPBusinessObjectsMobileSDK
 - For running on a simulator:
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/Logger
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/MobiCharts
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/MOBILumiraOffline
 - MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/SAPBusinessObjectsLimo

- MOBISDK/target/xcode-deps/frameworks/Release-iphonesimulator/com.sap.mobile/SAPBusinessObjectsMobileSDK
3. Use [Add files...](#) to add the following resources to your project:
 - MOBISDK/src/xcode/Resources/DefaultSettings.plist
 - MOBISDK/src/xcode/Resources/Bundles/SAPBI_Icons.bundle
 - MOBISDK/src/xcode/Resources/Localizable
 - MOBISDK/src/xcode/Resources/Bundles/ColorTheme.bundle
 - MOBISDK/src/xcode/Classes/Connections.plist
 - MOBISDK/src/xcode/Resources/DataBase/MobiDatabase.sqlite
 - MOBISDK/src/xcode/Classes/SingleClient/Common/Artifact/Artifacts.plist
 4. Access [Build Settings](#) [Other Linker Flags](#) and add `-ObjC` as a linker flag in your project.
 5. Access [Build Phases](#) [Link Binary With Libraries](#) and include the following frameworks in your project:
 - Accelerate.framework
 - AddressBook.framework
 - AddressBookUI.framework
 - AVFoundation.framework
 - CoreFoundation.framework
 - CoreGraphics.framework
 - CoreLocation.framework
 - CoreMedia.framework
 - Foundation.framework
 - ImageIO.framework
 - MapKit.framework
 - MessageUI.framework
 - MobileCoreServices.framework
 - SystemConfiguration.framework
 - UIKit.framework
 6. Access [Build Phases](#) [Link Binary With Libraries](#) and include the following frameworks in your project:
 - libstdc++.6.0.9.dylib
 - libxml2.dylib
 - libz.1.2.5.dylib
4. Modify your source files as explained in the related topic provided at the bottom.

Related Information

[Using SAP BI Libraries to Perform Tasks \[page 63\]](#)

7.2 Using SAP BI Libraries to Perform Tasks

Prerequisites:

1. You have downloaded the application project file `MOBSBOPIOSSDK00P_XXXXXXXX.ZIP` from the SAP Service Marketplace (SMP link).
2. You have the appropriate Xcode version to open and compile the project.
 - Use Xcode 8.x to open and compile SAP BusinessObjects Mobile application for iOS 6.5 SDK.
 - Use Xcode 7.x to open and compile SAP BusinessObjects Mobile application for iOS 6.4.x SDK.
3. You have iOS programming skills with good knowledge of Objective C.

Using SAP BI Libraries For Your Tasks

To authenticate to the SAP BusinessObjects BI platform and to embed a Web Intelligence/Dashboards view in your application, you need the following header files:

- `SAPBusinessObjectsMobileSDK/SAPBIAsyncOperation.h`
- `SAPBusinessObjectsMobileSDK/SAPBIDocument.h`
- `SAPBusinessObjectsMobileSDK/SAPBIOperationDelegate.h`

These header files are located in the `SAPBusinessObjectsMobileSDK` framework. The following table describes the key use of each of these three classes:

| Class (Header File) | Description |
|---------------------------------------|---|
| <code>SAPBIAsyncOperation.h</code> | This class exposes methods such as <code>login()</code> , <code>logout()</code> and <code>getDocumentView()</code> . You use these methods to authenticate and obtain the view for a specific Web Intelligence report or a Dashboard. |
| <code>SAPBIDocument.h</code> | This class contains the property to access the <code>UIViewController</code> for the document that you intend to view. You can either use the <code>UIViewController</code> directly, or get a <code>UIView</code> to embed within your application's <code>UIViewController</code> . |
| <code>SAPBIOperationDelegate.h</code> | Contains the methods that are called on completion of methods defined in the <code>SAPBIAsyncOperation</code> class. You must implement this delegate so that you are notified of successful or failed authentication. Similarly, the method <code>getDocumentSucceeded</code> is called with an instance of the <code>SAPBIDocument</code> class in case of a successful return for <code>getDocumentView()</code> . |

i Note

You can find more details for the methods of these classes in the respective header files.

7.3 Integrating with Third Party Security Solutions

Although SAP BusinessObjects Mobile is a secure solution, you may have an organizational need to integrate with a third party mobile security solution.

For mobile security solutions that take the approach of wrapping mobile apps, use the technique described in chapter 2 of this guide ("Building the Application with Your Certificate") to generate an unsigned SAP BusinessObjects Mobile IPA file. The IPA file can then be fed into the security wrapping tool by following the third party vendor's instructions. For mobile security solutions that require further integration, contact your SAP account representative for additional information.

Related Information

[Building the Application with your Certificate \[page 7\]](#)

7.3.1 Wrapping the Application with Citrix Xenmobile (iOS)

Mobile application management allows you to securely manage and deliver mobile applications to users. Citrix Xenmobile provides a cloud based platform to wrap applications. Policies specific to organizations can be enforced and the applications can then be distributed to enterprise users securely through the CitrixWorx application.

Prerequisites

- SAP BusinessObjects Mobile iOS SDK (from SAP Service Market Place)
- Citrix Xenmobile Platform for wrapping and applying policies
- MDX Toolkit (from Citrix)
- Latest version of xCode installed on your Mac
- Provisioning profile and Production certificate (If you are deploying the application enterprise-wide, you require production standards as per Apple). For internal POC, you can continue to use a Developer Certificate and Provisioning Profile.

Production Certificate:

https://developer.apple.com/library/content/documentation/IDEs/Conceptual/AppDistributionGuide/MaintainingCertificates/MaintainingCertificates.html#//apple_ref/doc/uid/TP40012582-CH31-SW1

Provisioning Profile:

https://developer.apple.com/library/content/documentation/IDEs/Conceptual/AppDistributionGuide/MaintainingProfiles/MaintainingProfiles.html#//apple_ref/doc/uid/TP40012582-CH30-SW1

Configuring the Application

This process can be broadly classified into the following steps:

1. Creating the unsigned IPA out of SAP BusinessObjects Mobile SDK.
2. Creating the MDX format file using the MDX Toolkit provided by Citrix
3. Uploading the MDX format file to Xenmobile Platform, configuring the parameters for SAP BusinessObjects Mobile as required by your organization, and generating the executable application.
4. Distribution of the wrapped application.

Creating the unsigned IPA out of SAP BusinessObjects Mobile SDK

1. Unzip the downloaded SDK project
2. Double-click the *SAPBI_SDK.xcodeproj* project file to open the SDK project in Xcode IDE.

i Note

You may perform customizations at this stage based on your needs.

3. SAP BusinessObjects Mobile SDK project comes with a default Bundle ID for the application, which you need to change based on your organization's namespace.
 1. Select *SAPBI_SDK* project file.
 2. Under *Target*, choose *SAP BusinessObjects Mobile*.
 3. Choose the *General* tab and modify the Bundle identifier.
4. In *SAPBusinessObjectsMobile-Info.plist* of the unzipped SDK project folder, change the Bundle ID in the *Bundle Identifier* field.
5. Once you customize the application as per your requirement, do a clean and then build in the IDE. Check if the application is running successfully on iOS simulator.
6. Generate the IPA from xCode as mentioned in the procedure here: <https://developer.apple.com/library/tvos/documentation/IDEs/Conceptual/AppDistributionGuide/TestingYouriOSApp/TestingYouriOSApp.html>

Creating the MDX format file through the MDX Toolkit

Download MDX Toolkit from Citrix website. MDX toolkit is available for Mac OS X. You can refer to the following link for MDX installation detail: <http://docs.citrix.com/en-us/mdx-toolkit/10/xmob-mdx-kit-install.html>. To install MDX toolkit in Mac, simply double-click the downloaded DMZ and Install.

To generate the MDX file from IPA:

1. Open the *MDX Toolkit*. In the welcome screen, read the instructions and choose *Next*.
2. In the screen that appears, enter the IPA you have generated, and choose *Next*.
3. Fill in all the fields based on your enterprise needs and choose *Next*.
4. Choose the relevant provisioning profile and the iOS certificate.
5. Check if the Bundle ID defined in your SDK project and App ID are correct for the provisioning profile and choose *Yes/No*. Then, choose *Create*.
6. The IPA file is now generated and the confirmation window appears. Choose *Show in Finder* to see the generated MDX file.

Uploading the file to XenMobile Platform and Configuring the Required Policies

1. Login to the Citrix Xenmobile administration console with your credentials.

2. Choose the *Configure* tab, and then the *Apps* tab.
3. Choose *Add* to add the newly created MDX file.
4. Choose the MDX file which has been generated from MDX toolkit.
5. In the left panel, uncheck *Android* and *Windows*.
6. Enter the required details, select the *enterprise app category* which has been provided by your administrator, and choose *Next*.
7. Choose the MDX file and upload.
8. On the Settings screen that appears, configure the settings as mentioned below:
Below are the mandatory changes to be made:
 - Under *Encryption settings*, Select *Off* for *Enable Encryption*.
 - Under App Interaction:
 1. In the *Inbound document exchange (Open In)* field, select *Unrestricted* from the dropdown.
 2. In the *App URL schemes*, Add `;sapbi:` to the existing string
 3. Under *App Network Access*, In the *Network access* field, choose *Unrestricted or Secure Tunnel*.

The changes below are optional:

Under *App Restrictions*:

- Set *Block mic* record to *off* .

i Note

The application uses Mic for annotation.

- Camera
9. On the *Approvals* page, set *Workflow to Use* as *none* and choose *Next*.
 10. On the *Delivery Group Assignments* page, select the *All Users* option.
The policies mentioned above are specific to SAP BusinessObjects Mobile. You can apply more settings based on the security standards of your enterprise.
 11. After making the required changes, choose *Save and Finish*.

Distribution of the wrapped application

To access the configured Citrix wrapped application, download the Citrix Worx home application to your iOS device from Apple app store and login to the application with your citrix credentials. If you have been given access for downloading your customized SAP BusinessObjects Mobile application, you can download the same to your mobile from this Citrix Home application.

If there are any changes in policies, the application will be redeployed as per Citrix standards by Citrix MDM itself.

7.3.2 Blackberry Dynamics SDK

7.3.2.1 Prerequisites for Integrating the Application with Blackberry Dynamics SDK

When Blackberry Dynamics SDK is integrated with SAP BusinessObjects Mobile SDK, it securely manages data and devices such as secure email, calendar information, contact details, browser, access, tasks management, and document data.

System Requirements

- Mac with OS X El Capitan 10.12.3 or later.
- Xcode Version 8.3 or above from the App Store.

i Note

To install:

1. Press the Command (⌘) and Space button for spotlight search.
2. Type App Store and press the return key to open the App Store.
3. Search for *Xcode* and install.

- Blackberry Dynamics SDK

i Note

To install:

1. Go to <https://community.good.com/community/gdn> and log in. If you have not registered, then register to log in.
2. Go to [► Resources ► Blackberry Dynamics ► Downloads ►](#).
3. Under *SDK for iOS*, download *Blackberry Dynamics SDK v3.1.0.405*.
4. After the download is complete, unzip *BlackBerry_Dynamics_SDK_for_iOS_v3.1.0.405.tar*. The uncompressed .tar file gives *BlackBerry_Dynamics_SDK_for_iOS_v3.1.0.405.pkg*
5. Open the .pkg file and complete the installation process.

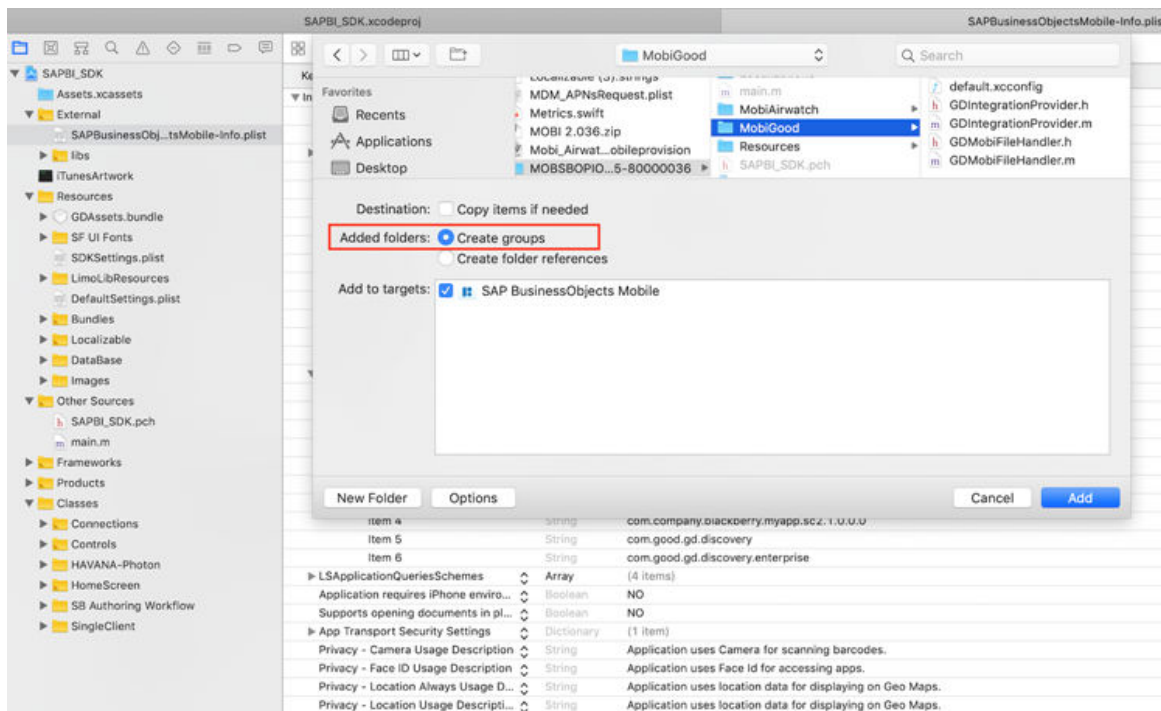
i Note

- SAP BusinessObjects Mobile SDK for Blackberry integration will work only on your device and not on simulator.
- The e-mail functionality is not supported.
- You need to create a new app ID in the format - `com.mycompany.app` with all the letters in lower case.

- SAP BusinessObjects Mobile SDK, which you can download from the Service Market Place.

7.3.2.2 Integrating the Application with the BlackBerry Dynamics SDK

1. Unzip the downloaded *SAP BusinessObjects Mobile SDK* zip.
2. Open the *SAPBI_SDK.xcodeproj* file with Xcode.
3. Right click *SAPBI_SDK* and choose *Add files to "SAPBI_SDK"*.
4. Add the *MobiGood* folder to the project as shown in the screenshot.

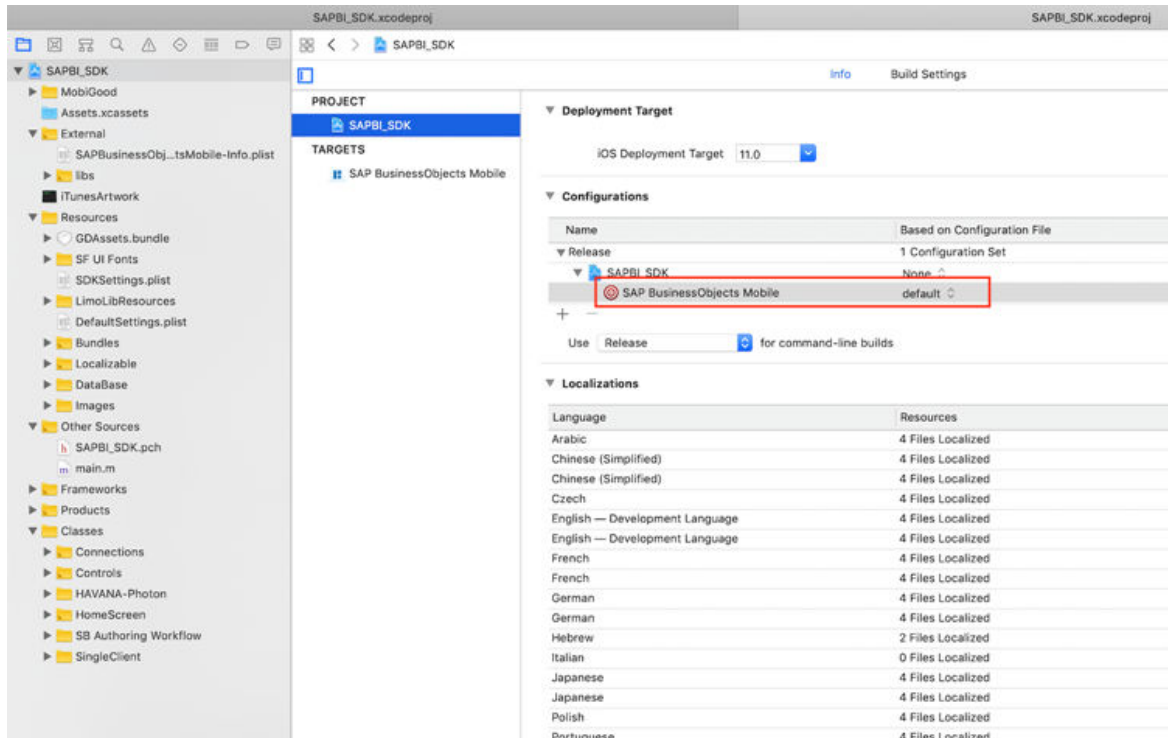


5. In the left pane, right click *Resources* and choose *Add files to "SAPBI_SDK"*.
6. Press the Command+Shift+G keys to show *Go to the folder*.
7. Enter the path: `/Users/{system name}/Library/Application Support/BlackBerry/Good.platform/iOS/Frameworks/GD.framework/Versions/A/Resources` and choose *Go*.

Note

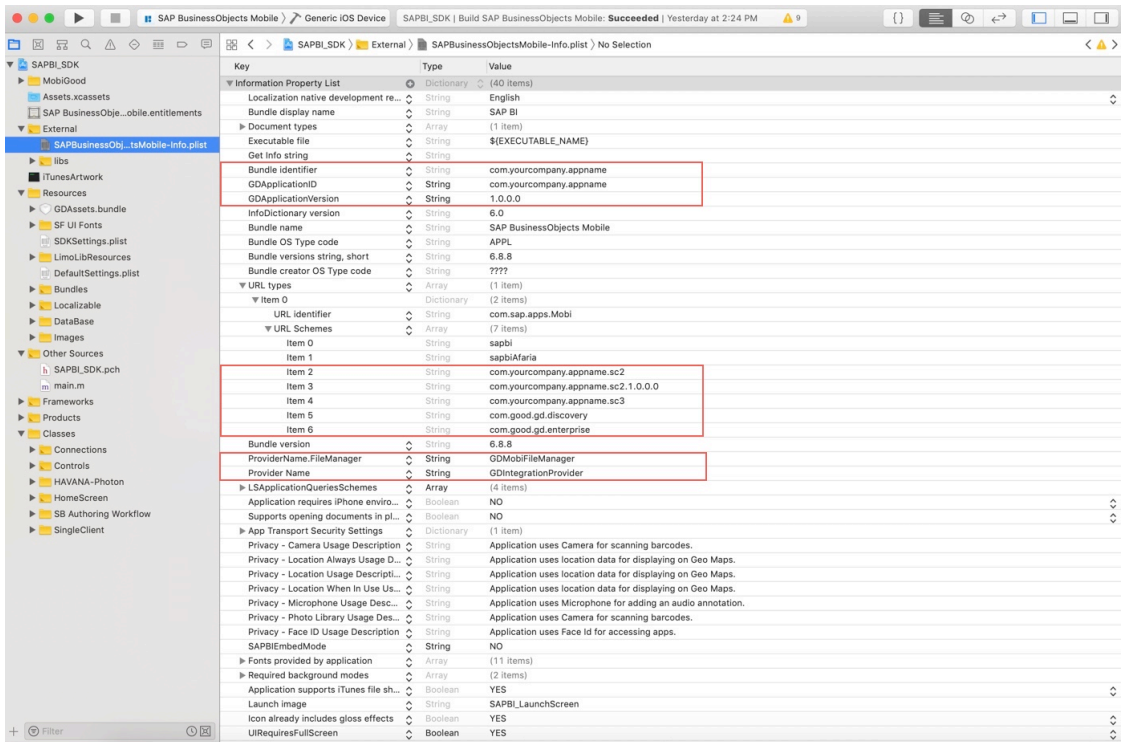
Replace {system name} with your system/home name.

8. Add *GDAAssets.bundle* under *Resources*.
9. Navigate to **SAPBI_SDK** > *SAP BusinessObjects Mobile* > *Build Phase* and add the following GD dependency frameworks:
 - CoreTelephony.framework
 - CoreData.framework
 - GD.framework
10. Go to **SAPBI_SDK Project** > *SAP BusinessObjects Mobile* > *Info*. Under *Release*, set *SAP BusinessObjects Mobile* to default.



11. Go to *SAPBusinessObjectsMobile-info.plist* and make the following changes:

- Modify the *Bundle identifier* to `<com.yourcompany.appname>`.
- Modify the *Provider Name* to `GDIntegrationProvider`.
- Modify the *ProviderName.FileManager* to `GDMobiFileManager`
- Add the key *GDApplicationID* with the value `<com.yourcompany.appname>`
- Add the key *GDApplicationVersion* with the value `1.0.0.0`.
- Navigate to **URL Types** > **Item 0** > **URL Schemes** and add the following:
 - `<com.yourcompany.appname.sc2>`
 - `<com.yourcompany.appname.sc2.1.0.0.0>`
 - `<com.yourcompany.appname.sc3>`
 - `com.good.gd.discovery`
 - `com.good.gd.discovery.enterprise`



i Note

- Dashboard content is not supported for Blackberry integration.
- To exclude Dashboard, go to **Classes > SingleClient > Common > Artifact > Artifacts.plist**
- Open **Item 4**, for *unsupportedPlatforms* and add `iPhone; iPad`.

- Go to the project **SAPBI_SDK > Targets > Capabilities > Keychain Sharing > Switch On**.
- Connect the device and choose the **Run** button on Xcode to install the application on your device.
- In the blackberry console create GD App ID under **Manage AppAdd App** button.
- From **Public App Store, Custom, Web, GD Entitlement ID and Version only**, select the **GD Entitlement ID and Version only** option.
- To generate access keys, navigate to **Users and Groups > User > Edit User > ACCESS KEY**.
- Enter your **email address** and **PIN**. Then, the authentication application asks for the Blackberry's application password. The SAP BusinessObjects Mobile application appears on your device.

i Note

You cannot open a Web Intelligence document in the SAP BI Mobile application wrapped with Blackberry

Dynamics SDK when the **Legacy Mode** of connection (**Settings > Additional Settings > Connection Settings > Legacy Mode**) is disabled.

7.3.3 Airwatch SDK

7.3.3.1 Airwatch Application Wrapping

To wrap the application using Airwatch ensure that your device is enrolled with Airwatch.

1. Download the **SDK for iOS**, which is the client library.
2. Download **SAP BI Mobile SDK.zip**. Unzip and navigate to `src > xcode`.
3. Right-click `SAPBI SDK.xcodeproj > Open With > xcode 8.0 or above`.
4. In the left pane, right-click `SAPBI_SDK > Add files to > SAPBI_SDK`.
5. In the window that appears, choose *MobiAirwatch* that contains:
 - o AWIntegrationProvider.h
 - o AWIntegrationProvider.mand choose *Add*.
To Disable the SAP BusinessObjects Mobile Application Password, return `No` in the **shouldEnableApplicationPassword** method.
6. To add the frameworks, select *SAPBI_SDK*
7. Choose the *Build Phase* tab.
8. In the *Link Binary with Libraries* drop down list, add the following frameworks:
 - o AssetsLibrary.framework
 - o AudioToolbox.framework
 - o CoreData.framework
 - o CoreMotion.framework
 - o CoreTelephony.framework
 - o CoreText.framework
 - o Libc++.tbd
 - o Libstdc++.6.0.9.tbd
 - o Libz.tbd
 - o MediaPlayer.framework
 - o SafariServices.framework
9. Right-click *SAPBI_SDK*.
10. Add **AWKit.bundle** and **SDKLocalization.bundle**
11. To receive a call back from the AirWatch agent, your application should expose a custom scheme in the info.plist. To do so,
 1. Navigate to Supporting Files in Xcode.
 2. Select *SAP BusinessObjectsMobile-Info.plist*
 3. Add the parameters indicated in the screenshot below.

| | | | |
|--------------------------------|-------|------------|---------------------------|
| ▼ URL types | ▲ | Array | (1 item) |
| ▼ Item 0 | | Dictionary | (2 items) |
| URL identifier | ▲ | String | com.sap.apps.Mobi |
| ▼ URL Schemes | ▲ | Array | (3 items) |
| Item 0 | | String | sapbi |
| Item 1 | | String | sapbiAfaria |
| Item 2 | | String | sapbiAirWatch |
| ▼ LSAApplicationQueriesSchemes | ▲ + - | Array | ⌵ (5 items) |
| Item 0 | | String | afaria |
| Item 1 | | String | afaria-getdata |
| Item 2 | | String | afaria-enterprise-getdata |
| Item 3 | | String | afaria-appstore-getdata |
| Item 4 | | String | airwatch |

12. Change *Provider Name* to AWIntegrationProvider.
13. Navigate to your Xcode build settings and set *Enable Bitcode* to No.
14. Generate the IPA file and upload it to the Airwatch Admin Console.
15. Log on to the Airwatch console, choose ► *Apps & Books* ► *List ViewInternal* ► *Add Application* ►.
16. Select the IPA file and click on *Choose*.
17. In the window that appears, choose *Save*. Then choose *Continue*, then *Save and Assign*.
18. Choose *Add Assignment* .
19. Select the *Smart Group* that you would like to provide access to.

i Note

In the *Advanced* tab, under *Application Configuration*, you can push additional connections via Airwatch.

20. Choose *Save and Publish*.
21. Choose ► *Application* ► *Devices* ► and select the assigned device.
22. Choose *Install*.

7.3.3.2 Consuming Client Certificates via Airwatch console

7.3.3.2.1 Configuring the Application Profile for Certificate Push

To configure the application profile for certificate push, follow the procedure below:

1. On the Airwatch console, in the left pane, choose ► *All Apps & Books Settings* ► *Settings and Policies* ► *Profiles* ►
2. Choose *Add Profile*.
3. Select *Configuration Type* as *Application Profile*.
4. In *Add Profile*, select *iOS Apple*.
5. In the *General* tab, enter the *Name*.
6. In the *Credentials* tab, set the *Credentials Source* field to *Upload*.
7. Upload the client certificate.
8. Enter the password for the uploaded certificate and choose *Save*.
9. Choose *Save Again*.

10. Close the newly added profile.
11. Navigate to ► [Apps and Books](#) ► [Applications](#) ⌵, and select the uploaded application.
12. Choose [Edit](#).
13. In the [More](#) tab, set the [SDK Profile](#) as [iOS Default Settings @ Global](#) and set the [Application Profile](#) as the one you created earlier.
14. Choose [Save and Assign](#).
15. Select the [Required Assignment](#), then choose [Save and Publish](#).
16. Choose [Publish](#). You can see the message "Saved Successfully".
17. In the left pane, choose [Applications](#).
18. In the [Devices](#) tab, select the assigned/enrolled devices and choose [Install](#).
19. Install and launch the application on your device, and enter the Airwatch password.

i Note

In case the client certificate expires, follow the same procedure as mentioned above. However, you need to delete the existing SAP BusinessObjects Mobile application on your device and re-install.

7.3.3.2 Setting up Airwatch for Integrated Authentication

To set up Airwatch for Integrated Authentication, you need to execute the following procedure:

1. Go to the Airwatch Console and navigate to ► [Settings](#) ► [Apps](#) ► [Settings and Policies](#) ► [Security Policies](#) ⌵
2. Select [Enabled](#) as the [Integrated Authentication](#).
3. Select [User Certificate](#).
4. In the [Allowed Sites](#) field, type "*".
5. Go to ► [Accounts](#) ► [List View](#) ⌵ and edit the enrollment user.
6. Choose the [Advanced](#) tab. Under [Certificates](#), enable [Use S/MIME](#) and upload the [certificate.p12](#) with a password and choose [Save](#).
7. In the [SDK](#) tab, choose the [SDK Profile](#) and [Application Profile](#). Choose [Save & Assign](#) to assign the devices.
8. In the [Update Assignment](#) tab, choose [Add Assignment](#) and choose [Save and Publish](#).
9. In the left panel, choose ► [APPS AND BOOKS](#) ► [Details View](#) ⌵.
10. In the [Devices](#) tab, select the required device and choose [Install](#).

7.3.3.3 Airwatch Application Configuration

The SAP BusinessObjects Mobile application for iOS supports Airwatch version 5.9.1 and above.

To configure the SAP BusinessObjects Mobile Application through Airwatch console, follow the below procedure:

1. Log on to the Airwatch console (<https://apidev.awmdm.com>).
2. Choose [Apps & Books](#), then choose [List View](#).
3. Select the [Public](#) tab.

4. Choose [Add Application](#).
5. In the [Add Application](#) dialog that appears, choose the following:
 - Platform: Apple iOS
 - Source: Search App Store
 - Name: SAP BusinessObjects Mobile and choose [Next](#)
6. In the window that appears, choose SAP BusinessObjects Mobile. You will be directed to the [Add Application](#) screen with multiple tabs.
7. Fill the empty fields of [Info](#) tab appropriate to your company.
8. In the [Assignment](#) tab, fill the [Assigned Groups](#) field with Smart Groups that can access the application.
9. In the [Deployment](#) tab, choose the [Push Mode](#) to push the application to users.
10. Enable [Send Application Configuration](#) and enter the Configuration Key and Configuration Value as shown below.

| Configuration Key | Value Type | Configuration Value |
|--|------------|--|
| <input type="text" value="conn.list.1"/> | String | sapbi://addconnection/?cms=f Insert Lookup Value |
| <input type="text" value="conn.list.2"/> | String | sapbi://addconnection/?cms=ε Insert Lookup Value |

The format of Configuration Value for Enterprise connection is : `sapbi://addconnection?cms=<CMS>&server_url=<Server URL>&authType=secEnterprise&ConnectionType=BOEConnection&username=<Username>&connection_name=<Connection name>`. For other connection types, refer the section "Creating an SAP BI URL To Add a Corporate Connection" in the *Mobile BI Report Designer's Guide*.

11. Choose the [Terms of Use](#) tab and make appropriate selections.
12. Choose [Save & Publish](#)
13. In the [View Device Assignment](#) window, choose [Publish](#).
14. Select against the name of the device enrolled with AirWatch and select [Manage devices](#) from the drop-down list.
15. In the [Manage Devices](#) window, choose [Install](#).

When you launch the application on your device, you find the connections defined in the Airwatch console as predefined connections.

7.3.4 MobileIron AppConnect SDK

7.3.4.1 Wrapping the application with MobileIron AppConnect

To wrap the application with MobileIron AppConnect, follow the procedure below:

1. Download SAP Business Objects Mobile SDK and run the project.
2. Create the IPA file of the project.
3. Download the AppConnect wrapper from MobileIron.
4. Open the application after installation.
5. In the [Introduction](#) window, choose [Continue](#).

6. Drag the generated IPA onto the App IPA File window, and choose *Continue*.
7. In the Signing Identity window, enable the *Sign with signing identity* option and select the correct signing identity from the drop-down list. Choose *Continue*.
8. Choose *Wrap* to generate the wrapped application.
9. Upload the wrapped application to MobileIron.
The wrapped application is now ready for distribution.

7.3.4.2 MobileIron Application Configuration

To configure the SAP BusinessObjects Mobile Application through the MobileIron console, follow the below procedure:

1. Log into your MobileIron account.
2. Select the *Apps* tab.
3. Choose *+Add*.
4. Choose *App Store* from the drop-down list to the left of the search bar.
5. In the search bar, enter SAP Business Objects Mobile.
6. Select the correct version of the application from the list, and choose *Next*. Details of the application is listed.
7. Choose *Next*.
8. Choose the *Custom* option.
9. Choose the *Users* and the *User Groups* to which you want to distribute the application.
10. Choose *Next*.
11. To add a new configuration, choose the *+* option beside *iOS Managed App Configuration*.
12. Enter a *Name* for the configuration.
13. Under *iOS Managed App Settings*, add a new key-value pair as shown in the following image.

iOS Managed App Settings

| Key | Value |
|-----------------------|--------------------------------------|
| conn.list.1 | sapbi://addconnection?cms=localhost& |
| + Add | |

The format of Configuration Value for Enterprise connection is : `sapbi://addconnection?cms=<CMS>&server_url=<Server URL>&authType=secEnterprise&ConnectionType=BOEConnection&username=<Username>&connection_name=<Connection name>`. For other connection types, refer the section "Creating an SAP BI URL To Add a Corporate Connection" in the *Mobile BI Report Designer's Guide*.

14. Choose *Next*. The new configuration is now added.
15. Choose *Done* to finish the application configuration.
The *Apps* tab now displays the configured applications.

Install the application on your device from the *Mobile Iron catalogue app*.

7.3.5 Microsoft Intune Application Configuration SDK

7.3.5.1 Adding Groups Into Microsoft Intune

In order to make the SAP Business Objects Mobile Application available to a particular set of users or devices enrolled with Microsoft Intune, you must create a group.

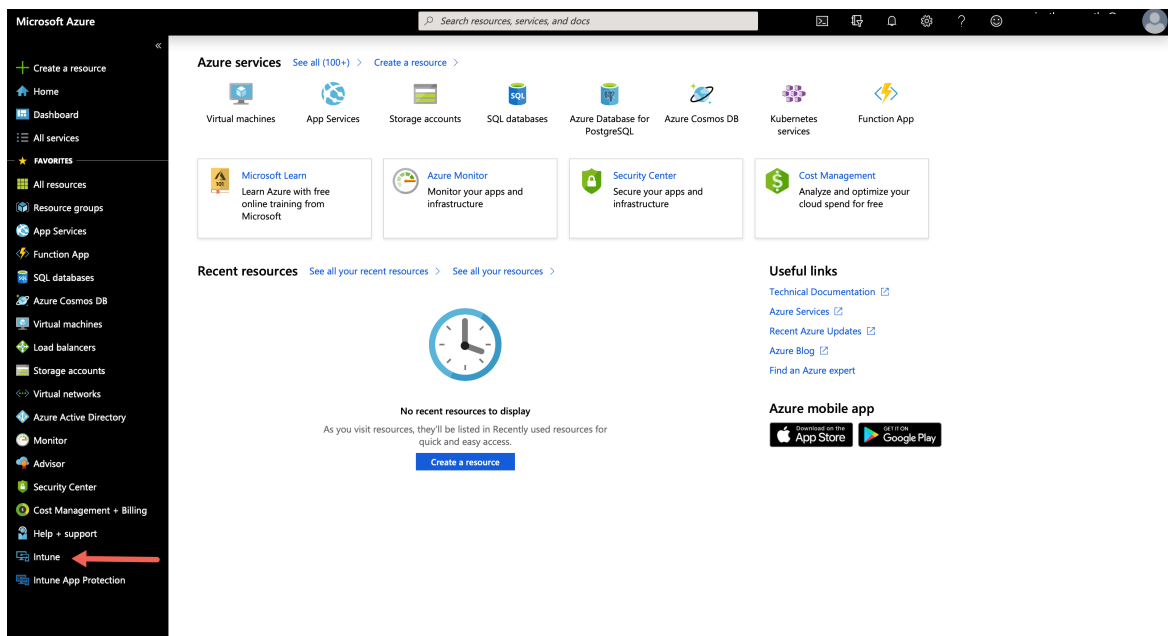
1. Log in to Microsoft Intune using your login credentials.

Server <https://portal.azure.com/>

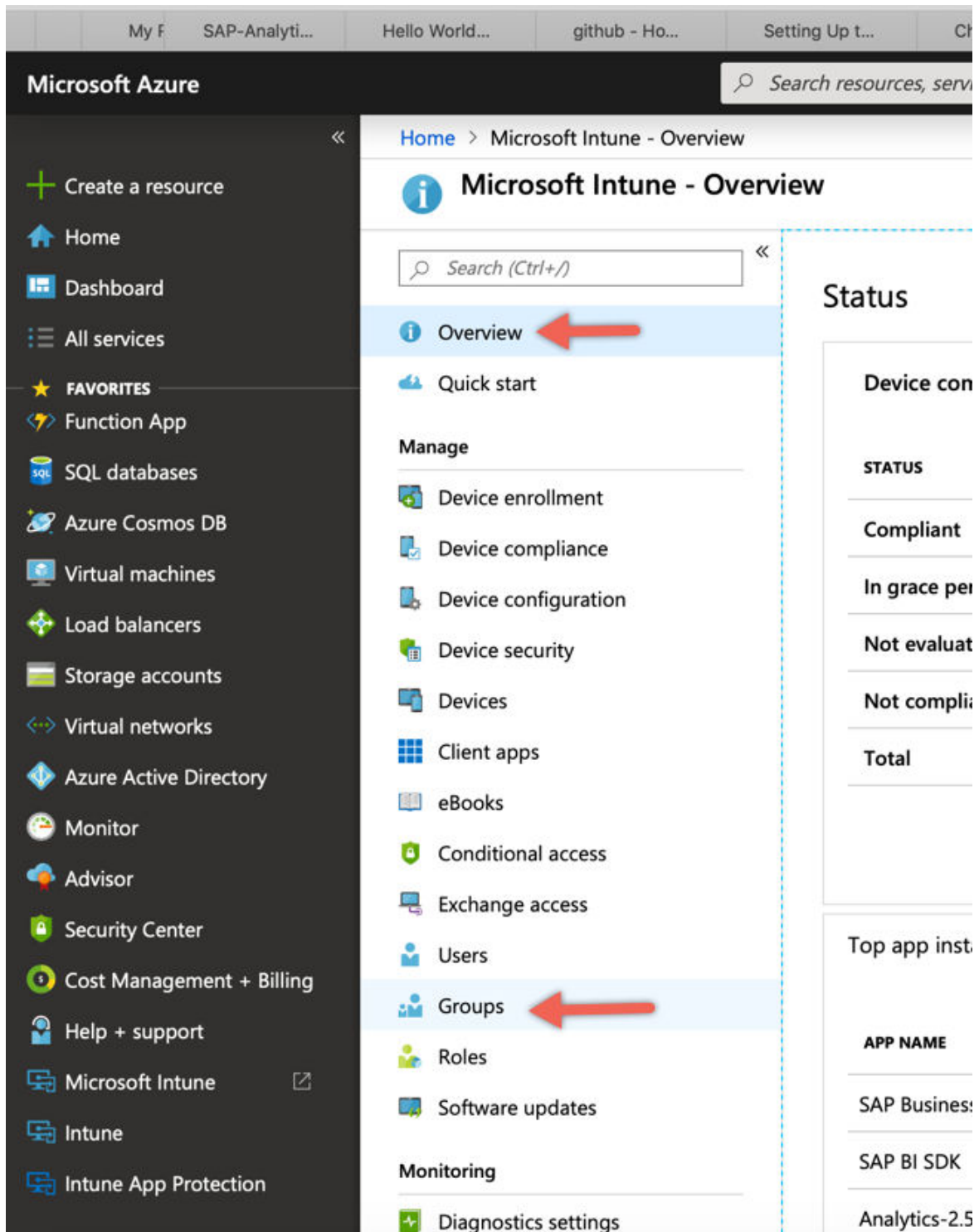
Username: <yourusername>

Password: <yourpassword>

2. Click *Intune* in the left-hand panel of the Microsoft Azure portal.



3. Click **Overview** > **Groups**.



4. Click *All groups*, and then *New group*.

Microsoft Azure

Search resources, services, and docs

Home > Microsoft Intune > Groups - All groups

Groups - All groups
SAP Labs India - Azure Active Directory

+ New group Delete Refresh

Search groups

| NAME | OBJEC |
|---------------|-------|
| SM SAC Mobile | a26e |

Settings

- General
- Expiration
- Naming policy

Activity

- Access reviews
- Audit logs

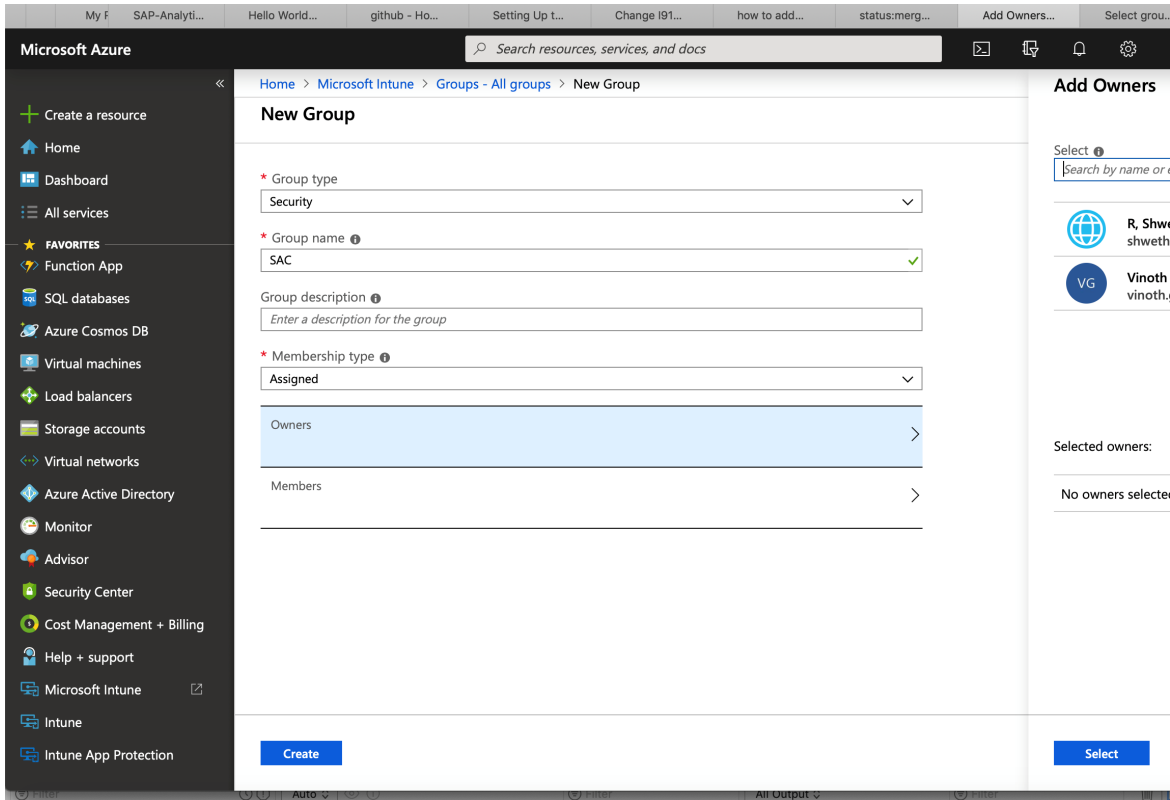
Troubleshooting + Support

- Troubleshoot
- New support request

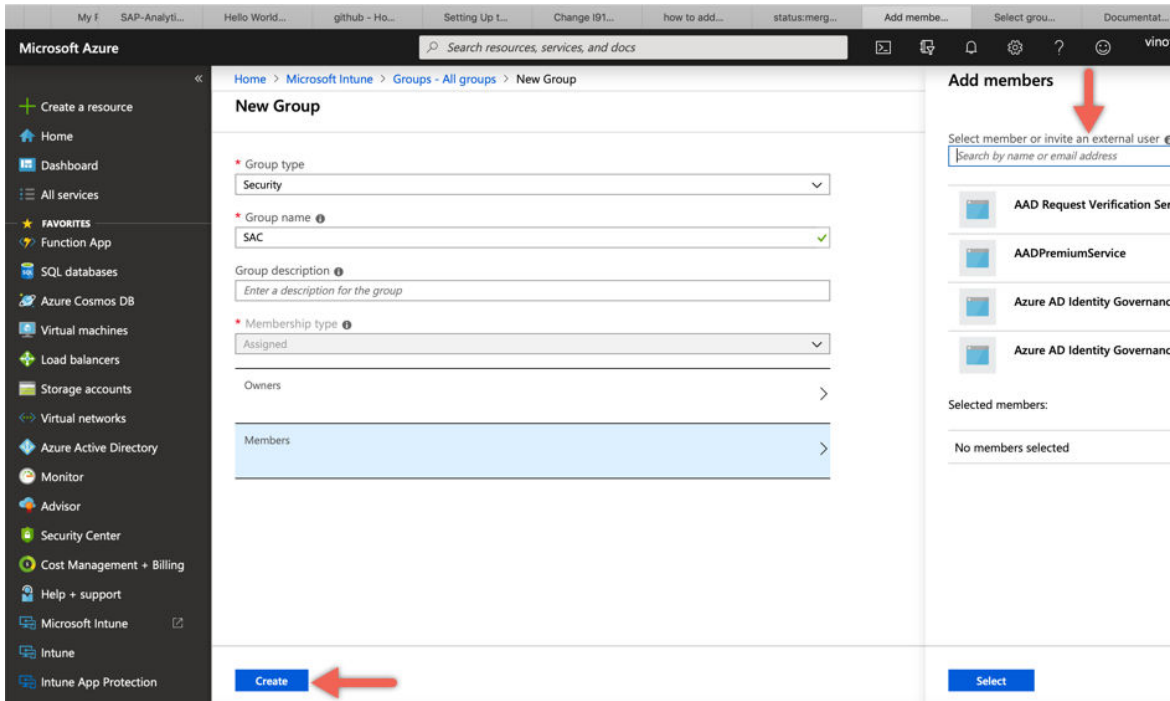
Navigation pane:

- Create a resource
- Home
- Dashboard
- All services
- FAVORITES
- Function App
- SQL databases
- Azure Cosmos DB
- Virtual machines
- Load balancers
- Storage accounts
- Virtual networks
- Azure Active Directory
- Monitor
- Advisor
- Security Center
- Cost Management + Billing
- Help + support
- Microsoft Intune
- Intune
- Intune App Protection

5. Enter the name of the group and fill in the required fields.



6. Select the membership owner and required members and then click *Create*, to create the group.



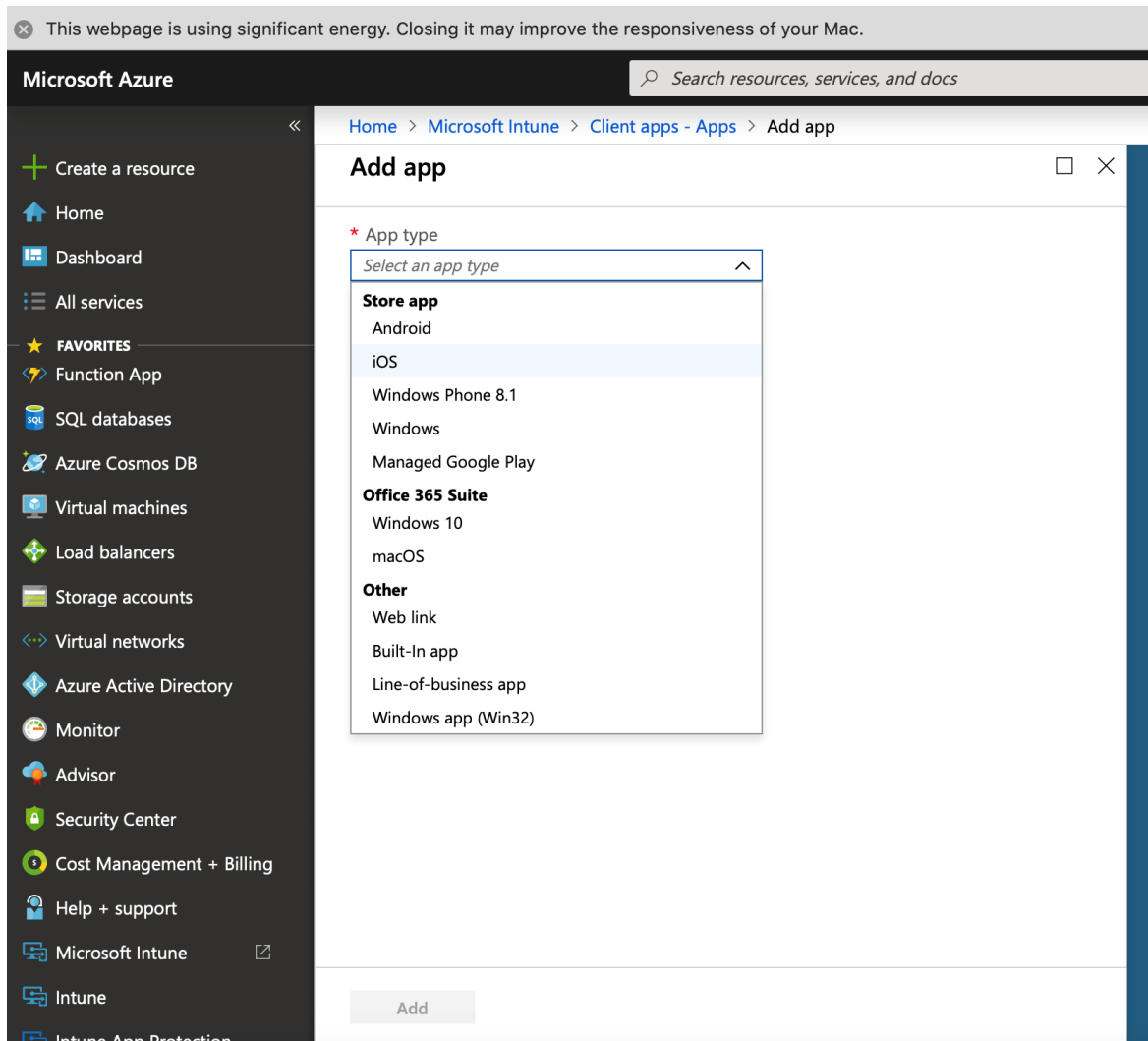
7.3.5.2 Registering the App Store Version of the Application into Microsoft Intune

You must register the App Store version of the application into Microsoft Intune.

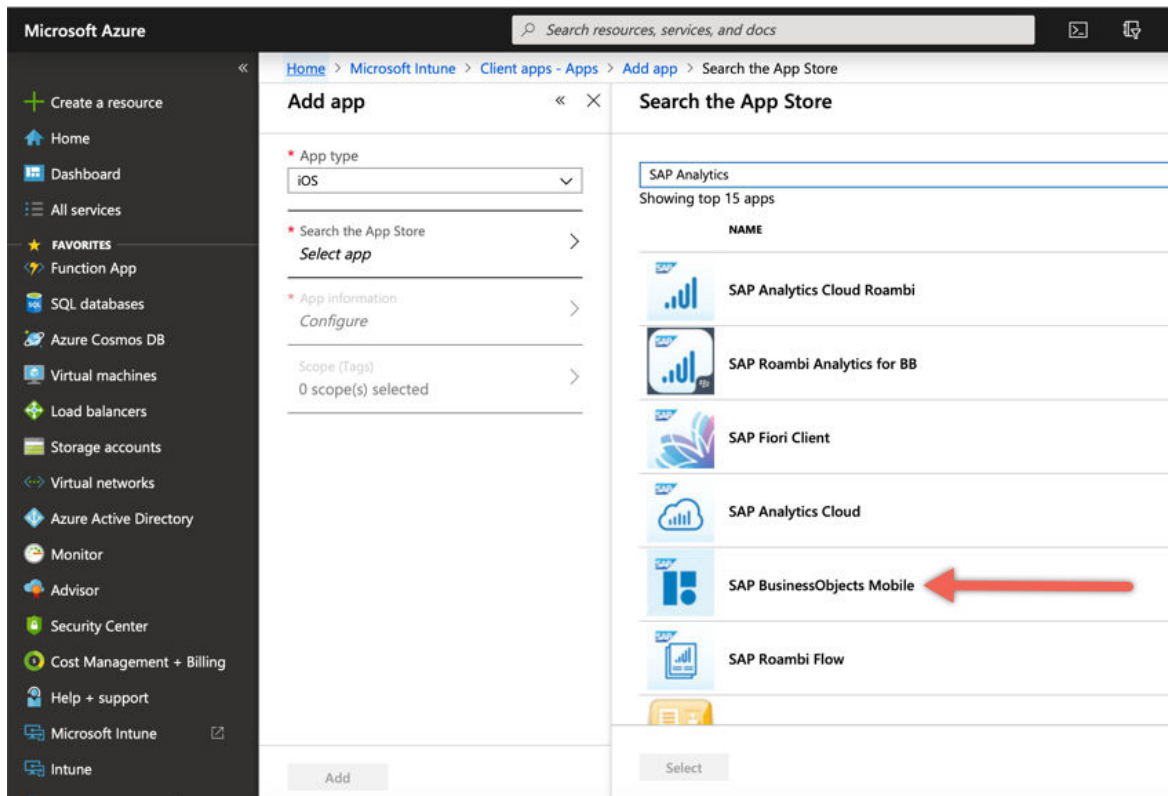
1. After logging in to the Microsoft Azure portal, find the SAP Business Objects Mobile Application on the Intune UI by navigating to **Intune > Client apps > Apps > Add**.

| NAME | TYPE | STATUS | ASSIGNED |
|---------------------------------|--------------------------|--------|----------|
| Analytics-2.48 | iOS line-of-business app | | Yes |
| Analytics-2.51 | iOS line-of-business app | | Yes |
| Microsoft Intune Company Portal | iOS store app | | No |
| SAP Analytics Cloud | iOS store app | | Yes |
| SAP BI SDK | iOS line-of-business app | | Yes |
| SAP BusinessObjects Mobile | iOS store app | | Yes |

2. Select **iOS** from the dropdown list to select the AppStore version of the application.



3. In the search bar, search for SAP BusinessObjects Mobile and the application in App Store.
4. Fill in the required fields and click *Add*.

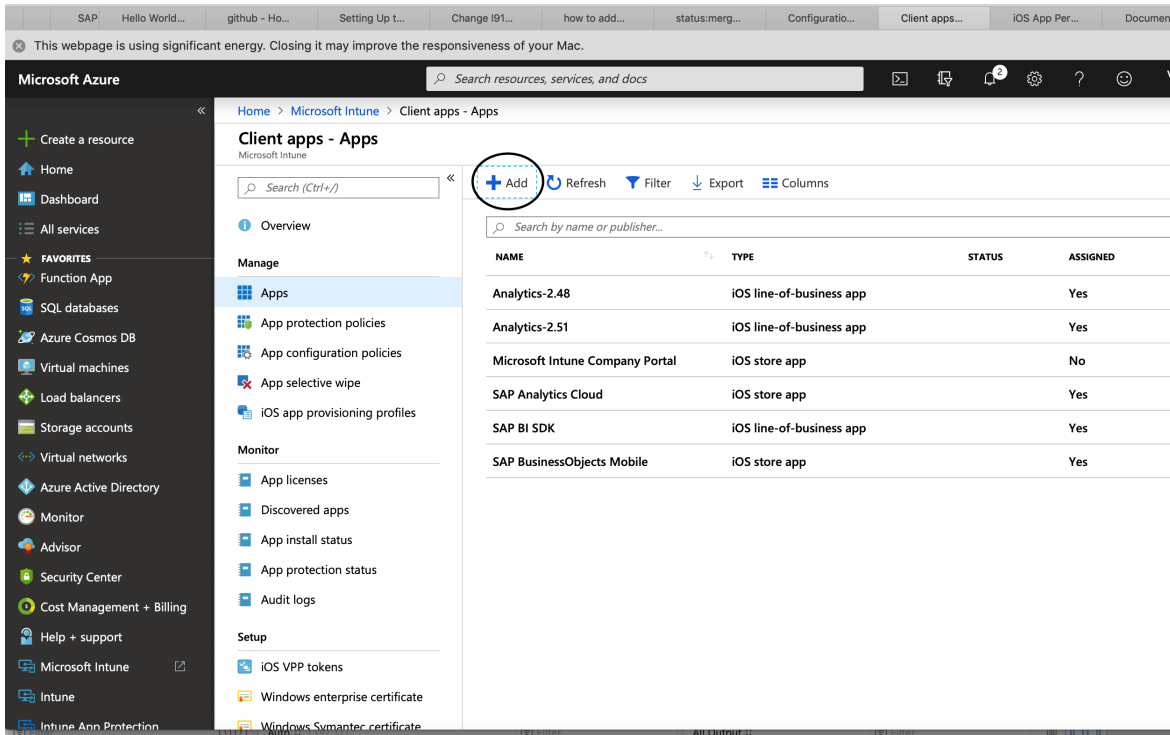


The application is added to the catalogue.

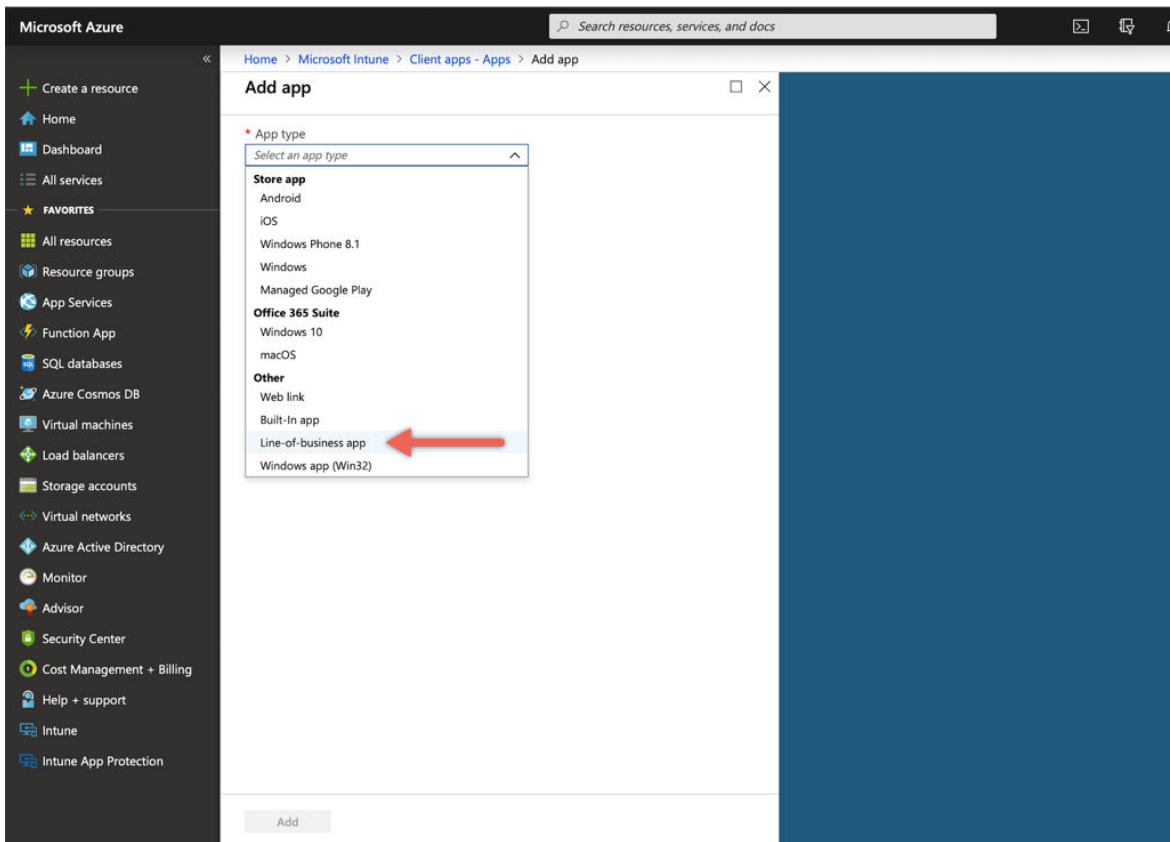
7.3.5.3 Registering the SDK Version of the Application into Microsoft Intune

As a prerequisite, download the SAP BusinessObjects Mobile SDK file.

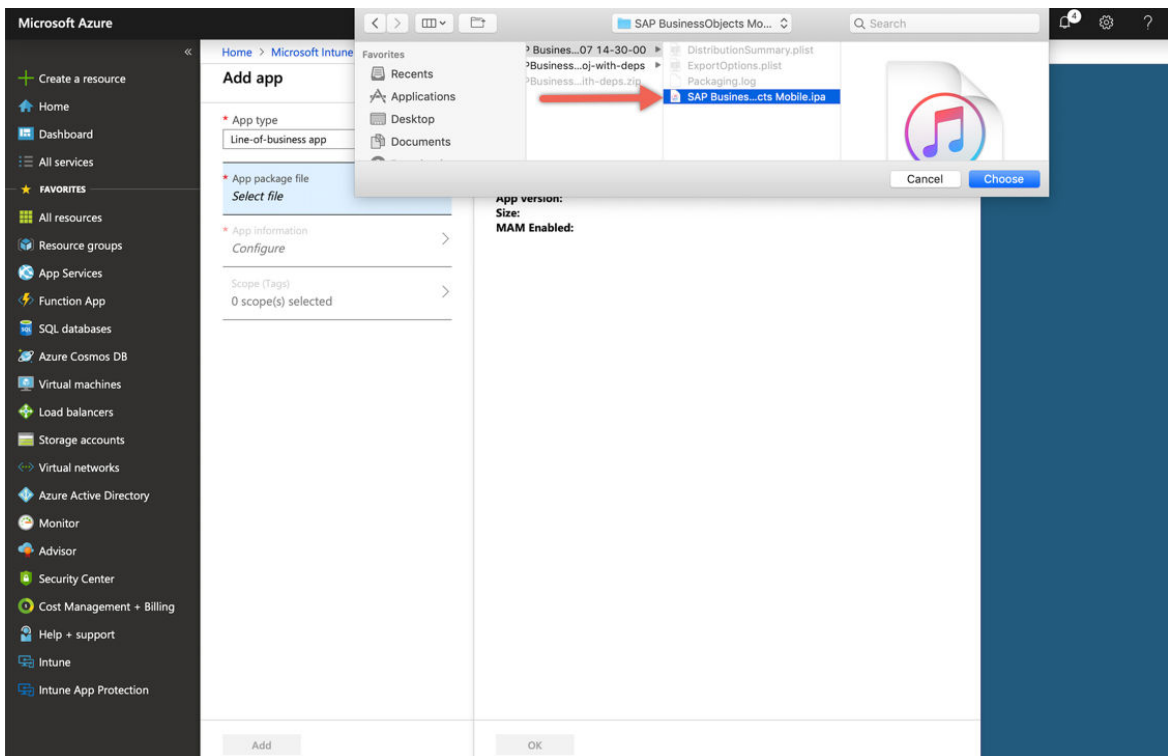
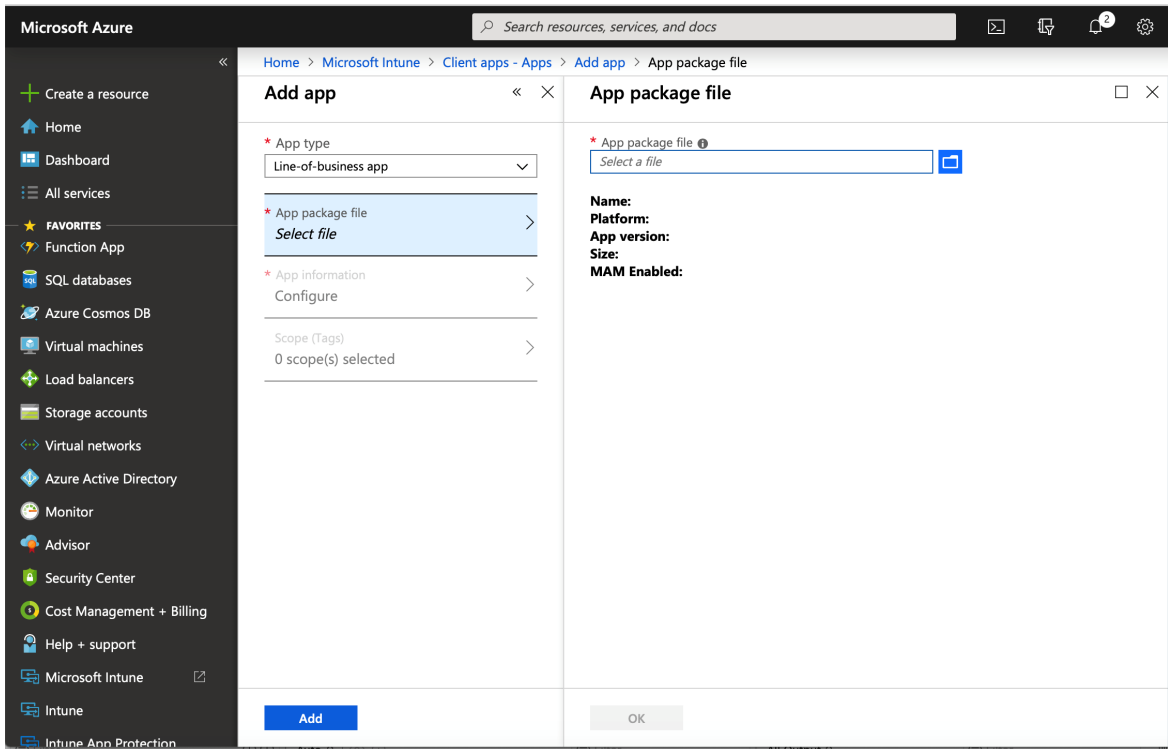
1. After logging in to the Microsoft Azure portal, navigate to **Intune** > **Client Apps** > **Apps**, and then add SAP Business Objects Mobile.



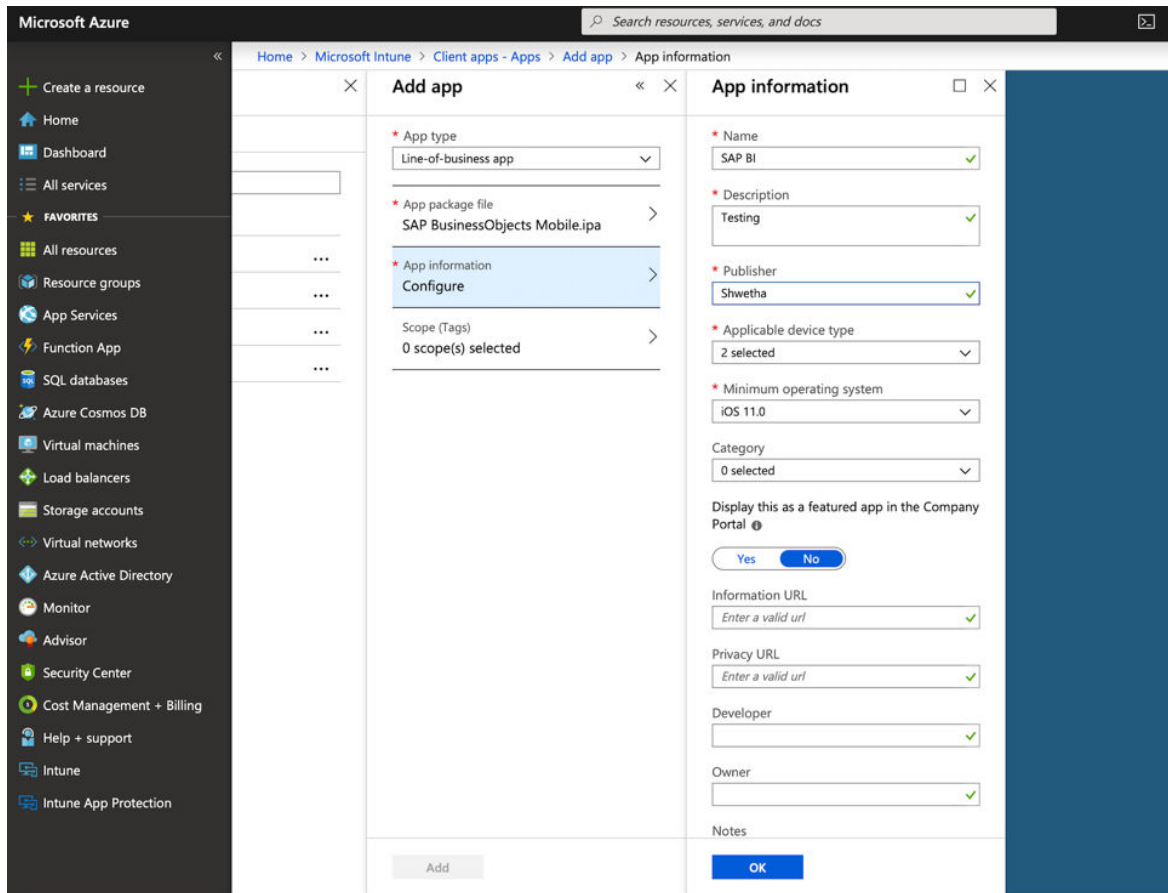
2. Select *Line-of-business app* from the *App type* dropdown list.



3. Select the .ipa file of the selected SDK, and click *Add* to upload the IPA to the portal.



4. Fill in the required fields and click **OK**.

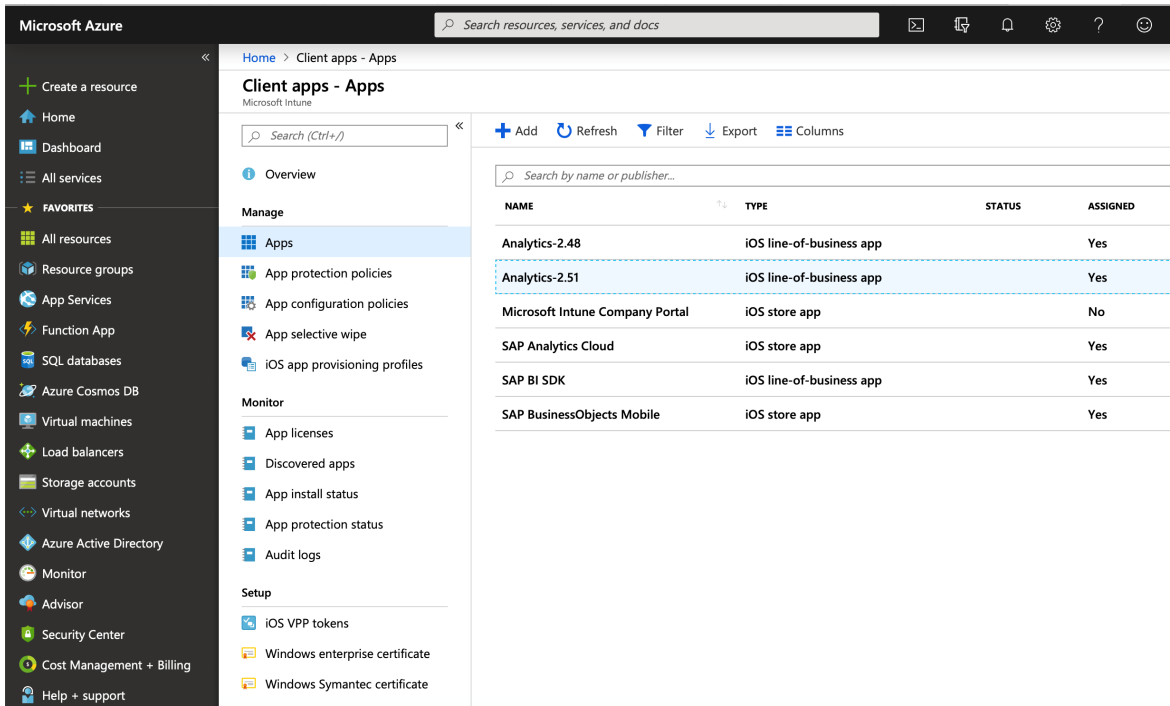


While navigating through the client apps, both the App Store and SDK applications should be listed.

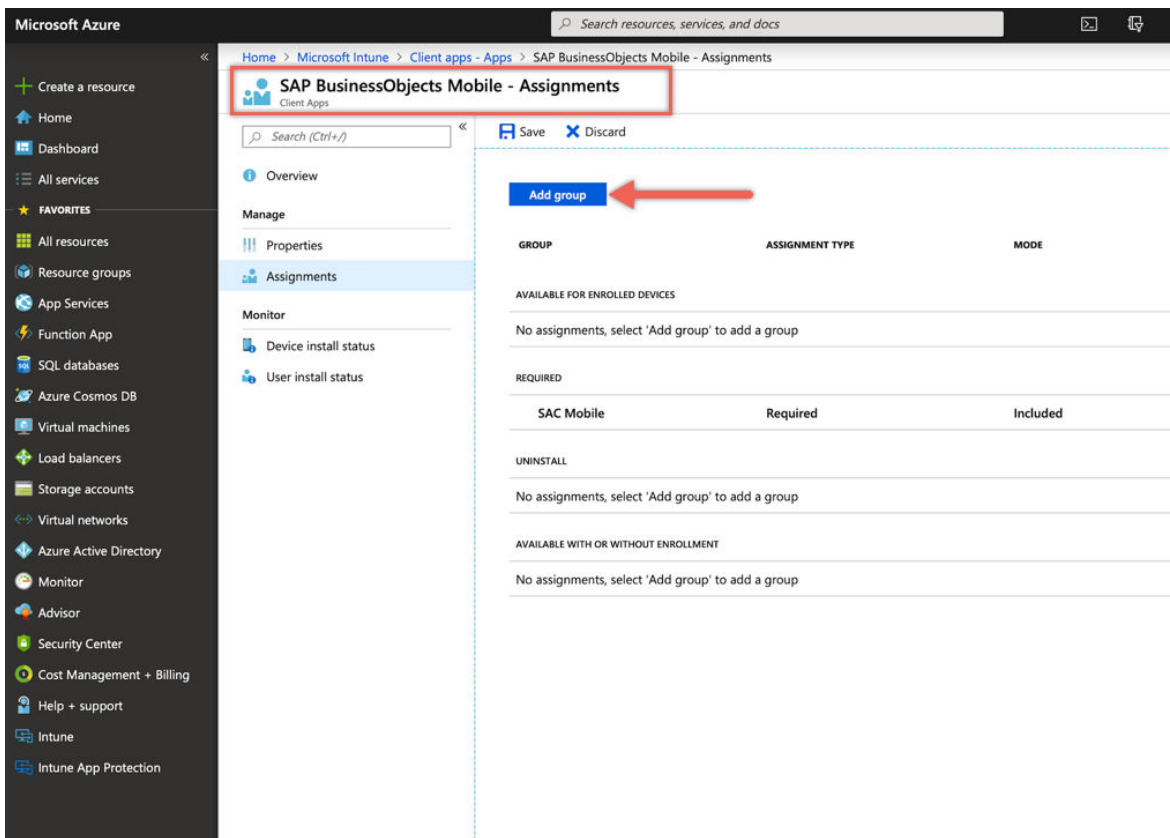
7.3.5.4 Including the Group to an Application

The uploaded applications of the App Store and SDK are listed under *Client Apps*. For assigning a group to an application, the same steps must be followed for both SDK and App Store Applications.

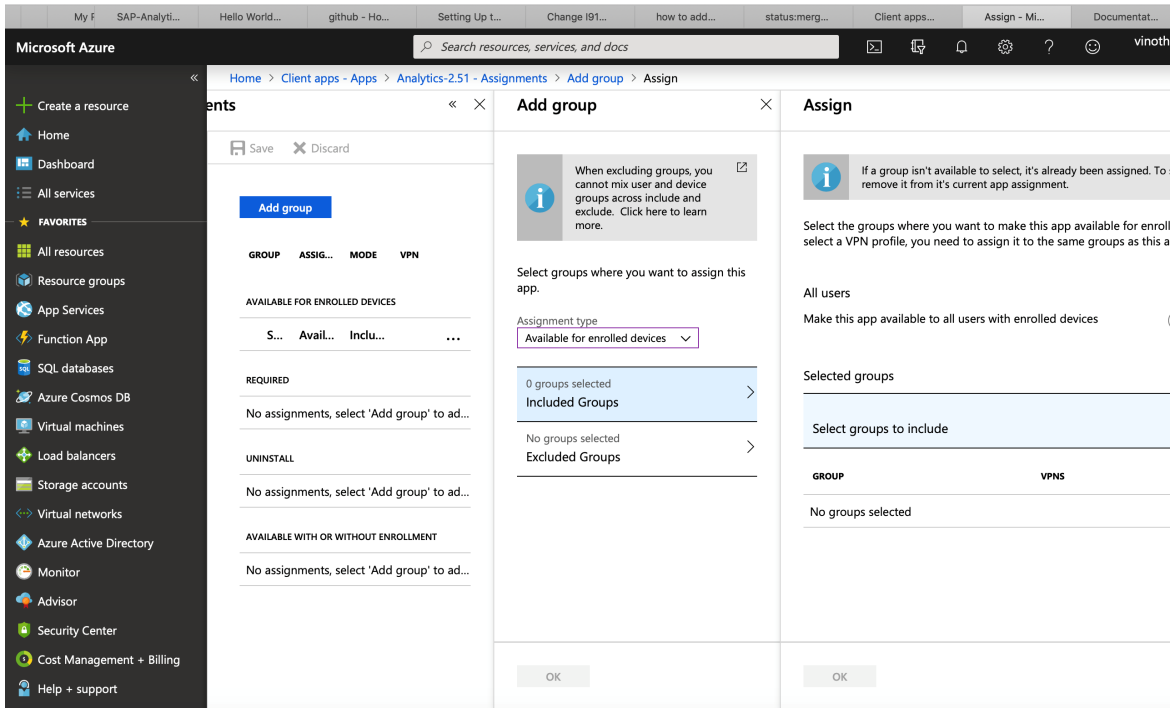
1. From the list of applications, select the application that needs to be assigned to the group.



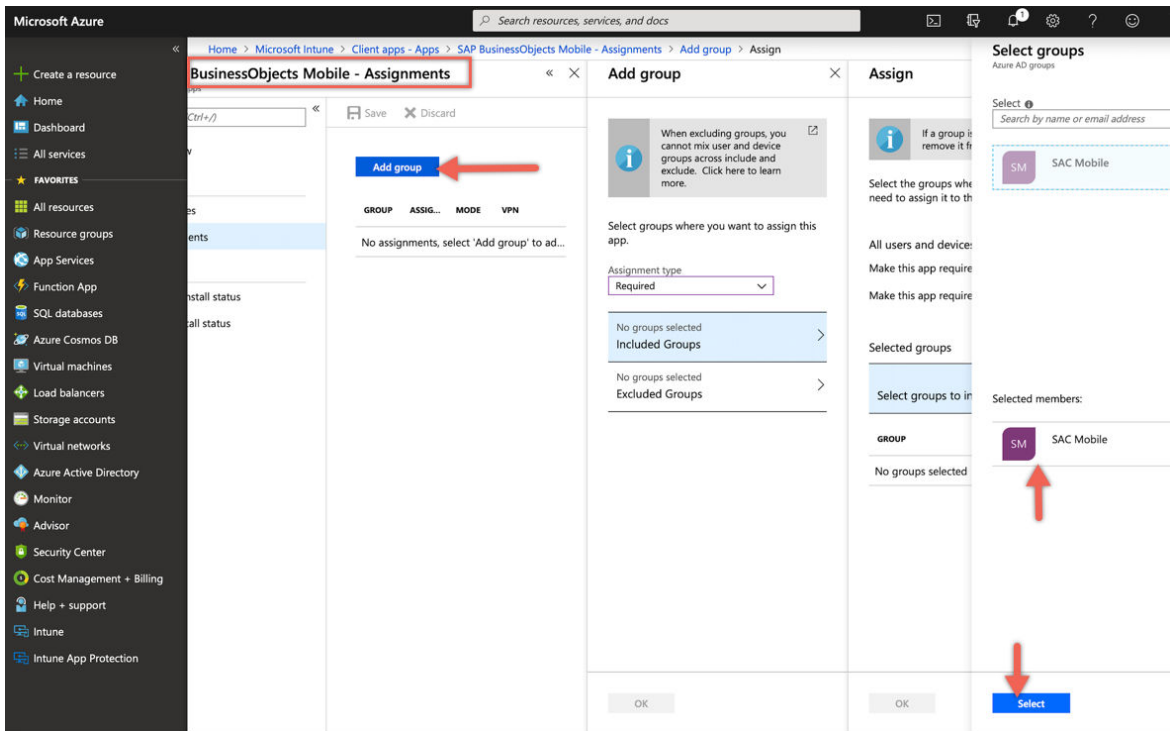
2. Select *Assignments* and click *Add group*, and then select included groups and choose the group to include and selected the group listed.



3. Select the assignment type as *Available for enrolled devices* and include the groups to which the app should be pushed to, on the device.



4. Select the group from the list and click **OK**.



7.3.5.5 Adding Application Configuration Policy in Microsoft Intune

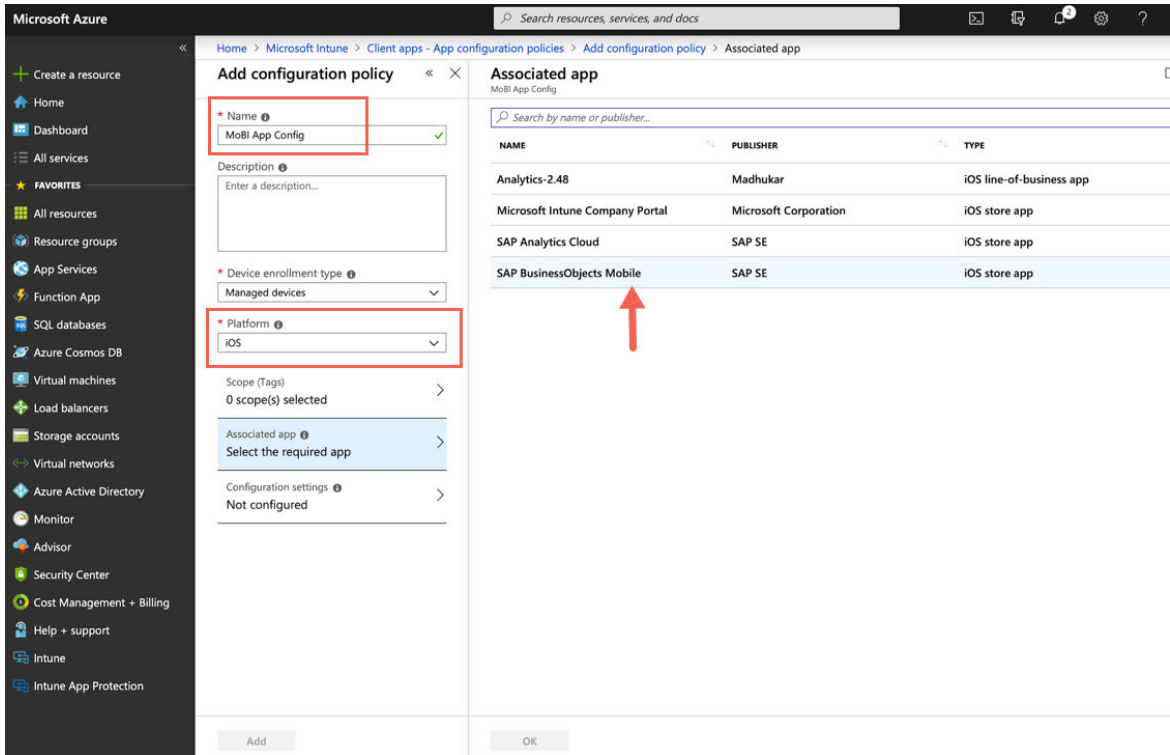
The last step is to add the Application configuration policy in Microsoft Intune.

1. After logging in to the Microsoft Azure portal, navigate to [Intune](#) > [Client Apps](#) > [App Configuration Policy](#) > [Add](#).

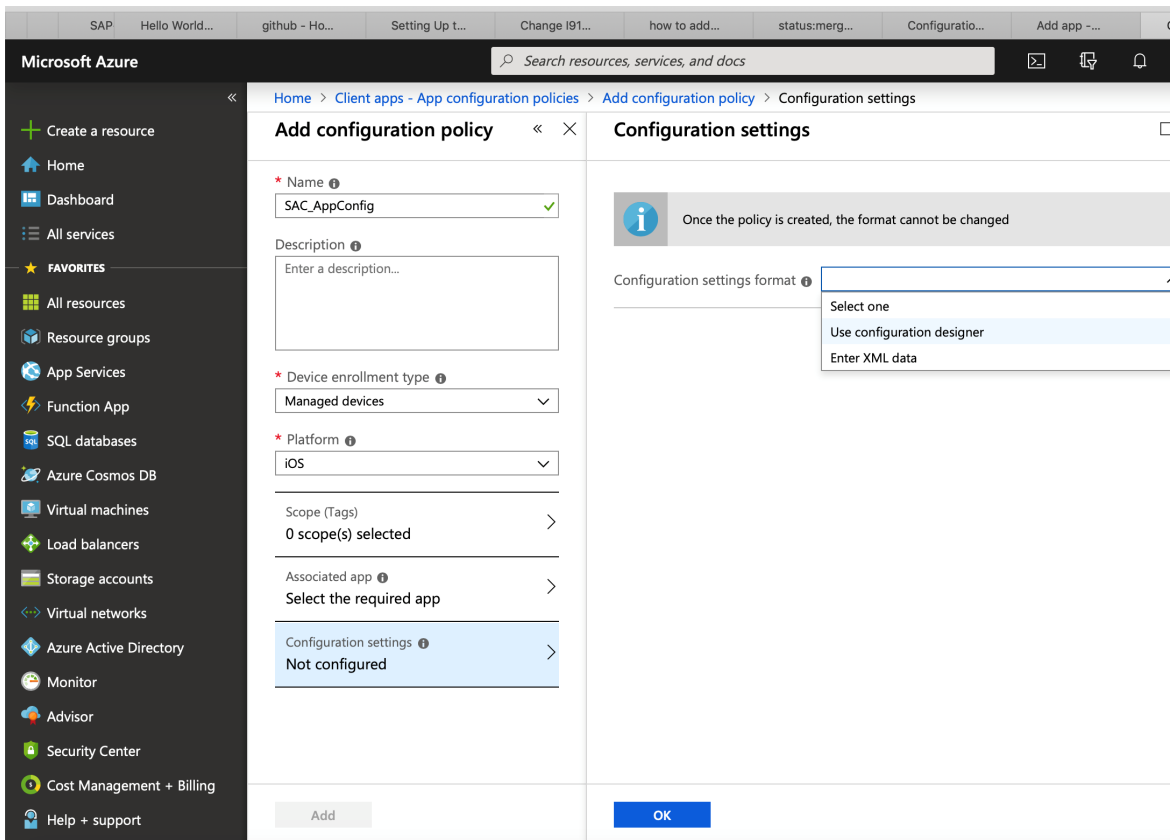
The screenshot displays the Microsoft Azure portal interface for managing Intune policies. The left sidebar contains navigation options like 'Create a resource', 'Home', 'Dashboard', and 'All services'. The main content area is titled 'Client apps - App configuration policies' and includes a search bar and a '+ Add' button. Below this, there are sections for 'Manage' (Apps, App protection policies, App configuration policies, App selective wipe, iOS app provisioning profiles) and 'Monitor' (App licenses, Discovered apps, App install status, App protection status, Audit logs). A 'Setup' section includes 'iOS VPP tokens', 'Windows enterprise certificate', and 'Windows Symantec certificate'. On the right, a table lists a policy named 'SAC_Appstore'.

| NAME | |
|--------------|---|
| SAC_Appstore | i |

2. Give a name to the policy and select *Managed Devices* as the Device enrollment type and the platform as iOS and map the associated application to which the app configuration policy needs to be applied to.



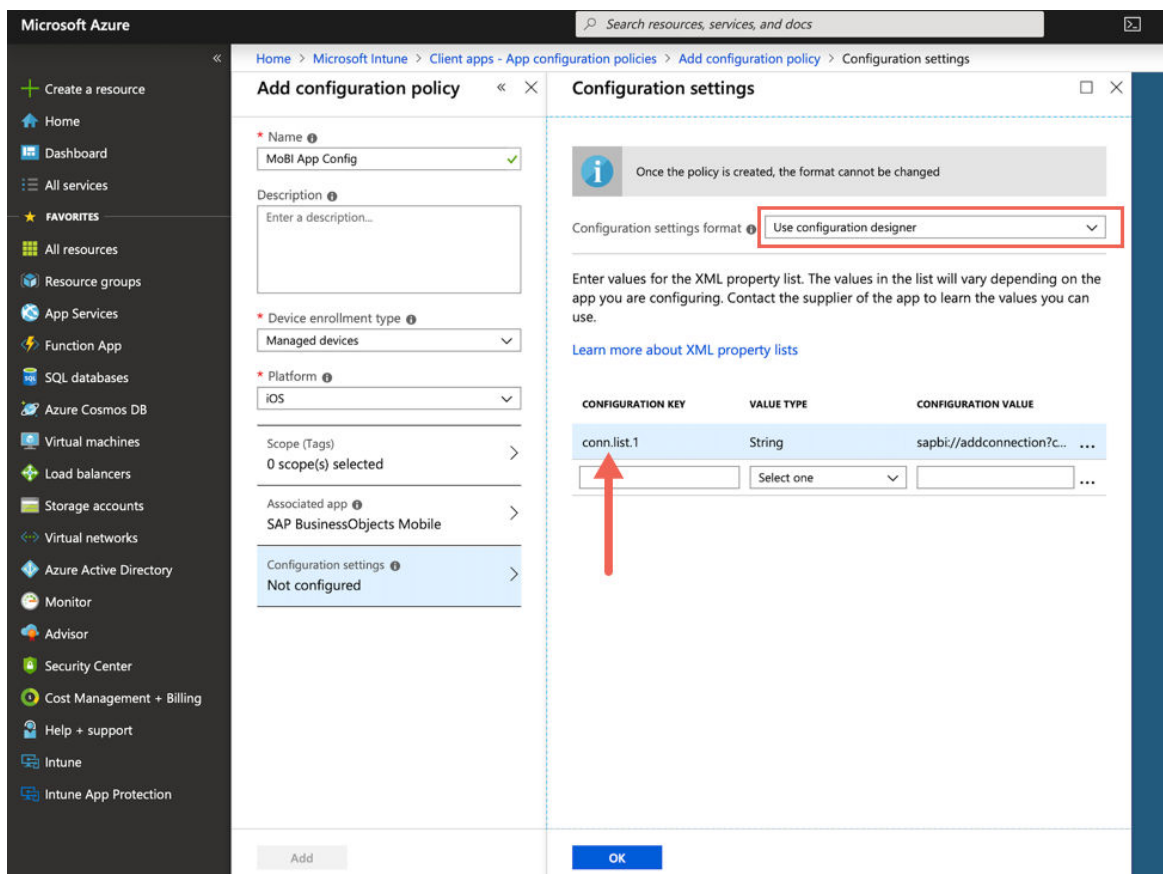
3. In the configuration settings format, select *Use configuration designer* from the dropdown list.



4. Enter the Key-Value pair with corresponding data types.

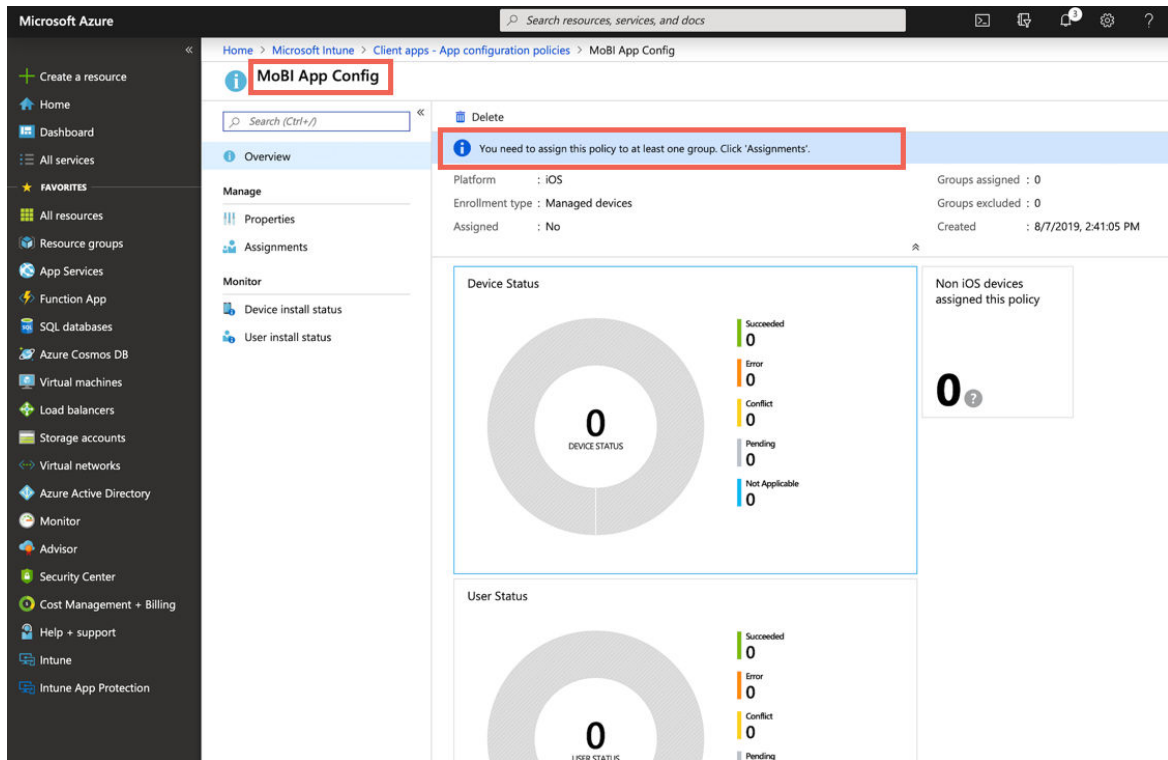
Here the Key is conn.list.1, value is String and configuration value is for adding a connection; that is, `sapbi://addconnection?cms=localhost&server_url=<server URL>&authType=secEnterprise&ConnectionType=BOEConnection&username=<user name>&connection_name=<connection name>`

5. Select the newly created application configuration policy and tap on the *Assignments* tab to assign it to the enrolled devices and corresponding users.



6. Select *All users + All devices* from the dropdown list and click *Save*.

The application configuration policy is now successfully mapped to the corresponding application.



7.3.6 Configuring Microsoft Authentication Library Cross app-SSO with SAP Business Objects Mobile Application

The Cocoa Pods should be installed first. If Cocoa Pods are not installed, use the `sudo gem install cocoapods` command to install the Pods.

Once the Cocoa Pods are installed, follow the procedure below.

1. After downloading the SAP Business Objects Mobile SDK Project, install Pods at xcode level using the `pod init` command. This initializes the pod file.
2. Once Pods are installed, open the pod file, and add MSAL framework as follows.
 - a. Add command `pod 'MSAL'`.
 - b. Comment .
3. Save the Pod file.
4. Now install the Pods using the `pod install` command.

```

# Uncomment the next line to define a global platform for your project
# platform :ios, '9.0'

target 'SAP BusinessObjects Mobile' do
  # Comment the next line if you don't want to use dynamic frameworks
  #use_frameworks!
pod 'MSAL'

  # Pods for SAP BusinessObjects Mobile

end

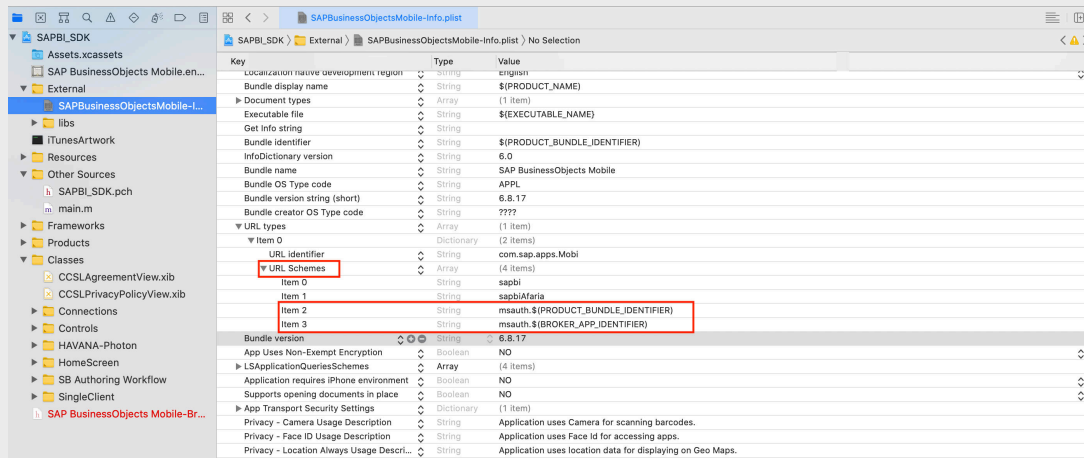
```

5. Once Pods are installed, open *SAPBI_SDK.xcworkspace*.

Note

To receive a call back from the MSAL agent, your application should expose a custom scheme in the *info.plist*. To do so,

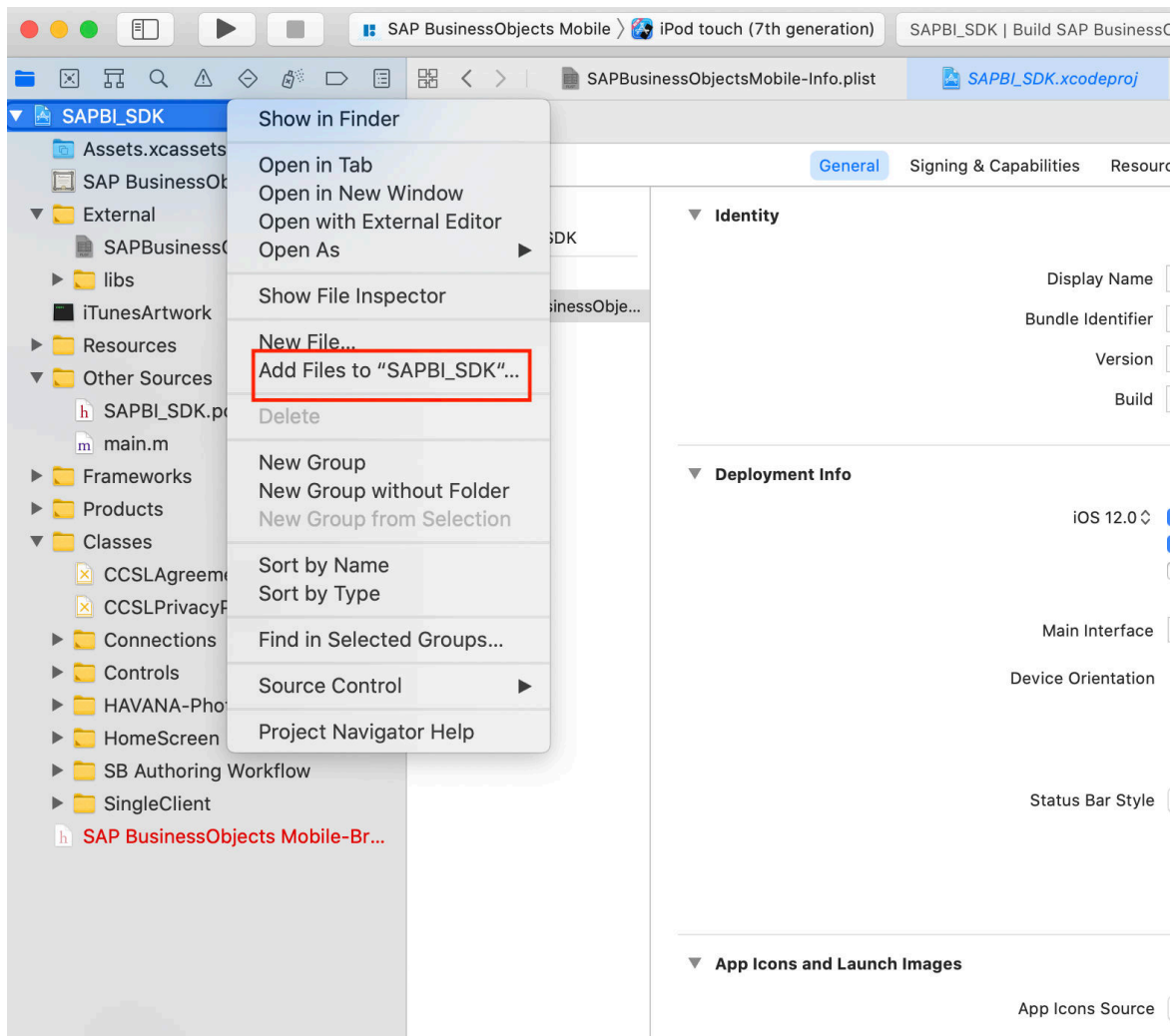
1. Navigate to the Supporting Files in Xcode.
2. Select *SAP BusinessObjectsMobile-Info.plist*.
3. Add the parameters indicated in the screenshot below.
4. Update *URL Schemes*.



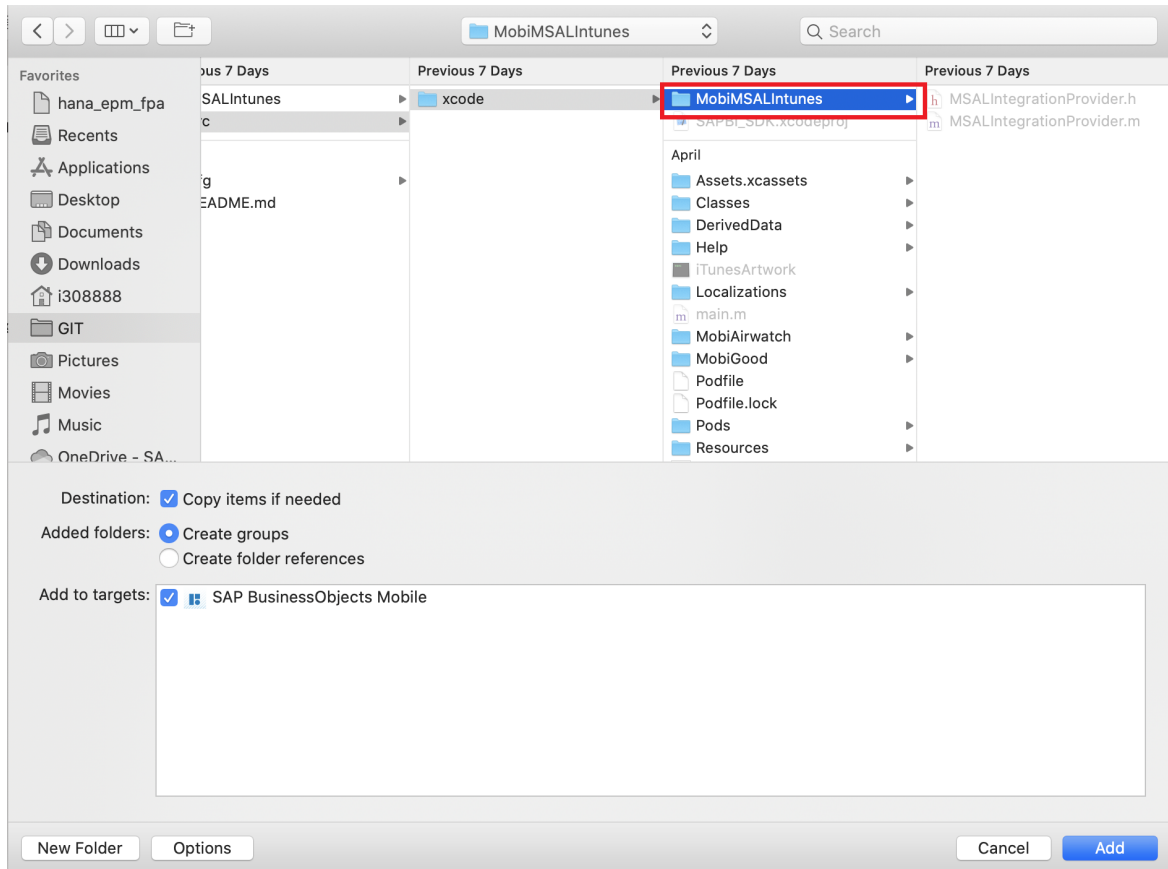
5. Update *LSApplicationQueries schemes*.

| | | |
|------------------------------------|---------|---------------------------|
| Bundle version | String | 6.8.15 |
| App Uses Non-Exempt Encryption | Boolean | NO |
| LSApplicationQueriesSchemes | Array | (7 items) |
| Item 0 | String | afaria |
| Item 1 | String | afaria-getdata |
| Item 2 | String | afaria-enterprise-getdata |
| Item 3 | String | afaria-appstore-getdata |
| Item 4 | String | msauth |
| Item 5 | String | msauthv2 |
| Item 6 | String | msauthv3 |

6. Change *Provider Name* to MSALIntegrationProvider.
7. Follow the steps below in MSAL, and integrate the project.
8. Right click on *Project*, and add files as follows.



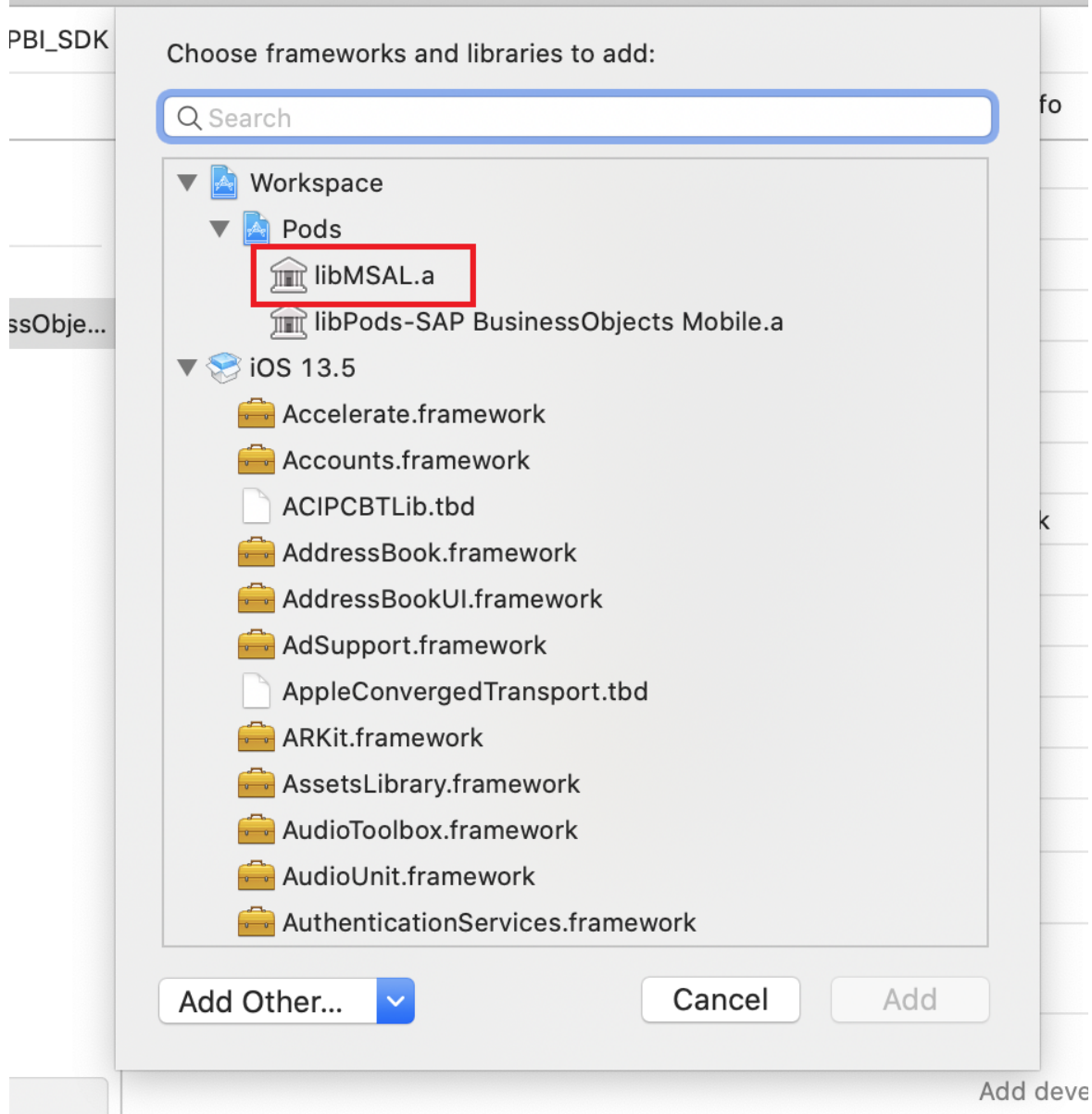
- a. Select the *MobiMSALIntunes* folder from the project.
- b. In the *Add to targets* field, select *SAP BusinessObjects Mobile*.
- c. Click *Add*.



Once the files are added, go to **General Project Settings** > **Frameworks, Libraries, and Embedded Content**.

Click the  Symbol.

Add *libMSAL.a* to the Project.



After all the preceding steps are complete, clean the Build and run the project.

Once we run the project, *SAP BI Mobile Application* is launched.

8 Customizing Mobile Dashboards

You can customize the Dashboards for SAP BusinessObjects Mobile application by using the Dashboards SDK. For more information, refer to the Dashboards Component SDK documentation, available on the SAP Help Portal (<http://help.sap.com/bodash41>).

9 Configuring Application's Behavior Using the SAP BusinessObjects Mobile Server

9.1 Configuring the Behavior of the Application by Setting Properties on the Server

Prerequisites

1. You have the SAP BusinessObjects Mobile server and the SAP BusinessObjects BI platform server in your network.
2. You have the SAP BusinessObjects Mobile (SAP BI) application for iOS installed on your iPhone/ iPad device.

You can configure some chief behavioral aspects of the application by configuring the client settings on the SAP BusinessObjects Mobile server. You configure client settings using the `clientsettings.properties` file on the Mobile server. If you are using the BI platform 4.1, you can also use the **Central Management Console (CMC)** interface for configuring client settings.

Note

- The location of the `clientsettings.properties` file on the SAP BusinessObjects Mobile server is based on the server version you are using. To find the file path for your specific server version, refer to the related topic of this chapter.
- For information on how to configure client settings using the CMC, see the related topic of this chapter.

The following is a typical default configuration of the `clientsettings.properties` file:

```
savePassword=false  
offlineStorage=false  
offlineStorage.ttl=365  
offlineStorage.appPwd=true
```

You can add the following features to the file if you need to customize some of these based on your requirements:

- `feature.email.enabled`
- `feature.collaboration.enabled`
- `feature.annotation.enabled`
- `feature.streamwork.enabled`
- `feature.jam.enabled`
- `feature.jam.url`
- `feature.jam.consumer.key`

- `feature.jam.consumer.secret`
- `feature.jam.callback.urlverb`
- `feature.help.enabled`
- `feature.help.url`
- `feature.help.video.enabled`
- `feature.help.video.playlistfeed`
- `feature.webi.refresh.enabled`
- `feature.webi.reportTitle.enabled`
- `feature.rel.legacy.mode`
- `feature.webi.default.palette.enabled`
- `feature.dashboard.sup.enabled`
- `feature.qrcode.enabled`
- `feature.show.samples.enabled`
- `feature.webi.personal.view.update.enabled`
- `feature.home.doc.person.enabled`
- `feature.info.doc.person.enabled`

For information on the significance and impact of the features above on the application, see the related topic of this chapter.

Related Information

[Customizing the Application Functionality \[page 33\]](#)

[Customizing the Protocols \[page 50\]](#)

[Mobile Server and Property File Locations Matrix \[page 102\]](#)

[Using the CMC to Define Properties on BI Platform 4.1 or 4.2 \[page 103\]](#)

9.2 Configuring the Session Timeout Duration of the Application

The number of concurrent sessions running on the SAP BusinessObjects Mobile server can impact its performance.

If you want a session of an SAP BI mobile application to be terminated after a defined period of inactivity, perform the following configuration on the Mobile server:

1. Open the `web.xml` file found at the following location:
`<Webapp-server-root>\webapps\MobileBIService\WEB-INF`

i Note

The `<WebApp-server- root>` is the root folder location specific to your deployed application server, for example: `C:\Program Files (x86)\SAP BusinessObjects\Tomcat6`

- Set the value of the `session-timeout` parameter found within the tags shown below:

```
<session-config>
<session-timeout>60</session-timeout>
</session-config>
```

i Note

- The `session-timeout` element defines the default session timeout interval for all sessions created in `MobileBIService` (Mobile server).
- The timeout duration begins when the device is locked or the application is sent to the background (user switches to other applications on the device).
- When a session times out, the user is logged off from the BI platform.
- The specified timeout must be expressed as a whole number of minutes. In the example above, the timeout is 60 minutes.
If the timeout is zero ('0') or less, sessions never time out.

9.3 Configuring the Thumbnail Background Image of BI Documents

Based on your requirements, you can customize the thumbnail image for BI documents that appear as tiles on the *Home* screen of the application.

If you do not configure a thumbnail image for BI documents on the server, the documents display a default image (based on document type) on the Home screen. Once you open the document and return to the Home screen, the default background image of the thumbnail is replaced by a snapshot of the first report of the document. The figure below illustrates this behavior for a Web Intelligence document:



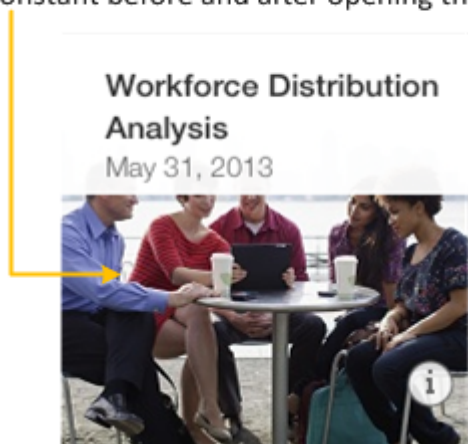
Default thumbnail for an unopened Web Intelligence document



Thumbnail after the Web Intelligence is opened at least once

However, once configured on the server (BI LaunchPad), the thumbnail background image remains **constant** on the Home screen. It does not change even after you open the BI document and return to the Home screen:

**Configured thumbnail background image
(remains constant before and after opening the BI document)**



To configure the thumbnail background image for a BI document, perform the following steps:

1. Access the [Properties](#) of the document on the BI platform (BI LaunchPad/Web Intelligence Rich client).
2. Enter the URL of your image in the Keywords field and choose [Save](#).

i Note

1. The recommended thumbnail image size for best results on the device screen is 208 pixels * 208 pixels.
2. The background image size should not be greater than one hundred Kilobyte (100 KB).
3. Valid image formats include .jpeg, .jpg and .png.
4. If multiple image URLs are specified in the BI document's Keywords field on the BI platform, the application considers only the last valid image URL for the thumbnail background image.
5. If there are multiple strings specified in the Keywords field, and if the image URL is not the last keyword string, a space is required between the URL and the comma (for example: keyword1, http://www.xyz.com/myImage.png , keyword2, keyword3).

9.4 Configuring Document-Specific Help

For an open BI document on the device, when you tap [Help](#) in the secondary report toolbar, the Mobile knowledge center opens by default.

If you want to replace this Help URL with some other URL based on your specific requirements for the BI document, you can do so by performing the procedure below:

1. Access the [Properties](#) of the document on the Mobile BI platform (BI LaunchPad/ Web Intelligence Rich client).
2. Enter the URL of your custom Help in the Keywords field with the syntax below:
docHelpURL=<new_URL>

new_URL is your custom URL, for example, <http://www.myHelpURL.com>

3. Choose *Save*.

i Note

In the current release of the application, you can use a BI document's Keywords field to customize either the document Help or the thumbnail image. However, you cannot implement both of these functionalities at the same time.

Related Information

[Configuring the Thumbnail Background Image of BI Documents \[page 99\]](#)

10 Mobile Server and Property File Locations Matrix

| SAP BusinessObjects Mobile server version | File location of clientsettings.properties and mobi.properties |
|--|---|
| BI 4.0, SP5 and lower versions | <WebApp root>\webapps\MobileBIService \WEB-INF\ |
| BI 4.0, SP6 | <WebApp root>\webapps\MobileBIService \WEB-INF\config\default |
| BI 4.1 and higher versions | <WebApp root>\webapps\MobileBIService \WEB-INF\config\default |
| BI 4.2 and higher versions | <WebApp root>\webapps\MobileBIService \WEB-INF\config\default |

i Note

The <WebApp root> is the root folder location specific to your deployed application server, for example: C:\Program Files (x86)\SAP BusinessObjects\Tomcat6

11 Using the CMC to Define Properties on BI Platform 4.1 or 4.2

If you have deployed SAP BusinessObjects mobile server 4.1 or 4.2 during your setup, you can use the Central Management Console (CMC) to configure client settings and mobile properties. These parameters are otherwise configured through the `clientsettings.properties` and `mobi.properties` files on the Mobile server, respectively.

You need to explicitly define the following parameter in the `MobileBIService\WebContent\WEB-INF\web.xml` file, so that the application can identify whether the Mobile configuration is defined on the CMC (BI platform) or in the properties files on the Mobile server:

```
<context-param>
<description>local-if configuration on server, boe-if configuration in CMC</
description>
  <param-name>mobile.server.configuration.location</param-name>
  <param-value>boe</param-value>
</context-param>
```

If you specify the `context-param` value as `boe` in the file above, your mobile server configuration on the CMC is considered.

To access mobile properties on the CMC, perform the following steps:

1. Open the *Central Management Console* on the BI platform server and access *Applications*. You'll find the *SAP BusinessObjects Mobile* application listed on the page.
2. Right-click *SAP BusinessObjects Mobile* and choose *Properties*. You'll see the following window:

| Mark for deletion | Key | Value |
|--------------------------|---------------------------------|----------------|
| <input type="checkbox"/> | default.corporateCategory | Mobile |
| <input type="checkbox"/> | default.personalCategory | Mobile |
| <input type="checkbox"/> | default.category.mobileDesigned | MobileDesigned |
| <input type="checkbox"/> | default.category.secure | Confidential |
| <input type="checkbox"/> | default.category.featured | Featured |
| <input type="checkbox"/> | default.imageSize | 1048576 |
| <input type="checkbox"/> | default.save.maxPages | 20 |

+ Add More...

3. Edit the category values in *Properties* as required.

i Note

- For information on the various categories, see the topic *Configuring Categories on the Mobile Server of the Administrator and Report Designer's guide* available on the SAP Help Portal.

4. To edit client settings, choose *Client Settings*:

| Mark for deletion | Key | Value |
|--------------------------|-----------------------|-------|
| <input type="checkbox"/> | savePassword | false |
| <input type="checkbox"/> | offlineStorage | false |
| <input type="checkbox"/> | offlineStorage.ttl | 365 |
| <input type="checkbox"/> | offlineStorage.appPwd | true |

+ Add More...

5. Save your property settings and exit from the CMC.

i Note



The **negative** mobile application **rights** that are granted via the CMC take precedence over the values specified for client settings via the CMC. For example, if the right To save documents to the local store of a device is not granted to the user in the CMC, users cannot download BI documents to their device, even if the `offlineStorage` property is set to `true` in the `client settings`. Also note that if the right to save documents is granted in the CMC, but `offlineStorage=false` in `client settings`, users cannot download BI documents to the mobile device. Downloading/saving BI documents to the device is possible only if the right is granted and `offlineStorage=true`.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
 - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
 - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

© 2020 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see <https://www.sap.com/about/legal/trademark.html> for additional trademark information and notices.