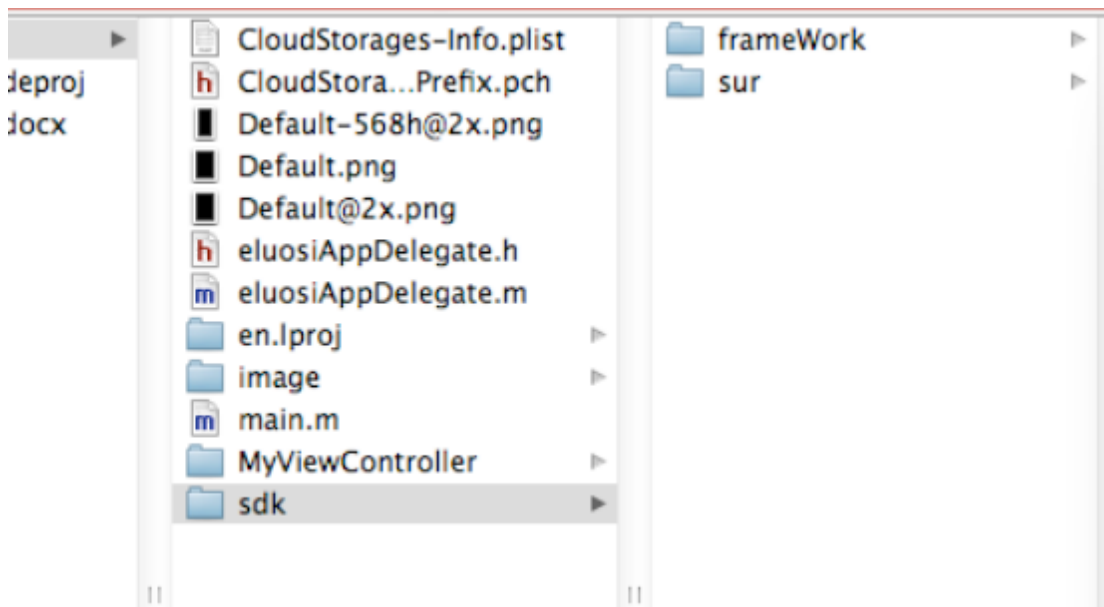
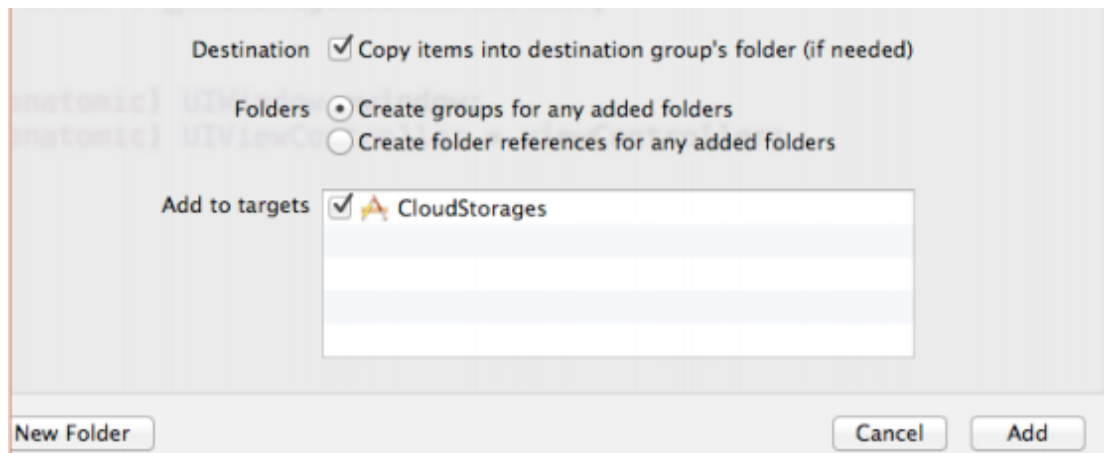


# Cloud Storage SDK Integrate Guidelines



## 2. Add Independent Framework

Open “build phases” page<”project setting”, tap “+” at left bottom in “Link Binary With Libraries”, add the following framework in the pop menu:

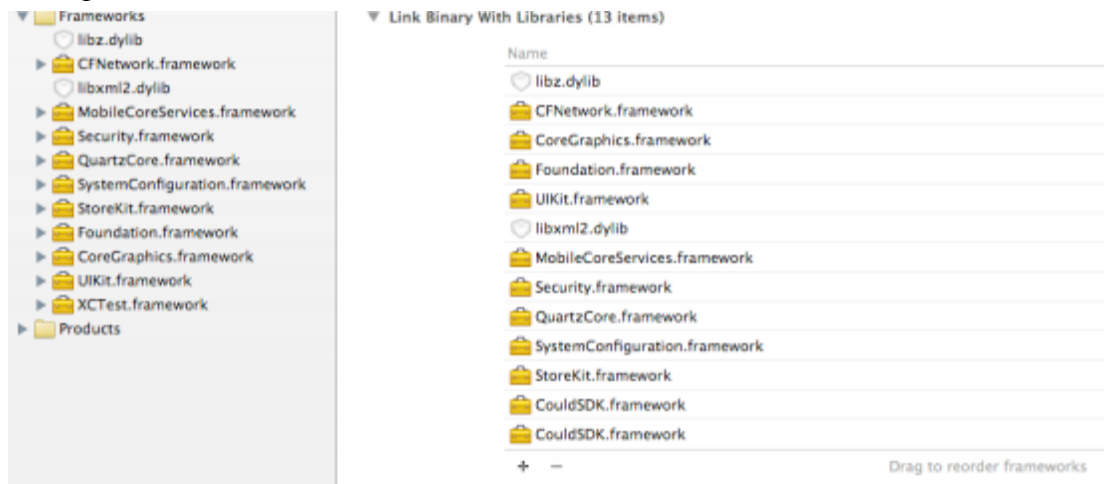
storeKit.framework

SystemConfiguration.framework

QuartzCore.framework

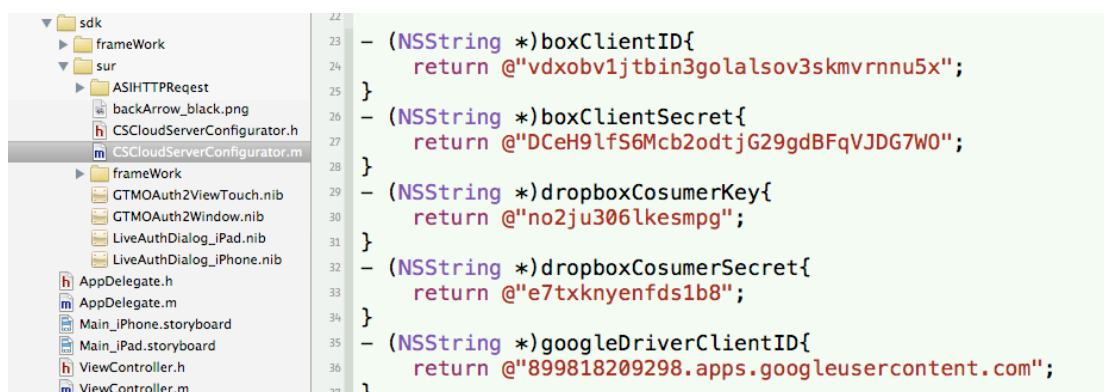
Security.framework  
 MobileCoreServices.framework  
 libxml2.dylib  
 UIKit.framework  
 Foundation.framework  
 CoreGraphics.framework  
 CFNetwork.framework  
 Libz.dylib

Example:



### 3. Set AppKey for All Clouds

Open sdk/sur/CSCloudServerConfigurator.m and set the relevant information (each Cloud AppleKey will get relevant information after its registration on the official website) of each cloud AppleKey. Shown below:



### Remarks

The link Box applies for the ClientID and ClientSecret:

<https://developers.box.com/>;

The link DropBox applies for the ClientSecret:

<https://www.dropbox.com/developers>

The link GoogleDrive applies for the ClientID and ClientSecret:

<https://code.google.com/apis/console/>

The link EverNote applies for the AccessKey and Secret:

<http://dev.evernote.com>

The link SugarSync applies for the AccessKey and privateAccessKey:

<https://www.sugarsync.com/developer/account>

The link skyDrive applies for the ClientID:

<http://dev.live.com>

## 4. Setting URL Schemes

**A.** Open \*-Info. Plist(\* implies your project name), create a new URL type in setting document (omit if it already exists). Open URL types – URL Schemes, add a new scheme under URL Schemes to authorize AppleKey for DropBox db-(DropBoxAppkey). Repeat for other cloud storage schemes.

▼ Information Property List	Dictionary	(17 items)
▼ URL types	Array	(1 item)
▼ Item 0	Dictionary	(2 items)
▼ URL Schemes	Array	(2 items)
Item 0	String	db-no2ju306lkesmpg
Item 1	String	en-eleeditor
URL identifier	String	

**B.** This SDK uses the ASIHTTPRequest library, which does not support ARC documents. Thus the arc of all documents shall be forbidden under ARC mod(-fno-objc-arc)

See image below for details:

ASIDataCompressor.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASICloudFilesContainer.m	...in wwwwww/sdk/sur/ASIHTTPRequest/CloudFiles	-fno-objc-arc
ASIS3ServiceRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/S3	-fno-objc-arc
ASIS3ObjectRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/S3	-fno-objc-arc
ASIHTTPRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASIS3Request.m	...in wwwwww/sdk/sur/ASIHTTPRequest/S3	-fno-objc-arc
ASIAuthenticationDialog.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASICloudFilesContainerXMLParserDelegate.m	...in wwwwww/sdk/sur/ASIHTTPReqes...	-fno-objc-arc
ASIFORMDataRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASIS3BucketRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/S3	-fno-objc-arc
ASICloudFilesObjectRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/CloudFiles	-fno-objc-arc
ASICloudFilesCDNRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/CloudFiles	-fno-objc-arc
ASICloudFilesRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/CloudFiles	-fno-objc-arc
Reachability.m	...in wwwwww/sdk/sur/ASIHTTPRequest/Reachability	-fno-objc-arc
ASICloudFilesContainerRequest.m	...in wwwwww/sdk/sur/ASIHTTPRequest/CloudFiles	-fno-objc-arc
ASIInputStream.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASINetworkQueue.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASIS3BucketObject.m	...in wwwwww/sdk/sur/ASIHTTPRequest/S3	-fno-objc-arc
ASIDataDecompressor.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
ASIS3Bucket.m	...in wwwwww/sdk/sur/ASIHTTPRequest/S3	-fno-objc-arc
ASICloudFilesObject.m	...in wwwwww/sdk/sur/ASIHTTPRequest/CloudFiles	-fno-objc-arc
ASIDownloadCache.m	...in wwwwww/sdk/sur/ASIHTTPRequest	-fno-objc-arc
...	...	...

## 5. AppDelegate.m Files Setting

```
1.in AppDelegate.m, # import
<CloudSDK/CSCloudServer.h>
2.in- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
, add code: [CSCloudServer setCloudConfigurator] ;
3.in- (void)applicationDidBecomeActive:(UIApplication
*)application, add code: [CSCloudServer
handleDidBecomeActive] ;
4.in- (BOOL)application:(UIApplication *)application
openURL:(NSURL *)url sourceApplication:(NSString
*)sourceApplication annotation:(id)annotation{
return [CSCloudServer openURL:url];
}
```