

iOS SDK

The **iOS Apstrata SDK** is a utility that is provided to developers of iOS-based applications. The iOS Apstrata SDK wraps the Apstrata REST APIs making it very easy to call these latter from within Objective C code.

The iOS SDK is provided as source code: an interface "apstrata.h" and an implementation class "apstrata.mm". All you need to do is to add this code to your own and compile it.

- Drop the client library into you code,
- Create an instance of "ApstrataiPhoneClient" passing it the URL of the Apstrata server, the key and secret of the Apstrata account you are using and the expected authentication mode ([more on authentication with Apstrata](#)).
- Call one of the methods exposed by the Android client.

Click [here](#) to download the iOS Apstrata SDK source code. The source code ships with a sample file.

The iOS Apstrata Client provides the following methods:

Method	Description	Parameters	Returned value
callAPIMethod	Invokes an Apstrata api by specifying the api name and parameters. This is the generic serving all-purposes method of the iOS client.	<ul style="list-style-type: none">• methodName: the API operation being invoked (example: "SaveUser", "PushNotification", etc.)• params: a dictionary of key/value entries representing the text parameters expected by the invoked API operation• files: a dictionary of NamedFileData instances associated to keys (this class is defined in the apstrata.h interface). Each instance of NamedFileData contains the name of the file to upload and the corresponding bytes (NSData). Each key represents a field of the targeted document to which the files will be attached. Note that if the document is schema-less, it is only possible to attach files to a unique field named "apsdb_attachments"	A string representing the response sent by the API
callAPIJson	Similar to the above except that it returns the response in JSON format	Same as above. You do not need to specify the expected returned format of the response.	A string representing the response sent by the API, in JSON format
callAPIXml	Similar to the above except that it returns the response in XML format	Same as above. You do not need to specify the expected returned format of the response.	A string representing the response sent by the API, in XML format
callAPIFile	This method is used typically to download a file	<ul style="list-style-type: none">• methodName: identifies the API operation being invoked• params: a list of key/value pairs representing the text parameters expected by the invoked API operation• path: the local path to a folder where to store the downloaded file. If path is null or empty, the downloaded file is not store an only a pointer to an NSData of the file content is returned	An buffer of bytes (NSData*)
getFullApiUrl	Use this method to retrieve the URL of a file attached to a document stored in your account	Same as callAPIMethod.	A string representing the url of the API call (useful to download an attached file)

Examples:

Create an instance of the iOS Apstrata client

```
NSString *url = @"https://sandbox.apstrata.com/apsdb/rest";
NSString *authKey = @"Your_auth_key_goes_here";
NSString *secret = @"You_account_secret_goes_here";
ApstrataiPhoneClient *client = [[ApstrataiPhoneClient alloc] initWithURL: url key: authKey secret: secret
authMode: SIMPLE];
```

Create and execute a Query

```
NSMutableDictionary *parameters = [[NSMutableDictionary alloc] init];
NSDictionary *files = [[NSMutableDictionary alloc] init];

NSString* methodName = @"Query";
[parameters setObject: @"apsdb.documentKey=\\"some_document_key_goes_here\\" forKey: @"apsdb.query"];
[parameters setObject: @"apsdb.documentKey" forKey: @"apsdb.queryFields"];
[parameters setObject: @"LoadTestStore" forKey: @"apsdb.store"];
NSString* result = [client callAPIJson: (NSString *)methodName params:parameters files: files];
```

Save a document with an attached file

```
NSMutableDictionary *parameters = [[NSMutableDictionary alloc] init];
NSDictionary *files = [[NSMutableDictionary alloc] init];

NSString *methodName = @"SaveDocument";
[parameters setObject: @"documentWithFile" forKey: @"apsdb.documentKey"];
NamedFileData *namedFileData = [[NamedFileData alloc] init];
NSData *fileData = [NSData dataWithContentsOfFile:@"/Users/[some_user]/Library/Application Support/iPhone simulator/textfile.txt"];
[namedFileData setFileData:fileData];
[namedFileData setFileName:@"textfile.txt"];
NSString *fileDataArray = [NSArray arrayWithObject:namedFileData];
[files setValue: fileDataArray forKey:@"apsdb_attachments"];
NSString *result = [client callAPIMethod:methodName parameters:parameters files:files];
```

Download an attached file (do not store it locally)

```
NSMutableDictionary *parameters = [[NSMutableDictionary alloc] init];
NSDictionary *files = [[NSMutableDictionary alloc] init];

NSString *methodName = @"GetFile";
NSString *path = @"";
NSString *srcPath = @"/Users/[some_user]/Library/Application Support/iPhone simulator/";
NSString *fileName = @"textfile.txt";
srcPath = [srcPath stringByAppendingString: fileName];
[parameters setObject: @"documentWithFile" forKey: @"apsdb.documentKey"];
[parameters setObject: fileName forKey: @"apsdb.fileName"];
[parameters setObject: @"apsdb_attachments" forKey: @"apsdb.fieldName"];
NSData* fileContent = [client callAPIFile: methodName params:parameters path:path];
NSFileManager *fileManager = [NSFileManager defaultManager];
NSData* initialFileContent = [fileManager contentsAtPath: srcPath];
bool contentMatch = [initialFileContent isEqualToData:fileContent];
if (contentMatch)
    NSLog(@"File was uploaded and matches src file");
else
    NSLog(@"File was not correctly uploaded or does not match src file");
```

Get attached file URL

```
NSMutableDictionary *parameters = [[NSMutableDictionary alloc] init];
NSDictionary *files = [[NSMutableDictionary alloc] init];

NSString *methodName = @"GetFile";
NSString *fileName = @"Ixia.gif";
[parameters setObject: @"documentWithFile" forKey: @"apsdb.documentKey"];
[parameters setObject: fileName forKey: @"apsdb.fileName"];
[parameters setObject: @"apsdb_attachments" forKey: @"apsdb.fieldName"];
NSString* imageUrl = [client getFullApiUrl:methodName parameters: parameters files:files];
```