Connecting to the MobileFirst Server

Overview

To provide the best startup experience, a MobileFirst Hybrid application starts without connecting to the MobileFirst Server. It is up to the developer to decide when should the application connect to the MobileFirst Server.

As an application developer, you are responsible for maintaining the offline or online state within the application, and ensure that the application can recover from failed attempts to connect to the MobileFirst Server. The above is also relevant for Native applications that use the MobileFirst SDK for connecting to the MobileFirst Server.

Connecting to the MobileFirst Server ensures a single HTTP session between client and server communications, and thus minimizes the risk of a request race condition where a session doesn't exist yet and several requests are sent simultaneously from the client to the server. Hence it is recommended to establish a connection with the MobileFirst Server before any other interaction.

Connecting to the MobileFirst Server ensures that APIs that return authentication-related data such as `getUserName()` or `isAuthenticated()` will have the necessary information retrieved from the server.

API Signatures

JavaScript

```
WL.Client.connect(
  {
    onSuccess: successCallback
,
    onFailure: failureCallback
  }
);
```

Objective-C

```
[[WLClient sharedInstance] wlConnectWithDelegate:myConnectListener];
```

Where `myConnectListener` is an instance of a class that conforms to the `WLDelegate` protocol, with `onSuccess` and `onFailure` methods.

Swift

```
WLClient.sharedInstance().wlConnectWithDelegate(myConnectListener)
```

Where `myConnectListener` is an instance of a class that conforms to the `WLDelegate` protocol, with `onSuccess` and `onFailure` methods.

Java

```
WLClient.createInstance(this).connect(myConnectListener);
```

Where `myConnectListener` in an instance of a class that implements `WLResponseListener`, with `onSuccess` and `onFailure` methods.

C#

```
WLClient.getInstance().connect(myConnectListener);
```

Where `myConnectListener` in an instance of a class that implements `WLResponseListener`, with `onSuccess` and `onFailure` methods.

> For detailed examples of connecting to the MobileFirst Server,see the Invoking Adapter Procedures tutorials (../../server-side-development/).

When to use the `WL.Client.connect` API

It is recommended to use the `WL.Client.connect` API in the following cases:

- A connection should always be initiated before making the first request of any type to the MobileFirst Server. Subsequent requests should be made only after the connect's `onSuccess` is invoked.
- You should initiate a new connection request after long periods of inactivity where session timeout might have occurred, such as application returning from background. Subsequent requests should be made only after the connect's `onSuccess` is invoked.