AT&T Developer Program

AT&T Adapters for IBM® Worklight

Installation and Setup Guide

Revision Date **December 24, 2012**

Legal Disclaimer

This document and the information contained herein (collectively, the "**Information**") is provided to you (both the individual receiving this document and any legal entity on behalf of which such individual is acting) ("**you**" and "**your**") by AT&T, on behalf of itself and its affiliates ("**AT&T**") for informational purposes only. AT&T is providing the Information to You because AT&T believes the Information may be useful to you. The Information is provided to You solely on the basis that you will be responsible for making your own assessments of the Information and are advised to verify all representations, statements and information before using or relying upon any of the Information. Although AT&T has exercised reasonable care in providing the Information to you, AT&T does not warrant the accuracy of the Information and is not responsible for any damages arising from your use of or reliance upon the Information. You further understand and agree that AT&T in no way represents, and you in no way rely on a belief, that AT&T is providing the Information in accordance with any standard or service (routine, customary or otherwise) related to the consulting, services, hardware or software industries.

AT&T DOES NOT WARRANT THAT THE INFORMATION IS ERROR-FREE. AT&T IS PROVIDING THE INFORMATION TO YOU "AS IS" AND "WITH ALL FAULTS." AT&T DOES NOT WARRANT, BY VIRTUE OF THIS DOCUMENT, OR BY ANY COURSE OF PERFORMANCE, COURSE OF DEALING, USAGE OF TRADE OR ANY COLLATERAL DOCUMENT HEREUNDER OR OTHERWISE, AND HEREBY EXPRESSLY DISCLAIMS, ANY REPRESENTATION OR WARRANTY OF ANY KIND WITH RESPECT TO THE INFORMATION, INCLUDING, WITHOUT LIMITATION, ANY REPRESENTATION OR WARRANTY OF DESIGN, PERFORMANCE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, OR ANY REPRESENTATION OR WARRANTY THAT THE INFORMATION IS APPLICABLE TO OR INTEROPERABLE WITH ANY SYSTEM, DATA, HARDWARE OR SOFTWARE OF ANY KIND. AT&T DISCLAIMS AND IN NO EVENT SHALL BE LIABLE FOR ANY LOSSES OR DAMAGES OF ANY KIND, WHETHER DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, SPECIAL OR EXEMPLARY, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF GOODWILL, COVER, TORTIOUS CONDUCT OR OTHER PECUNIARY LOSS, ARISING OUT OF OR IN ANY WAY RELATED TO THE PROVISION, NON-PROVISION, USE OR NON-USE OF THE INFORMATION, EVEN IF AT&T HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES OR DAMAGES.

Table of Contents

1. Introduction

The AT&T Adapters for IBM Worklight (‘AT&T Adapters’) provide a simplified way for Worklight developers to access the AT&T API platform services.

By significantly reducing the complexity of building applications that use the AT&T platform services, the AT&T Adapters help developers quickly bring robust hybrid mobile applications to market.

The AT&T Adapters facilitate access to the following AT&T platform APIs:

- SMS
- SPEECH
- NOTARY
- PAYMENT
- OAUTH
- Device Capability

Note: To learn more about the AT&T API platform, visit the AT&T Developer Program website at the following location: http://developer.att.com.

2. Architectural Overview

The following illusrtration shows the relationship between the mobile device where the app is running, the Worklight server, and the AT&T API platform.

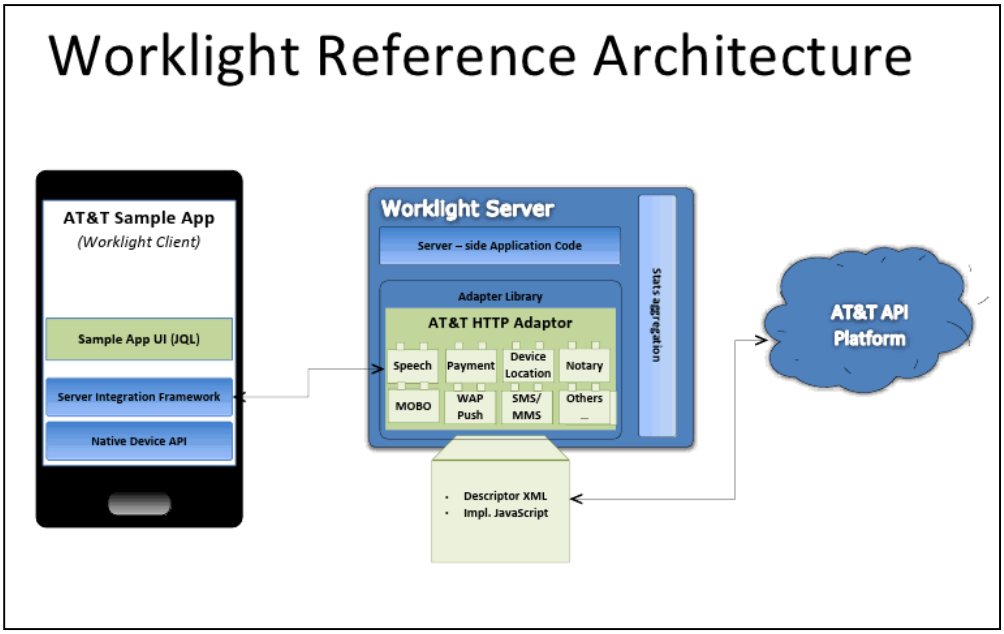

Figure 1: AT&T Adapters for Worklight - Architecture

3. Installing the Components

This section contains instructions for installing.

3.1 Pre-requisites

Perform these actions in the following order.

1. Download the AT&T Worklight project code from the following locatrion:

   https://github.com/attdevsupport/ATT_APIPlatform_Worklight

2. Download Eclipse and Worklight studio by following the instructions mentioned at http://www.ibm.com/developerworks/mobile/worklight/download/

3. Modify the eclipse.ini file as mentioned in : http://pic.dhe.ibm.com/infocenter/wrklight/v5r0m5/index.jsp?topic=%2Fcom.ibm.worklight.help.doc%2Fdevenv%2Ft_next_steps.html

4. Install Android SDK or point Eclipse to an existing installation. The Eclipse install will prompt for this.

5. Install Android ADT plugin for Eclipse per: http://developer.android.com/tools/sdk/eclipse-adt.html

6. Install XCode which would be required for iOS development. Worklight launches iPhone/iPad environments in XCode.

3.2 Import Projects

To import projects, use the following procedure:

1. Expand the ATTWLAdapterProject.zip and ATTWLApplicationProject.zip files and Import the ATTWLAdapterProject and ATTWLApplicationProject (sample app) projects into Eclipse.

2. From Eclipse, click File/Import. Expand the General folder, choose ”Existing Projects Into Workspace”, and then click Next:

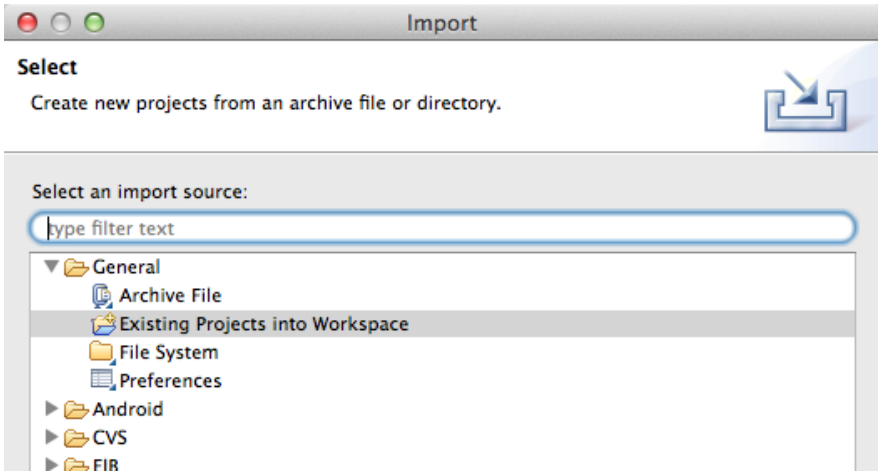

Figure 2: "Existing Projects and Workspace" location.

3. Choose "Select root directory", and click Browse.

4. Select the ATTWLAdapterProject and ATTWLApplicationProject folders, click Open, then click Finish.

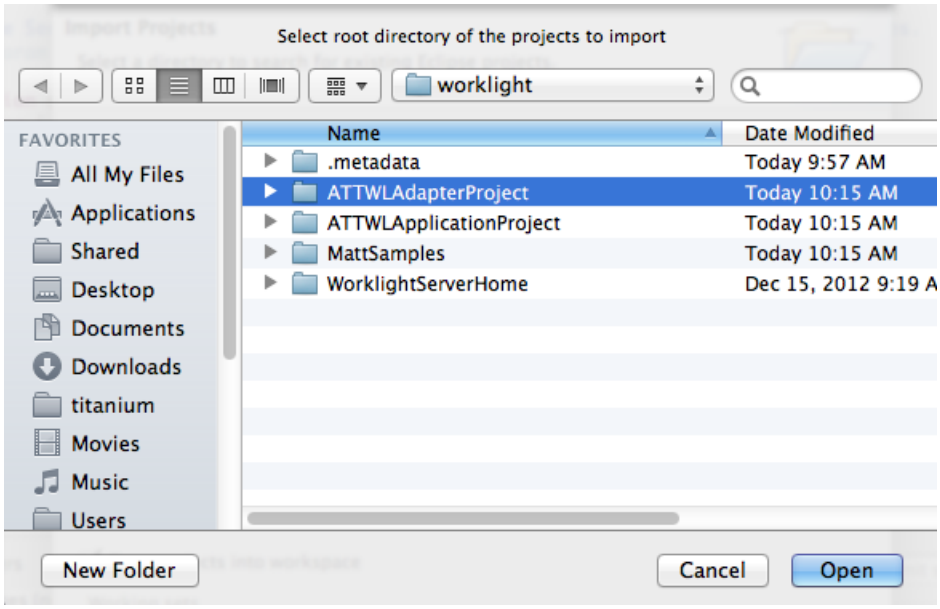

Figure 3: ATTWLAdapterProject and ATTWLApplicationProject folder location.

3.3 Configuration changes

Configuration consists of registering your application on AT&T Developer Portal with the proper services and endpoints, depending on the type of client-side application.

To register an application use the following procedure.

1. Go to https://devconnect-api.att.com and login with your valid username and password.
2. Select "My Apps" from the bar at the top of the page, and then click "Setup a New Application".

After your application is registered, you will receive an API key and Secret key. These keys are necessary to get your applications working with the AT&T Platform APIs.

To configure your Worklight application, use the following procedure:

1. Open server/conf/worklight.properties
2. Populate the following parameters in worklight.properties as specified below:-
    - apiKey: This parameter is required. Specify the API key that you received when you registered your aspplication .
    - secretKey: This parameter is required, Specify the Secret key that you received when you registered your application.
    - scope : Specify scopes for the ATT API to be invoked by your application.

3.4 Building & Deploying Adapter

To build an AT&T Worklight adapter, use the following procedure.

1. Expand the "*ATTWLAdapterProject"* in eclipse package explorer.
2. Expand the "adapters" folder to see the available adapters.

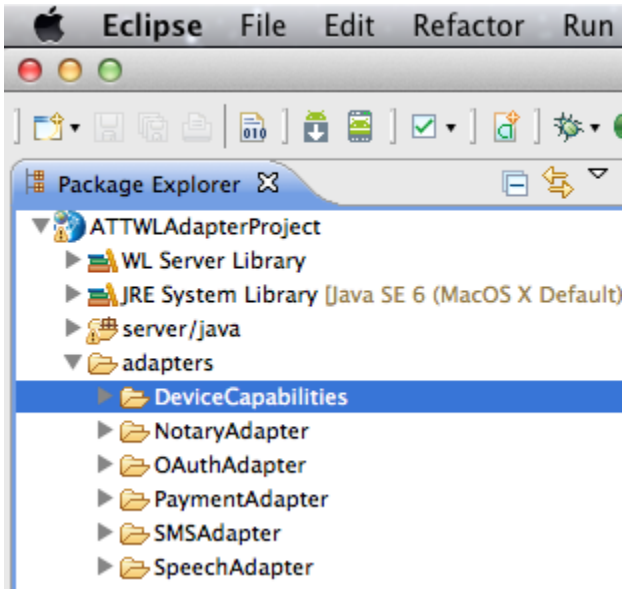

Figure 4: Available adapters.

3. Right click on an adapter folder and click “Run As -> Deploy Worklight Adapter”. Repeat this for each adapter folder. This will build the “.adapter” file in the /bin folder.

   NOTE: Deployment will happen only for this Worklight adapter project. If your sample is in a separate Worklight project, the adapters need to be redeployed via the console. This is described in a later section.

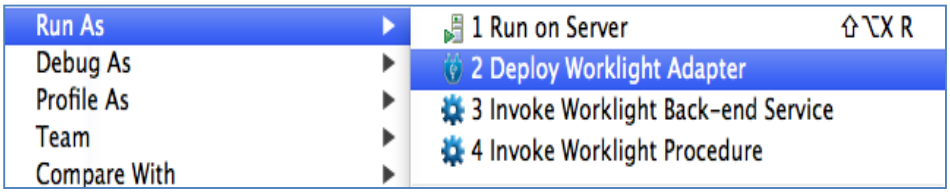

Figure 5: Building a Worklight Adapter.

3.5 Start Worklight Server

To start the Worklight server use the following procedure.

1. Navigate to the /apps/ATTWLKitchenSink/common/js/sms.js file.
2. Specify your own shortcode in the ‘shortCode’ variable that is declared in the first line of the file.

3. Right click on the ATTWLApplicationProject project and select “Start Worklight Server”.

Figure 6: Starting the Worklight server

3.6 Deploying Adapters for The Sample Application

To deploy adapters for the sample application use the following procedure.

1. Open the Worklight Console at the following loaction:

   http://<server_address>:<port>/console.

   Note: For the Worklight developer edition, by default the URL would be:

   http://localhost:8080/console

2. The IBM Worklight Console opens in a “Catalog” page that enables you to work with Applications and Adapters. Click “Choose File” and select the path of the adapters (.adapter files are generated in the bin folder of the ATTWLAdapterProject).

Figure 7: The IBM Worklight Catalog.

3. As soon as “Submit” is clicked, a message is displayed that indicates whether the deployment action succeeded or failed.

Figure 8: IBM Worklight adapter deployment success or failure message.

4. The details of the deployed adapter are added to the catalog. Clicking on “Show Details” shows the connectivity details for the adapter and the list of procedures it exposes.

Figure 9: Details of deployed adapter.

5. Repeat this procedure for each adapter.

3.7 Build and Deploy Sample App

To build and deploy a sample app used the following procedure.

1. Inside the application project folder, expand the “apps” folder and select “ATTWLKitchenSink”.

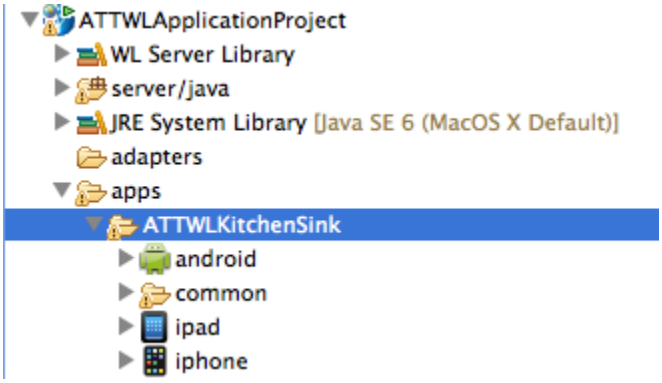

Figure 10: Select “ATTWLKitchenSink”.

2. Right click on the application folder and select “Build all and deploy”.

Figure 11: Building and deploying applications.

3. The details of the deployed application are added to the catalog and can be accessed using the URL: http://localhost:8080/console

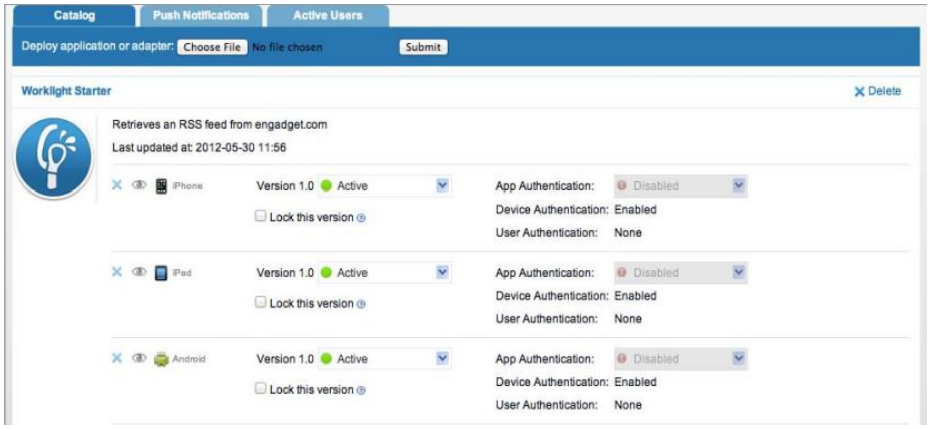

Figure 12: Details of the deployed application.

4. For the android environment, a new android project is created automatically after the application is deployed. The name of the android project created is:

   <project-name><application-name>Android

   For the sample application, the name of the project is:

   ATTWLApplicationProjectATTWLKitchenSinkAndroid

   This project can be executed as a normal android application. For example, RunAs/Android Application.

5. For the iPhone or iPad environment, right click on the “iphone” or “ipad” folder, right click and select “Run As -> Xcode Project”. This deploys your project to the Xcode where it can be executed.

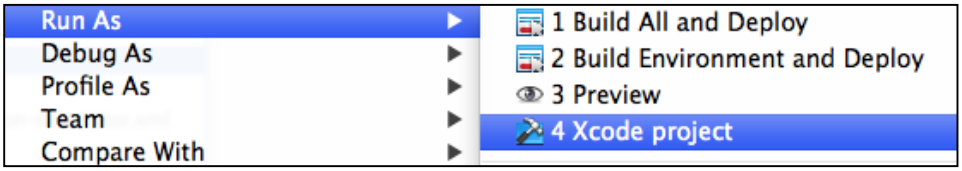

Figure 13: iPhone or iPad environment.

3.8 Using the Adapters

Worklight applications can invoke adapter procedures to communicate with any data source without being subjected to same origin constraints.

1. Invoke an adapter procedure to create an invocationData object.

```
var invocationData = {

   adapter : 'SMSAdapter',

   procedure : 'sendSMS',

   parameters : [ {'body' : {"Message" : "Hello All",
"Address" : "555-555-1212"}, 'contentType' :
'application/json', 'accept' : 'application/json',
'accessToken':'Bearer ' +  window.localStorage.accessToken} ]

};
```

This object consists of a JSON block of properties:

- **adapter** – A string that contains the name of the adapter as specified in the <wl:adapter> element.
- **procedure** – Procedure name as defined in the xml file.
- **parameters** – An array of parameters passed on to the procedure.

2. Define the failure and success behavior in an options object.

```
var options = {
     onSuccess : smsSuccess,
     onFailure : smsFailure,
     invocationContext: {}
    };
```

This object consists of a JSON block of properties:

- **onSuccess** - The function to be invoked on successful completion of the asynchronous call.
- **onFailure** – The function to be invoked on failure.
- **invocationContext** – Optional parameter. An object that is returned to the success and failure handlers.

3. Invoke the procedure using the invocationData object and options object.

```
WL.Client.invoke.Procedure(invocationData, options)
```

Note: To learn more about installing Worklight adapters, refer to the following guide: http://public.dhe.ibm.com/ibmdl/export/pub/software/mobile-solutions/worklight/docs/wl_5_0_admin.pdf