AT&T API Platform Adapter for IBM® Worklight®

Installation and Setup Guide

Publication Date: August 28, 2013

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

Contents

Table of Figures

Table of Tables

Table of Examples

1 Introduction

The AT&T API Platform Adapter for IBM® Worklight® provides a simplified way for Worklight developers to access the AT&T API Platform and RESTful APIs. By significantly reducing the complexity of building applications that use the AT&T API Platform, the adapter helps developers quickly bring robust hybrid mobile applications to market.

The adapter facilitates access to the following AT&T Platform RESTful APIs:

- Advertisements
- Device Capability
- Notary
- OAuth
- Payment
- SMS
- Speech to Text
- Text to Speech

Note: To learn more about the AT&T API Platform, see the AT&T Developer Program website.

2 Architectural Overview

Figure 1 shows the relationship between the mobile device where the app is running, the Worklight server, and the AT&T API platform.

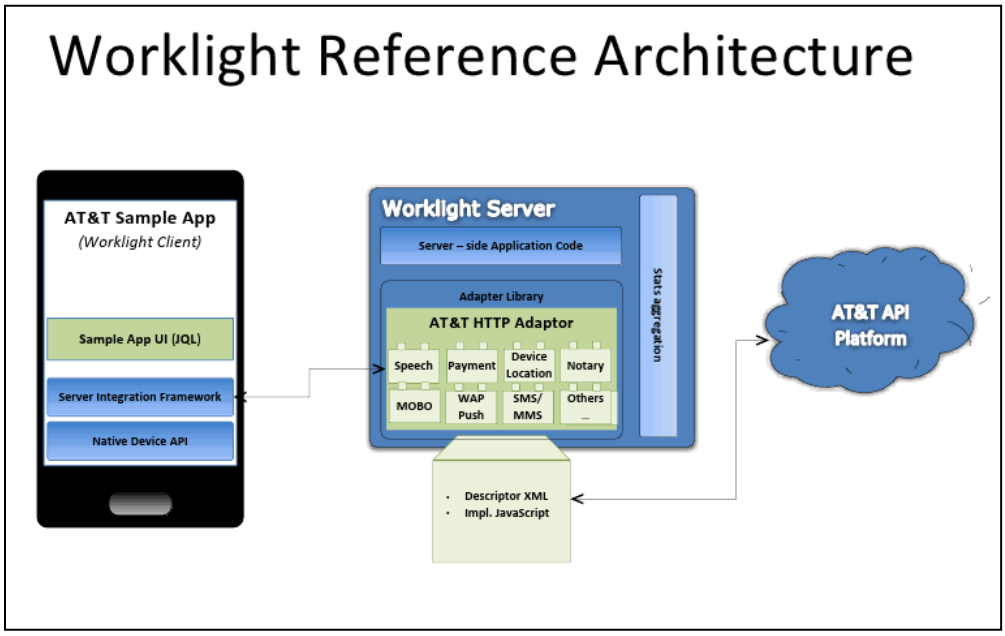

Figure 1: AT&T Worklight Architecture

3 Installing the Worklight Adapters Components

You must complete the following steps to install the required components for creating applications (apps) with the adapter.

- Download Worklight Studio
- Download the AT&T API Platform Adapters for IBM® Worklight®
- Build and deploy the adapters
- Use the adapters in your mobile app
- Deploy your app to the Worklight server

3.1 Installing the Tools

Perform the following steps to install the required tools for creating an adapter app.

1. Download the AT&T Worklight project code from GitHub.
2. Download Eclipse and Worklight studio by following the instructions on the IBM developer website.
3. Modify the eclipse.ini file by following the instructions on the IBM Worklight Information Center.
4. Install the Android SDK, or if is already installed point Eclipse to the existing location.
5. Install the Android Development Tools (ADT) plugin for Eclipse by following the instructions on the Android developer website.
6. If you are doing iOS development, install XCode by following the instructions on the Apple developer website. Worklight launches iPhone/iPad environments in XCode.

3.1.1 Importing Projects

Perform the following steps to import projects.

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.
3. Expand the General folder, choose Existing Projects Into Workspace as shown in Figure 2
4. Click Next.

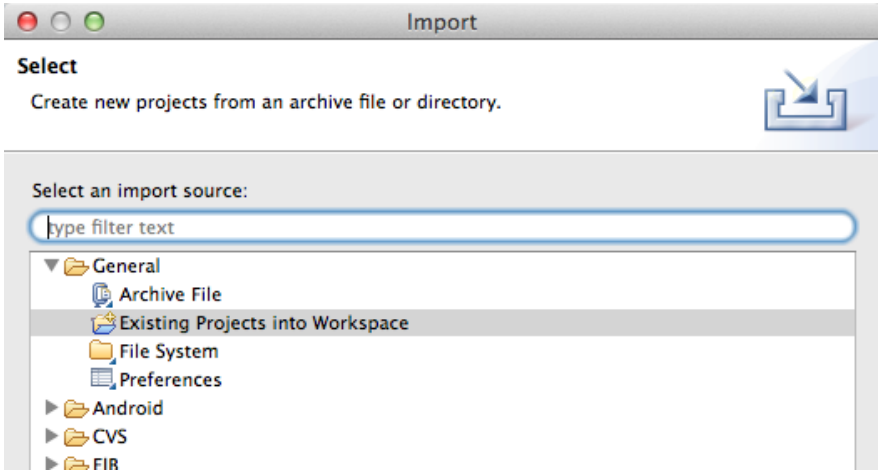

Figure 2: "Existing Projects and Workspace" location

5. Choose Select root directory, and click Browse.
6. Select the ATTWLAdapterProject folder and the ATTWLApplicationProject folder as shown in Figure 3.
7. Click Open.
8. Click Finish.

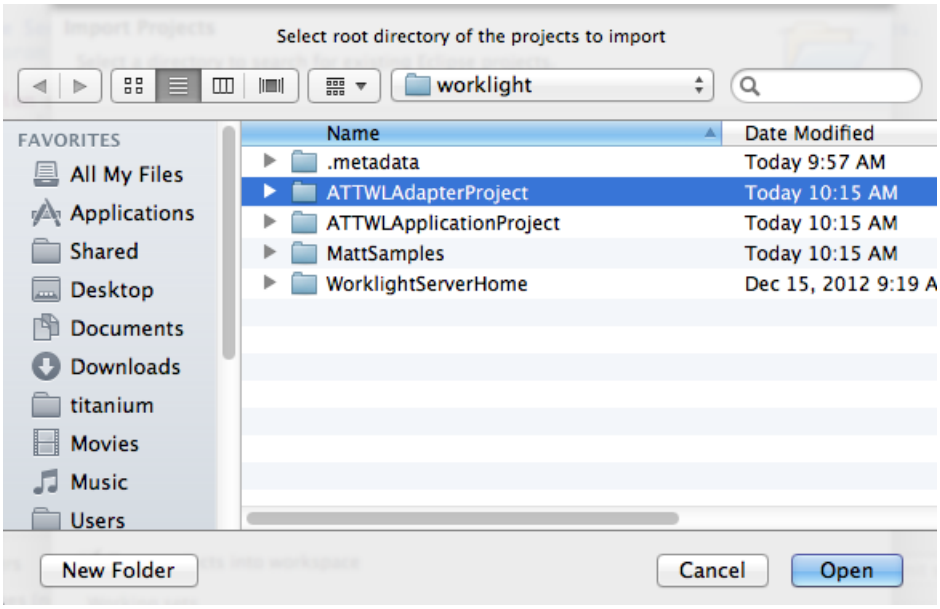

Figure 3: Adapter Project and Application Project folder locations

3.1.2 Registering your Application

Before you can access the AT&T Platform services, you must register your application for the services that you want to access and to get an API key and secret key. These keys are necessary to call the underlying AT&T RESTful APIs that access the services.

Perform the following steps to register an application.

1. Create a new developer account on the AT&T Developer website, or login to an existing account.
2. Select My Apps from the bar at the top of the page
3. Click Setup a New Application.

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

3.1.3 Configuring your WorkLight Application

Perform the following steps to configure your Worklight application.

1. Open server/conf/worklight.properties.
2. Add the property values from the following table. All values are case-sensitive and should be separated by commas.

Property	Value	Description
apiKey	myApiKey	API key that you received when you registered your application.
secretKey	mySecretKey	Secret key that you received when you registered your application.
scope	myScope	Reference to the ATT service that is invoked by your application, for example, • Advertising is ADS • Device Capabilities is DC • Notary is NOTARY • Payment service is PAYMENT • SMS is SMS • Speech To Text is SPEECH • Text To Speech is TTS

Table 1: Configuration properties for Worklight

4 Building and Deploying a Worklight Adapter

Perform the following steps to build an AT&T Worklight adapter.

1. Expand the ATTWLAdapterProject in the Eclipse package explorer.
2. Expand the adapters folder to see the available adapters, as shown in the following figure.

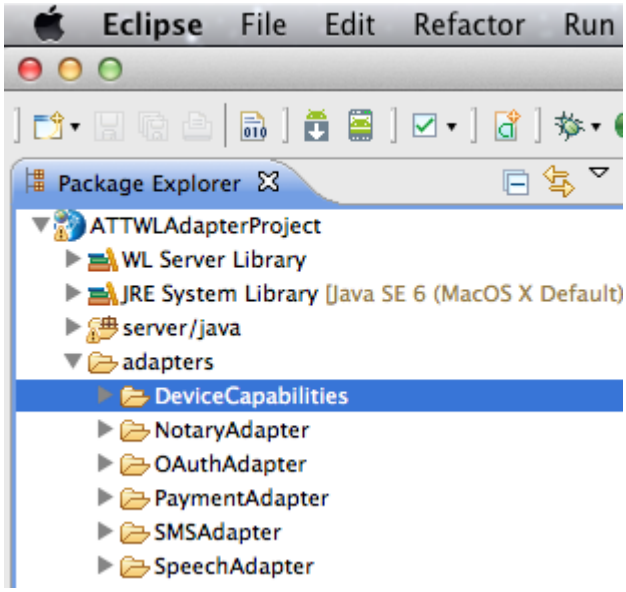

Figure 4: Available adapters.

3. Right click on an adapter folder and click Run As -> Deploy Worklight Adapter, as shown in the following figure. Repeat this for each adapter folder. This builds the .adapter file in the /bin folder.

   These steps deploy the Worklight adapter project. If your sample is in a separate Worklight project, the adapters need to be redeployed using the console.

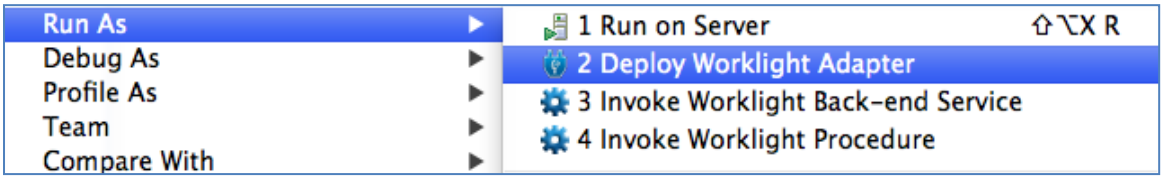

Figure 5: Building a Worklight Adapter.

5 Starting the Worklight Server

Perform the following steps to start the Worklight server.

1. Navigate to the /apps/ATTWLKitchenSink/common/js/sms.js file.
2. Specify your own short code in the shortCode variable that is declared at the top of the file.
3. Right click on the ATTWLApplicationProject project and select Start Worklight Server, as shown in the following figure.

Figure 6: Starting the Worklight server.

6 Deploying Adapters for the Sample Application

Perform the following steps to deploy adapters for the sample application.

1. When you are using the IBM Worklight Developer Studio, open the Worklight Console at the following location:

   http://localhost:8080/console

   The IBM Worklight Console opens in a Catalog page that lets you work with Applications and Adapters, as shown in the following figure.

Figure 7: IBM Worklight Catalog Page.

2. Click Choose File and select an adapter from the bin folder of the ATTWLAdapterProject.
3. Click Submit.

   Worklight displays a message that indicates whether the deployment action succeeded or failed. Repeat this procedure for each adapter.

   The deployed adapter is added to the catalog. Refresh the page to see the details, as shown in the following figure.

Figure 8: Details of deployed adapter.

7 Building and Deploying the Sample Application

Perform the following steps to build and deploy the sample application.

1. Open the ATTWLApplicationProject folder in Project Explorer.
2. Expand the apps folder.
3. Select ATTWLKitchenSink.

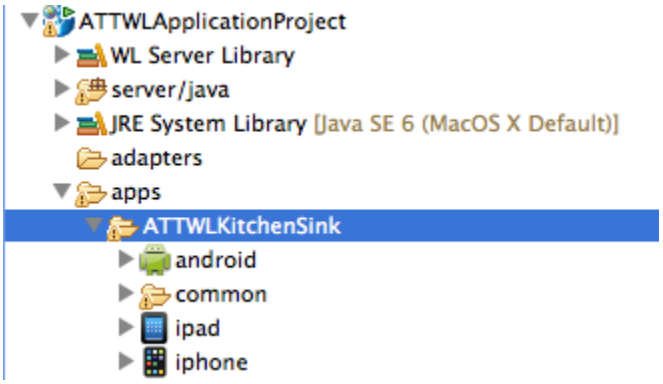

Figure 9: Selecting the ATTWLKitchenSink app.

4. Right click on ATTWLKitchenSink and select Build all and deploy, as shown in the following figure.

Figure 10: Building and deploying applications.

5. The details of the deployed application are added to the catalog and can be accessed at http://localhost:8080/console, as shown in the following figure.

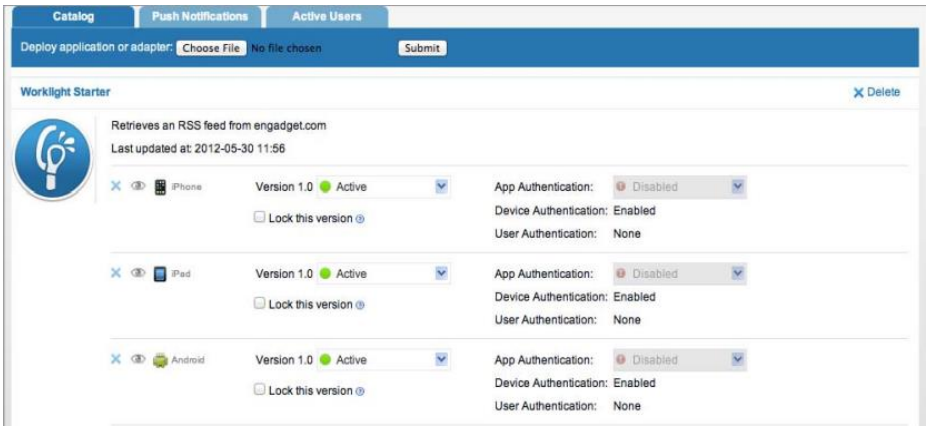

Figure 11: Details of the deployed application.

7.1 Android Deployment

For the android environment, the following new android project is created automatically after the application is deployed:

`project-nameapplication-nameAndroid`

Project-name is the name of the project and application-name is the name of the app. 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`

7.2 IPhone Deployment

For the iPhone or iPad environment, right click on the iphone or ipad folder, and select Run As -> Xcode project, as shown in the following figure. This deploys your project to the Xcode environment where it can be executed.

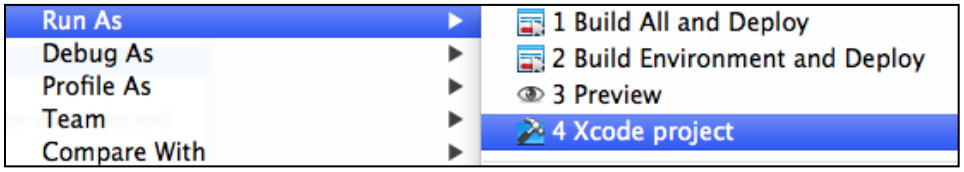

Figure 12: Deploying the project in the iPhone or iPad environment.

8 Using the Adapters to Create a new Worklight Application

To create a new Worklight application, see Working with Worklight.

1. In your application, invoke the ATT adapters from Java Script. Worklight applications can invoke adapter procedures to communicate with any data source without being subjected to same origin constraints.
2. Invoke an adapter procedure to create an invocationData object.

invocationData Object Code Sample

```
var invocationData = {
      adapter : 'SMSAdapter',
      procedure : 'sendSMS',
      parameters : [{'body' : { "outboundSMSMessage": {"Message" :
"Hello All",  "Address" : "555-555-1212"}},
'contentType' : 'application/json', 'accept' : 'application/json',
'accessToken':'Bearer ' +  window.localStorage.accessToken}]
    };
```

Example 1: Creating an invocationData object

The invocationData object consists of the following JSON block of properties:

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

Table 2: invocationData Object Properties

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

Options Object Code Sample

```
var options = {
      onSuccess : yourSuccessCallback,
      onFailure : yourFailureCallback,
      InvocationContext {}
    };
```

Example 2: Creating an Options object

The options object consists of the following JSON block of properties:

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

Table 3: Options Object Properties

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

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

To learn more about installing Worklight adapters, see the IBM Worklight Administration Guide.