Tag-based notifications in native Windows 8 applications

Overview

Tag notifications are notification messages that are targeted to all the devices that are subscribed to a particular tag. Tags represent topics of interest to the user and provide the ability to receive notifications according to the chosen interest.

Setting up tags

Tags are defined in the `application-descriptor.xml` file:

```
<application xmlns="http://www.worklight.com/application-descriptor" id="HybridTagNotifications" platformVersio
n="7.0.0.00.20150312-0731">
...
...
...
<tags>
  <tag>
    <name>my tag 1</name>
    <description>About my tag 1</description>
  </tag>
  <tag>
    <name>my tag 2</name>
    <description>About my tag 2</description>
  </tag>
</tags>
```

API methods for tag-based notification

Client-side API methods:

- `WL.Client.Push.subscribeTag(tagName,options)` - Subscribes the device to the specified tag name.
- `WL.Client.Push.unsubscribeTag(tagName,options)` - Unsubscribes the device from the specified tag name.
- `WL.Client.Push.isPushSupported()` - Returns `true` if push notifications are supported by the platform, or `false` otherwise.
- `WL.Client.Push.isTagSubscribed(tagName)` - Returns whether the device is subscribed to a specified tag name.

Common API methods for tag-based and broadcast notifications

Client-side API

- `WLNotificationListener` Defines the callback method to be notified when the notification arrives.
- `client.getPush().setWLNotificationListener(listener)` This method sets the implementation class of the `WLNotificationListener` interface.
- `client.getPush().setOnReadyToSubscribeListener(listener)` This method registers a listener to be used for push notifications. This listener must implement the `onReadyToSubscribe()` method.
- The `onMessage(props,payload)` method of `WLNotificationListener` is called when a push notification is received by the device.

- **props** - A JSON block that contains the notification properties of the platform.
- **payload** - A JSON block that contains other data that is sent from MobileFirst Server. The JSON block also contains the tag name for tag-based or broadcast notification. The tag name appears in the "tag" element. For broadcast notification, the default tag name is `Push.ALL`.

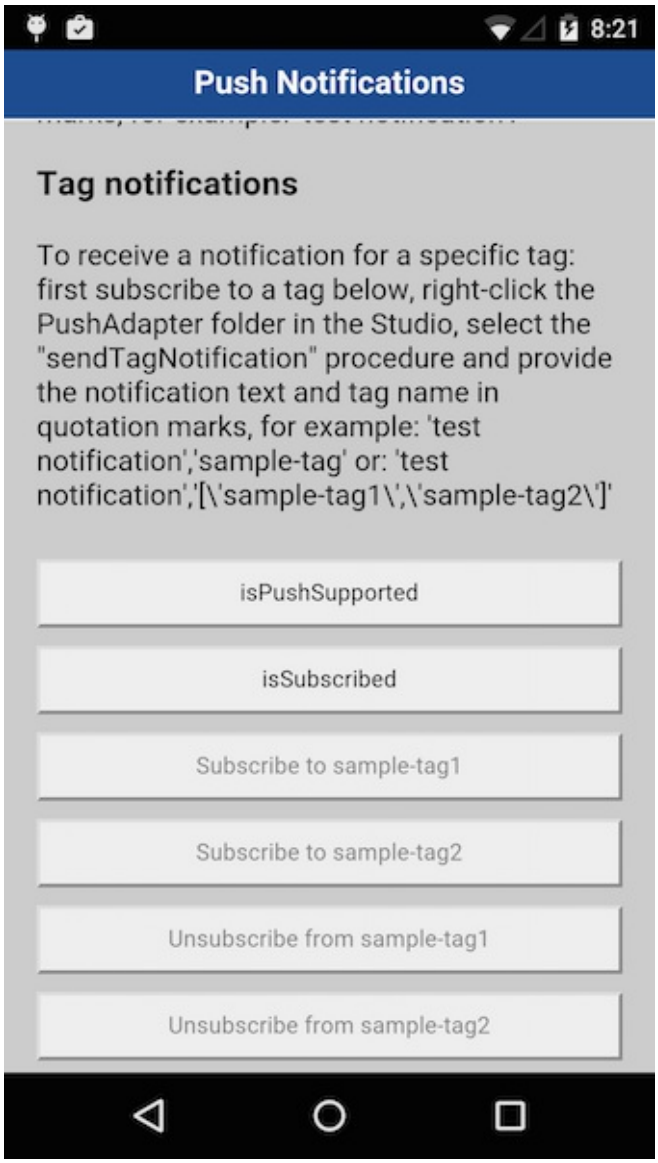
Server-side API

`WL.Server.sendMessage(applicationId,notificationOptions)` **applicationId** - (mandatory) The name of the MobileFirst application. **notificationOptions** - (mandatory) A JSON block containing message properties. Submits a notification based on the specified target parameters.

> For more information about tag-based notification, see the "tag-based notification" topic in the user documentation.

Sample application

Click to download (http://public.dhe.ibm.com/software/products/en/MobileFirstPlatform/docs/v700/TagBasedPushNotificationsProject.zip) the Studio project. The provided sample application demonstrates how to send a broadcast notification to the default autosubscribed `push.ALL` tag. The sample also demonstrates how to send notifications to two predefined tags, sample-tag1 and sample-tag2, to which the user can subscribe in the application.



8:22
Push Notifications
Broadcast notification
To receive a broadcast notification: right-click the PushAdapter folder in the Studio, select the "sendBroadcastNotification" procedure and provide the notification text in quotation marks, for example: 'test notification'.
Tag notifications
To receive a notification for a specific tag: first subscribe to a tag below, right-click the PushAdapter folder in the Studio, select the "sendTagNotification" procedure and provide the notification text and tag name in quotation marks, for example: 'test notification','sample-tag' or: 'test notification','[\'sample-tag1\',\'sample-tag2\']'
isPushSupported
isSubscribed
Subscribe to sample-tag1