



Learn IT your way
All for about \$1/day! [Learn more](#)

Development

✓ Microsoft, Mobile
✓ Java, Cloud, UX

IT Pro

✓ Windows Server
✓ ITIL, PMP, CompTIA

[home](#) [articles](#) [quick answers](#) [discussions](#) [features](#) [community](#)
[help](#)

Search for articles, questions, tips



Articles » Mobile Development » Android » General

Next →

Article

Browse Code

Stats

Revisions (2)

Alternatives

Comments &
Discussions (6)

WCF and Android: Part II

By **deveck**, 5 Apr 2012

★ ★ ★ ★ ★ 4.50 (4 votes)

Sign Up to vote

Tweet 7

G+1 0

Like 2

Introduction

The first part of this series [WCF and Android Part I](#) describes how to setup a JSON REST WCF service. This post shows how to consume this service using Android.

Simple Webservice invocation

Android already comes with a ready to use HTTP-client located in the package **org.apache.http.client**.

[Collapse](#) | [Copy Code](#)

```
HttpParams httpParams = new BasicHttpParams();
ThreadSafeClientConnManager connMgr = new ThreadSafeClientConnManager(httpParams, new
SchemeRegistry());
HttpClient client = new DefaultHttpClient(connMgr, httpParams);
```

Therefore a simple webservice invocation looks as follows:

[Collapse](#) | [Copy Code](#)

```
HttpGet get = new HttpGet("http://requesturi/method");
HttpResponse response = client.execute(get);
HttpEntity entity = response.getEntity();
```

JSON serialization

Simple parameters can be passed in the URL but more complex objects are likely passed in the http body using the POST method. For modern and fast webservices it is common to use **JSON** for data serialization. Android has some JSON features built in but [Google GSON](#) provides a very flexible way to serialize and deserialize JSON objects, even for arbitrary object types. Download the library and include it in your build-path. The first step is to create a **GsonBuilder**, where some de-/serialization options can be specified.

[Collapse](#) | [Copy Code](#)

```
GsonBuilder gson = new GsonBuilder()
    .registerTypeAdapter(Date.class, new DateSerializer())
    .serializeNulls()
    .setDateFormat(DateFormat.LONG)
    .setFieldNamingPolicy(FieldNamingPolicy.UPPER_CAMEL_CASE)
    .setPrettyPrinting()
    .create();
```

Beside some others this example attaches a custom DateSerializer (thanks to [Benjii Me](#)) to be able to read WCF serialized DateTime values. That's all you need to do. Serialization now looks as follows:

[Collapse](#) | [Copy Code](#)

```
String jsonString = gson.toJson(myObject);
```

and deserialization very similar:

[Collapse](#) | [Copy Code](#)

```
MyObjectType myObj = gson.fromJson(jsonString, MyObjectType.class);
```

About Article

A **Technical Blog**, originally posted at <http://www.deveck.net/deveck-and-android-part-ii>

How to consume a JSON REST WCF service using Android.

Type **Technical Blog**

Licence **Apache**

First Posted **5 Apr 2012**

Views **24,946**

Bookmarked **14 times**

C# Dev WCF Intermediate
Android



SpreadsheetGear
Performance Spreadsheet Components

NEW! WPF Controls

NEW! Silverlight Controls

NEW! Multithreaded Calculation Engine

f(x)
NEW! 64 Additional Excel Functions

SpreadsheetGear
Download Free Trial

Top News

[Man throws away trove of](#)

Webservice invocation with JSON-POST arguments

Every webservice requests only allows a single JSON object in the HTTP-body, therefore if multiple arguments are required they need to be wrapped in a single JSON object and the WCF service contract needs to be aware of it (BodyStyle=WebMessageBodyStyle.WrappedRequest).

Once the webservice is aware of the BodyStyle the client side is quite straight forward. Create a Map from name to object, add all objects with the name of the webservice parameter and serialize the whole map:

```
HashMap<String, Object> postParameters = new HashMap<String, Object>();
postParameters.put("param1", objParam1);
postParameters.put("param2", objParam2);
postParameters.put("param3", objParam3);
gson.toJson(postParameters);
```

[Collapse](#) | [Copy Code](#)

A complete webservice call looks as follows:

```
HttpPost post = new HttpPost(requestUri);
post.setHeader("Accept", "application/json");
post.setHeader("Content-type", "application/json");

String jsonParameters = gson.toJson(postParameters);
post.setEntity(new StringEntity(jsonParameters));

HttpResponse response = client.execute(post);
HttpEntity entity = response.getEntity();
```

[Collapse](#) | [Copy Code](#)

Reading the webservice response

Assuming that the webservice was called successful a **HttpEntity** is returned. This entity can be easily converted to the corresponding object as shown above. The only difficulty is to convert the **HttpEntity** to a JSON string:

```
gson.fromJson(convertStreamToString(entity.getContent()), MyObjectType.class);

public static String convertStreamToString(InputStream is) {
    BufferedReader reader = new BufferedReader(new InputStreamReader(is));
    StringBuilder sb = new StringBuilder();

    String line = null;
    try {
        while ((line = reader.readLine()) != null) {
            sb.append(line + "\n");
        }
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        try {
            is.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    return sb.toString();
}
```

[Collapse](#) | [Copy Code](#)

Original post blogged on [b2evolution](#).


License

This article, along with any associated source code and files, is licensed under [The Apache License, Version 2.0](#)

About the Author



deveck

Student
Austria 

Bitcoin worth \$7.5 million

Get the Insider News free each morning.

Related Videos



Related Articles

[A Beginner's Tutorial for Understanding Windows Communication Foundation \(WCF\)](#)

[WCF does not support Compression out of the box, so fix it](#)

[Android - Stock Market Watch \(COINS\) in C# using Visual Studio 2010](#)

[What's the Difference between WCF and Web Services?](#)

[Windows Communication Foundation FAQ quick starter: Part 1](#)

[A Beginner's Tutorial on How to Host a WCF Service \(IIS Hosting and Self Hosting\)](#)

[4 Simple Steps to Consume WCF Service using Silverlight](#)

[Create RESTful WCF Service API: Step By Step Guide](#)

[Eight steps to enable Windows authentication on WCF BasicHttpBinding](#)

[Understanding SynchronizationContext: Part III](#)

[A Windows Communication Foundation \(WCF\) Overview](#)

[MonoAndroid: Calling secondary activity](#)

[Basic Authentication on a WCF REST Service](#)

[WCF: From a Beginner's perspective & a Tutorial](#)

[Windows Communication Foundation QuickStart - Multiple Binding VS2010](#)

[MonoAndroid: Using AlertDialog](#)

[A Silverlight Sample Built with Self-Tracking Entities and WCF Services - Part 1](#)

[Home Automation with Netduino and Kinect](#)

[WCF Service Method Level Security using Message Contract](#)

[Build C# and .NET Mobile Apps with Xamarin Development for Android - Part I](#)

Related Research

No Biography provided

[Article Top](#)



Learn Agile: Ten Tips for Launching and Testing High Quality Apps for the American Market



The Essential Guide to Mobile App Testing: Tips for Developers in USA & Canada

Comments and Discussions

You must [Sign In](#) to use this message board.

Search this forum

Go

☒ Profile popups

Spacing

Relaxed

Noise

Very High

Layout

Normal

Per page

10

Update

First

Prev

Next

Proper link to Part I	suhani26	6-Nov-13 20:03
Two more pieces: (a) Encrypt .NET, decrypt Android, (b) XmlDocument parsing [modified]	devvvy	25-Jan-13 16:15
My vote of 4	pedibär	5-Nov-12 21:10
Link to Part I broken	milter000	18-Sep-12 19:55
Re: Link to Part I broken	Corke Ecratius	12-Nov-12 9:43
Hope for code	ttssrs	5-Sep-12 23:57

Last Visit: 31-Dec-99 18:00

Last Update: 3-Dec-13 6:45

Refresh

1

[General](#) [News](#) [Suggestion](#) [Question](#) [Bug](#) [Answer](#) [Joke](#) [Rant](#) [Admin](#)

Use Ctrl+Left/Right to switch messages, Ctrl+Up/Down to switch threads, Ctrl+Shift+Left/Right to switch pages.