



# Python on Android

Sajjad Anwar  
@geohacker





```
package com.example.helloandroid;

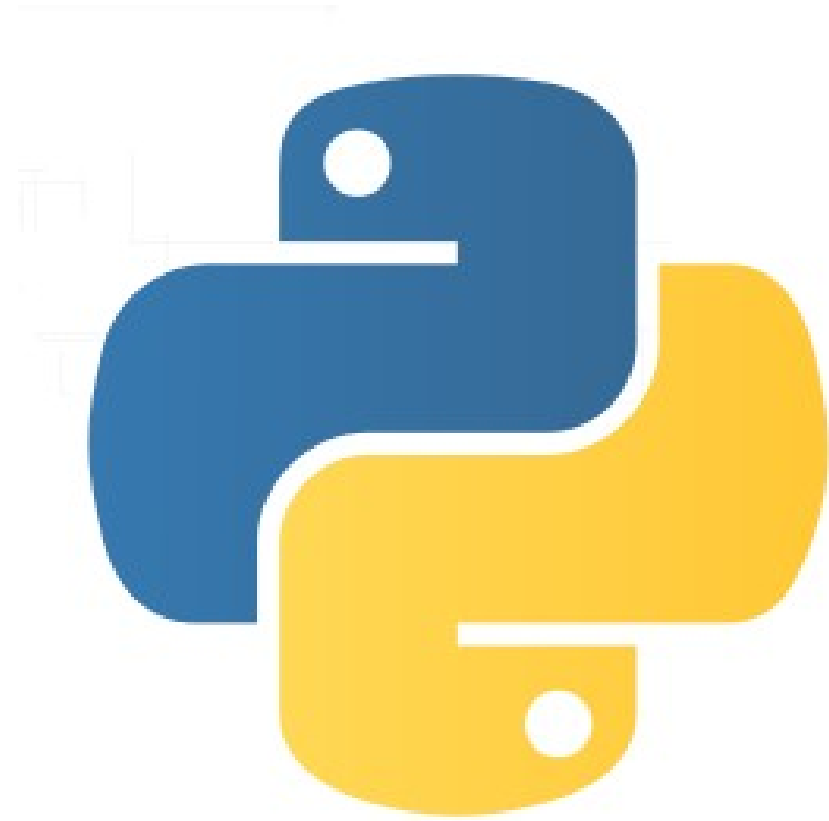
import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HelloAndroid extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("Hello, Android");
        setContentView(tv);
    }
}
```

```
import android
droid = android.Android()
droid.makeToast("Hello, Android!")
```



15 LOC



3 LOC



# Scripting Layer for Android

<http://android-scripting.googlecode.com>

- Started in 2009 by Damon Kohler.
- Brings scripting languages to Android.
- Supports Python, Ruby, Perl, Lua, JavaScript, BeanShell and more.
- Access to most of the android resources.
- Quick application prototyping.



# SL4A Services

- `ActivityResult`
- `Android`
- `ApplicationManager`
- `BatteryManager`
- `Camera`
- `CommonIntents`
- `Contacts`
- `Event`
- `EyesFree`
- `Location`
- `MediaPlayer`
- `MediaRecorder`
- `Phone`
- `Preferences`
- `SensorManager`
- `Settings`
- `Sms`
- `SpeechRecognition`
- `ToneGenerator`
- `WakeLock`
- `Wifi`
- `UI`

# Examples

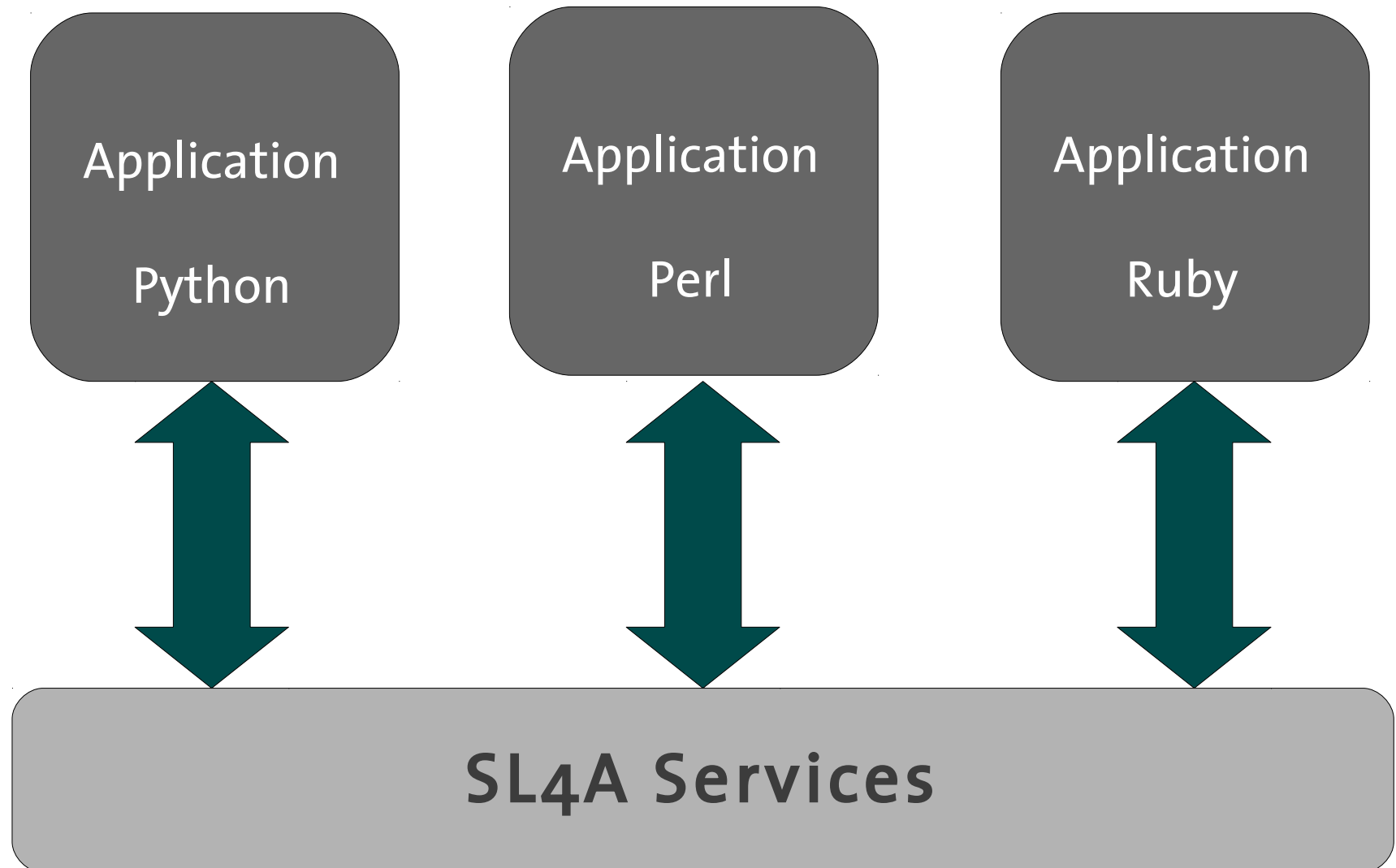
# Py4A

- Additional python modules added:
  - Bluez, Twisted, Zope, pyEphem
- Current maintainers:
  - Naranjo Manuel Francisco
  - Robbie Matthews

# This is NO magic!

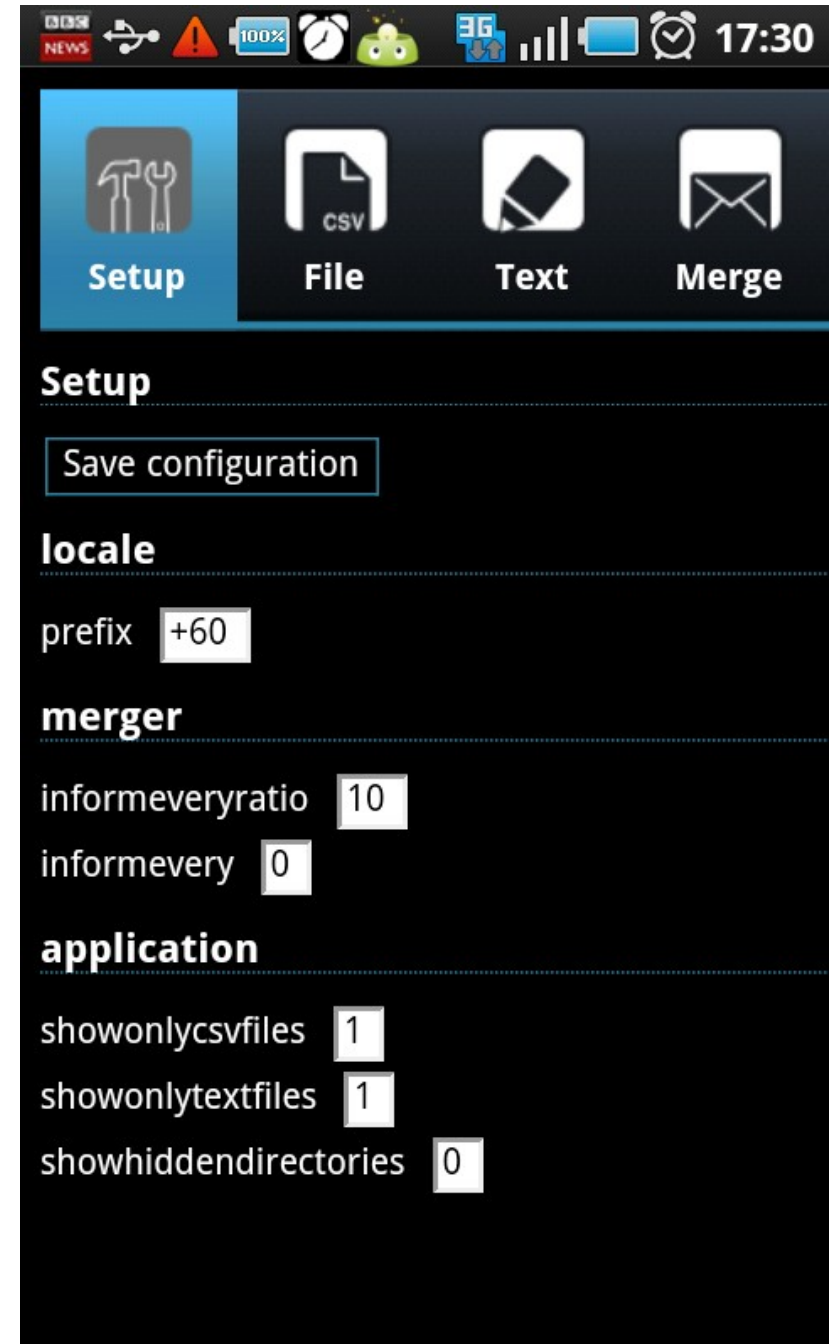
- Android functionality is abstracted into methods.
- These methods are grouped in subsystems called 'Facades'.
- A JSON-RPC server exposes the methods contained in these Facades.

# Architecture



# User Interface

- Frontend – webView  
(HTML + JS)
- Backend – Python
- Communicated through  
events.





# Yes. You can package as well!

- Download the template project archive
- 
- <http://android-scripting.googlecode.com/hg/android/>
- `script_for_android_template.zip`
- Modify the following according to your project
  - Import into Eclipse
  - 
  - Modify or put your script into the `res/raw` directory.
  -
- Refactor package name from `com.dummy.fooforandroid` to
- your package name.
- <http://code.google.com/p/android-scripting/wiki/SharingScripts>
-



Thank you!