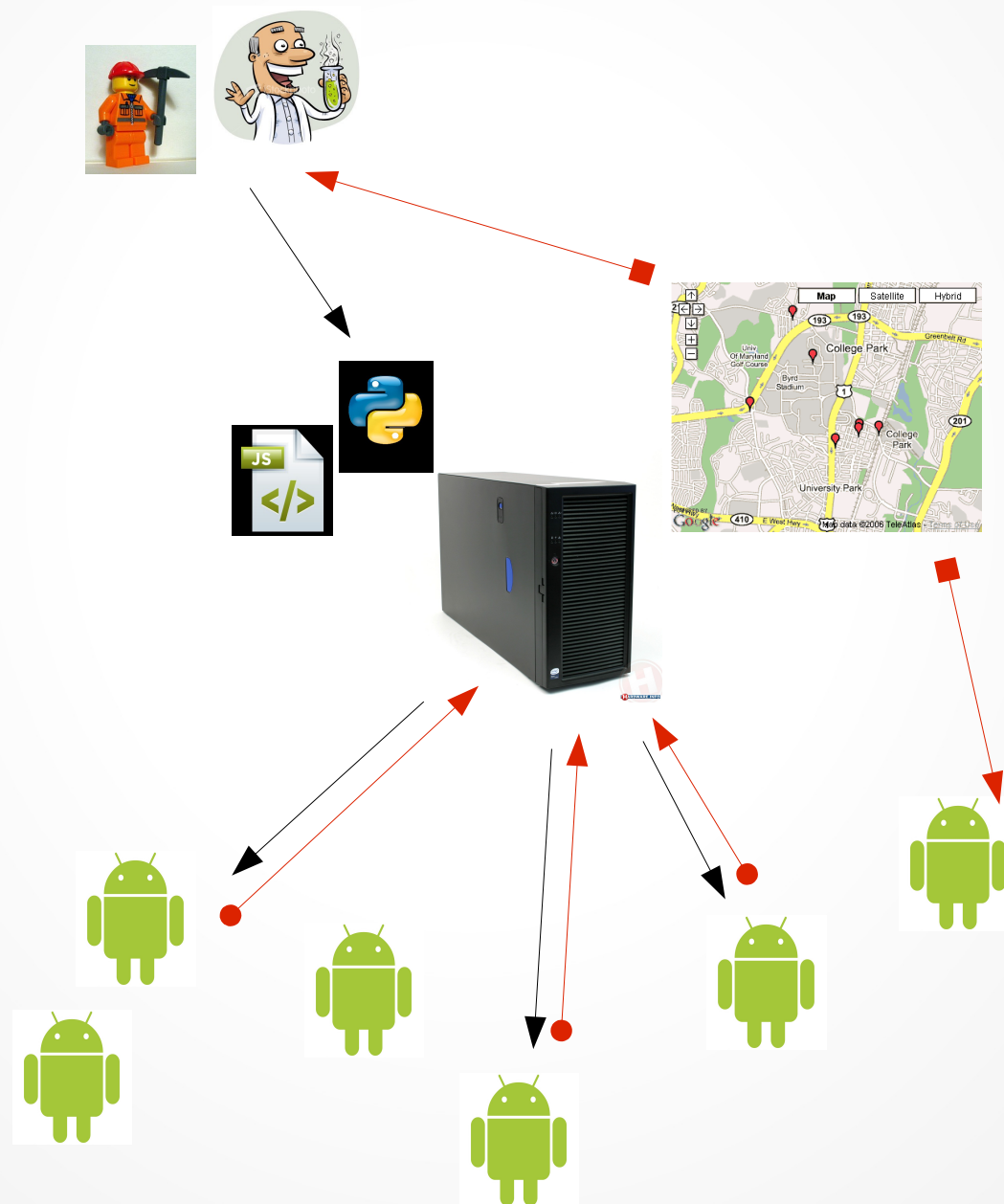


Το πρόβλημα

- υπηρεσία που παρέχει σε χρήστες δυνατότητα συλλογής μετρήσεων από τους αισθητήρες των android κινητών συσκευών
-
- χρήστες (testers) δίνουν το test script τους σε ένα κεντρικό σύστημα και αυτό αναλαμβάνει να το μοιράσει σε ένα δίκτυο από κινητές συσκευές ώστε να το τρέξουν τοπικά.
- Οι κινητές συσκευές στέλνουν τις μετρήσεις πίσω στο σύστημα ώστε να είναι διαθέσιμες στους testers αλλά και στο κοινό (ιδιοκτήτες κινητών συσκευών και κοινό)

Το πρόβλημα



Σκοπός δημιουργίας

- ένα σύνολο μετρήσεων του περιβάλλοντος στο οποίο κινούνται οι χρήστες είναι διαθέσιμο στο κοινό
- άμεσα συμπεράσματα για το περιβάλλον των χρηστών
- εξαγωγή πληροφορίας με επιπλέον επεξεργασία (offline) και συνδυασμός με άλλα δεδομένα
- δημιουργία κοινότητας από testers
- Internet of things
- ???

οι κινητές συσκευές διαθέτουν ένα σύνολο από αισθητήρες

Η πλατφόρμα του android υποστηρίζει 3 κατηγορίες αισθητήρων:

- Motion sensors

αισθητήρες μέτρησης της επιτάχυνσης και των δυνάμεων περιστροφής στους 3 άξονες. Αυτή η κατηγορία περιλαμβάνει accelerometers, gravity sensors, gyroscopes, και rotational vector sensors

- Environmental sensors

αισθητήρες μέτρησης διάφορων παραμέτρων του περιβάλλοντος, όπως την θερμοκρασία και την πίεση στην ατμόσφαιρα, το φωτισμό και την υγρασία. Αυτή η κατηγορία περιλαμβάνει barometers, photometers, και thermometers.

- Position sensors

αισθητήρες μέτρησης της φυσικής θέσης της συσκευής. Αυτή η κατηγορία περιλαμβάνει orientation sensors and magnetometers

Οι αισθητήρες που υποστηρίζει το Android

Table 1. Sensor types supported by the Android platform.

Sensor	Type	Description	Common Uses
<u>TYPE_ACCELEROMETER</u>	Hardware	Measures the acceleration force in m/s^2 that is applied to a device on all three physical axes (x, y, and z), including the force of gravity.	Motion detection (shake, tilt, etc.).
<u>TYPE_AMBIENT_TEMPERATURE</u>	Hardware	Measures the ambient room temperature in degrees Celsius ($^{\circ}\text{C}$). See note below.	Monitoring air temperatures.
<u>TYPE_GRAVITY</u>	Software or Hardware	Measures the force of gravity in m/s^2 that is applied to a device on all three physical axes (x, y, z).	Motion detection (shake, tilt, etc.).
<u>TYPE_GYROSCOPE</u>	Hardware	Measures a device's rate of rotation in rad/s around each of the three physical axes (x, y, and z).	Rotation detection (spin, turn, etc.).
<u>TYPE_LIGHT</u>	Hardware	Measures the ambient light level (illumination) in <u>lx</u> .	Controlling screen brightness.
<u>TYPE_LINEAR_ACCELERATION</u>	Software or Hardware	Measures the acceleration force in m/s^2 that is applied to a device on all three physical axes (x, y, and z), excluding the force of gravity.	Monitoring acceleration along a single axis.
<u>TYPE_MAGNETIC_FIELD</u>	Hardware	Measures the ambient geomagnetic field for all three physical axes (x, y, z) in <u>μT</u> .	Creating a compass.

Οι αισθητήρες που υποστηρίζει το Android

<u>TYPE_ORIENTATION</u>	Software	Measures degrees of rotation that a device makes around all three physical axes (x, y, z). As of API level 3 you can obtain the inclination matrix and rotation matrix for a device by using the gravity sensor and the geomagnetic field sensor in conjunction with the <u>getRotationMatrix()</u> method.	Determining device position.
<u>TYPE_PRESSURE</u>	Hardware	Measures the ambient air pressure in <u>hPa</u> or <u>mbar</u> .	Monitoring air pressure changes.
<u>TYPE_PROXIMITY</u>	Hardware	Measures the proximity of an object in cm relative to the view screen of a device. This sensor is typically used to determine whether a handset is being held up to a person's ear.	Phone position during a call.
<u>TYPE_RELATIVE_HUMIDITY</u>	Hardware	Measures the relative ambient humidity in percent (%).	Monitoring <u>dewpoint</u> , absolute, and relative humidity.
<u>TYPE_ROTATION_VECTOR</u>	Software or Hardware	Measures the orientation of a device by providing the three elements of the device's rotation vector.	Motion detection and rotation detection.
<u>TYPE_TEMPERATURE</u>	Hardware	Measures the temperature of the device in degrees Celsius (°C). This sensor implementation varies across devices and this sensor was replaced with the <u>TYPE_AMBIENT_TEMPERATURE</u> sensor in API Level 14	Monitoring temperatures.

Οι αισθητήρες που υποστηρίζει το Android

- Οι τιμές των αισθητήρων είναι διαθέσιμες μέσα από το API του android
- Μία εφαρμογή που τρέχει σε μία συσκευή μπορεί να έχει στην διαθεσή της αυτές τις μετρήσεις.

Πιθανό σενάριο υλοποίησης του συστήματος

- Ένας κεντρικός server
- android app στις συσκευές των χρηστών

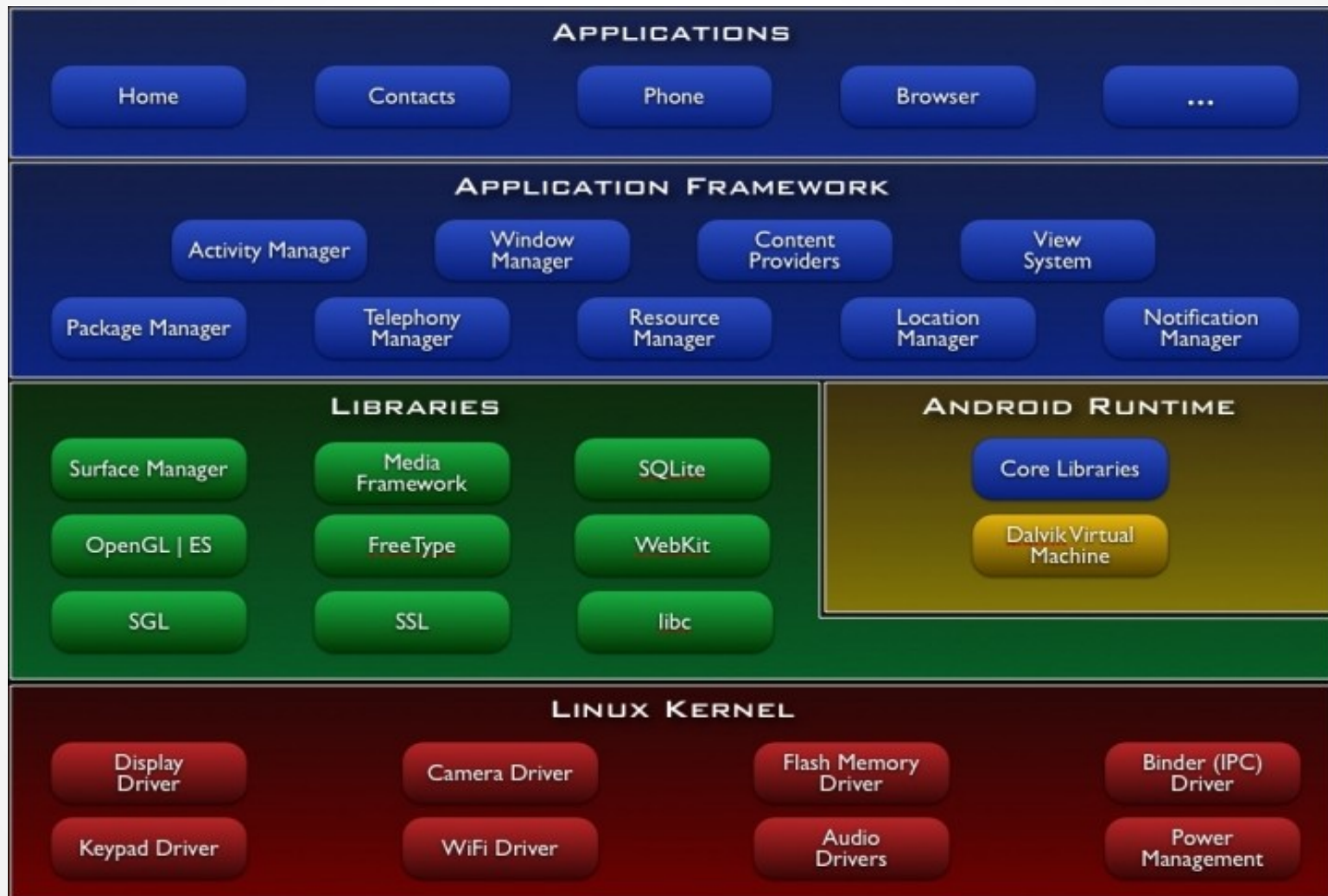
το android app

- κάθε συσκευή θα διαθέτει μία εφαρμογή η οποία θα τρέχει στο background (αλλά θα προσφέρει και interface) και θα επικοινωνεί με τον κεντρικό server.
- Θα δέχεται τα test scripts (pushes) των χρηστών (testers).
- Θα έχει την δυνατότητα να εκτελεί scripts σε ένα σύνολο από script γλώσσες προγραμματισμού (python, javascript ...).
- Θα στέλνει τα αποτελέσματα πίσω στον κεντρικό server σε μία προκαθορισμένη μορφή, ώστε να είναι διαθέσιμα στον tester και στο κοινό.
- Θα δίνει την δυνατότητα στον χρήστη να επιλέγει τις πληροφορίες που θέλει να είναι διαθέσιμες στον δίκτυο
- Θα δίνει την δυνατότητα στον χρήστη να προσπελάουν τα αποτελέσματα των testers μέσα από την συσκευή του (κίνητρο για την διαθέση των πόρων του στο δίκτυο – πρόσβαση στις πληροφορίες του περιβάλλοντος όταν τις χρειάζεται).

Ο κεντρικός server

- Θα αναλαμβάνει την επικοινωνία με τις κινητές συσκευές.
- Θα μοιράζει στο δίκτυο τα scripts tests (pushes) και θα συλλέγει τα αποτελέσματα
- Θα παρέχει ένα interface στους χρήστες (testers) για την εκτέλεση και την παρουσίαση των αποτελεσμάτων
- Θα παρέχει σε επισκέπτες τα δεδομένα των tests.

Τα επίπεδα του λειτουργικού συστήματος Android (Linux under the hood)



Το λειτουργικό σύστημα Android

- Το λειτουργικό σύστημα android είναι βασισμένο στον πυρήνα του λειτουργικού συστήματος Linux (linux kernel)
- Το Linux παρέχει δομικά χαρακτηριστικά του πυρήνα του όπως device drivers, memory and process management, network stack, and security. Ο kernel παρέχει επίσης ένα επίπεδο αφαίρεσης μεταξύ του hardware και των εφαρμογών. Μπορούμε να πούμε πως ο πυρήνας του Linux είναι ο σκελετός, οι μύες και τα όργανα ενός σώματος android.
- native libraries γραμμένες σε c/c++ και είναι γραμμένες για συγκεκριμένο υλικό
- Το android runtime αποτελείται από την εικονική μηχανή Dalvik και core java libraries. Αυτό το κομμάτι παρέχει την υποστήριξη για την core Java language και μεγάλος μέρος της λειτουργικότητας της γλώσσας Java. Κάθε εφαρμογή android τρέχει μέσα στον χώρο της δικής της διεργασίας σε ένα ιδιωτικό αντίγραφο της εικονικής μηχανής Dalvik.
Η Dalvik VM είναι ένα είδος JVM για low processing power και low memory environments. Δεν τρέχει .class αρχεία, αλλά .dex files, τα οποία παράγονται από .class αρχεία κατά το compilation.
- Το επίπεδο application framework παρέχει όλα τα απαραίτητα components που χρειάζεται για να τρέξει μία εφαρμογή android.

Android Application Anatomy

- Όλες οι εφαρμογές android είναι βασισμένες σε 3 core components
 - Activities
 - Services
 - Receivers
- Τα core components ενεργοποιούνται μέσω μηνυμάτων τα οποία ονομάζονται intents.

Activities

- Τα activities είναι components τα οποία παρέχουν μία οθόνη με την οποία η χρήστες μπορούν να αλληλεπιδράσουν ή να εκτελέσουν μία ενέργεια όπως: κλήση , χρήση της κάμερας, αποστολή e- mail, ή web browsing.
- Σε κάθε μία διεργασία παρέχεται ένα window στο οποίο ζωγραφίζει το interface του χρήστη. Το παράθυρο μπορεί να καταλαμβάνει όλη την οθόνη ή μέρος αυτής πάνω από άλλα παράθυρα.
- Οι android εφαρμογές αποτελούνται από ένα ή περισσότερα activities συζευγμένα μαζί (για παράδειγμα tabs όπου κάθε tab μπορεί να έχει το δικό του activity). Κάθε application έχει ένα main activity το οποίο μπορεί να ξεκινήσει άλλα activities.

Services

- Ένα service είναι ένα application component το οποίο δίνει την δυνατότητα σε ένα μία εφαρμογή να εκτελεί μια εργασία μεγάλης διάρκειας (longer-running) καθώς ο χρήστης αλληλεπιδρά με κάποιο άλλο μέρος της εφαρμογής (MP3 playback).
- Ένα service δεν είναι μία ξεχωριστή διεργασία, αλλά τρέχει στην διεργασία της εφαρμογής στην οποία είναι μέρος.

Receivers

- Ένας receiver είναι ένα application component το οποίο δίνει την δυνατότητα σε μία εφαρμογή να λαμβάνει intents τα οποία γίνονται broadcast από το σύστημα ή από άλλες εφαρμογές, ακόμα και αν κανένα από τα components της εφαρμογής δεν τρέχει.

Intents

- Ένα intent είναι μία αφαιρετική περιγραφή μίας ενέργειας που πρέπει να εκτελεστεί.
- Κάθε εφαρμογή ξεκινάει με ένα intent
- Ένα activity εκκινεί άλλα activities με τη χρήση intents
- Intents στέλνονται σε receivers για να κάνουν trigger handlers γεγονότων ή να ξεκινήσουν κάποιο service

Πίσω στο πρόβλημα:
Τι υπάρχει στην κοινότητα – διαθέσιμες λύσεις ...???

- Ambient Dynamix Project
- Android scripting layer (SL4A)
- on{x}

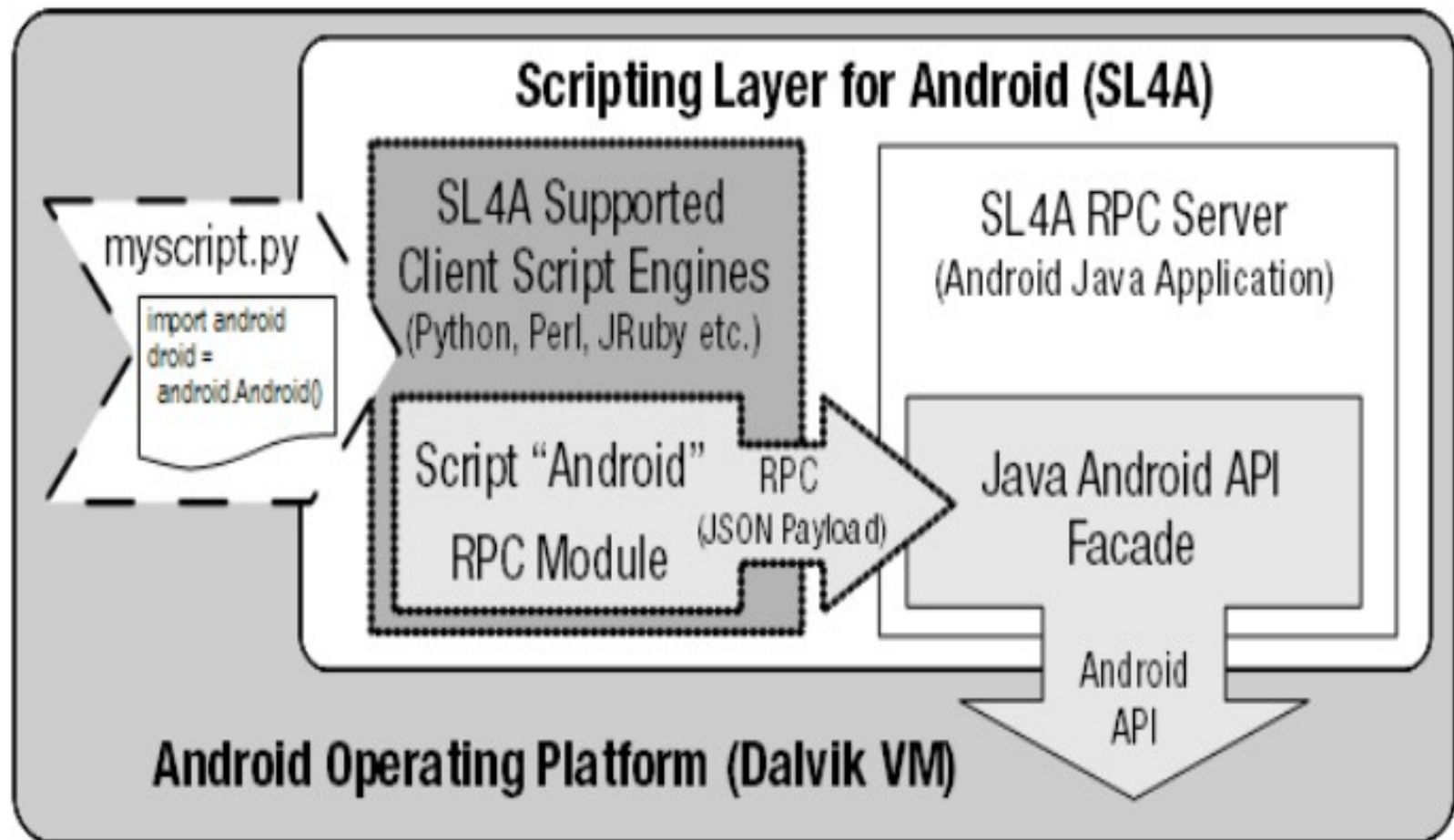
What Is Scripting Layer for Android?

- Android scripting layer (SL4A) ή Android Scripting Environment (ASE), είναι ένα open-source project
- Το android scripting layer είναι μία υποδομή που επιτρέπει την διαλειτουργικότητα των scripting μηχανών (οι οποίες έχουν γίνει ported για την πλατφόρμα του android) με το API του android μέσω remote procedure calls (RPCs) σε ένα server ο οποίος είναι υλοποιημένος ως μία standard android java εφαρμογή.
- Scripting γλώσσες που υποστηρίζει: Python, Beanshell, JRuby, Lua, Perl, PHP, and Rhino (java με χρήση javascript).
- Υπάρχει αντίστοιχη εφαρμογή android η οποία υποστηρίζει την εγκατάσταση interpreters και την εκτέλεση scripts.

SL4A Architecture

- Στο κατώτατο επίπεδο το SL4A είναι ένας scripting host to οποίο σημαίνει πως ως μία εφαρμογή περιλαμβάνει διαφορετικούς interpreters, κάθε ένας από τους οποίους επεξεργάζεται μία συγκεκριμένη scripting γλώσσα.
- Υπάρχει ένα αντίγραφο του κώδικα για κάθε μία γλώσσα που υποστηρίζει το οποίο έχει γίνει cross-compile για την ARM. αρχιτεκτονική χρησιμοποιώντας το Android Native Development Kit (NDK), και φορτώνεται ως library όταν ξεκινάει ένας interpreter. Μέτα από αυτό το script μεταφράζεται γραμμή προς γραμμή.
- Πάρέχει interface προς το λειτουργικό σύστημα, χρησιμοποιώντας remote procedure calls (RPC), και JSON.

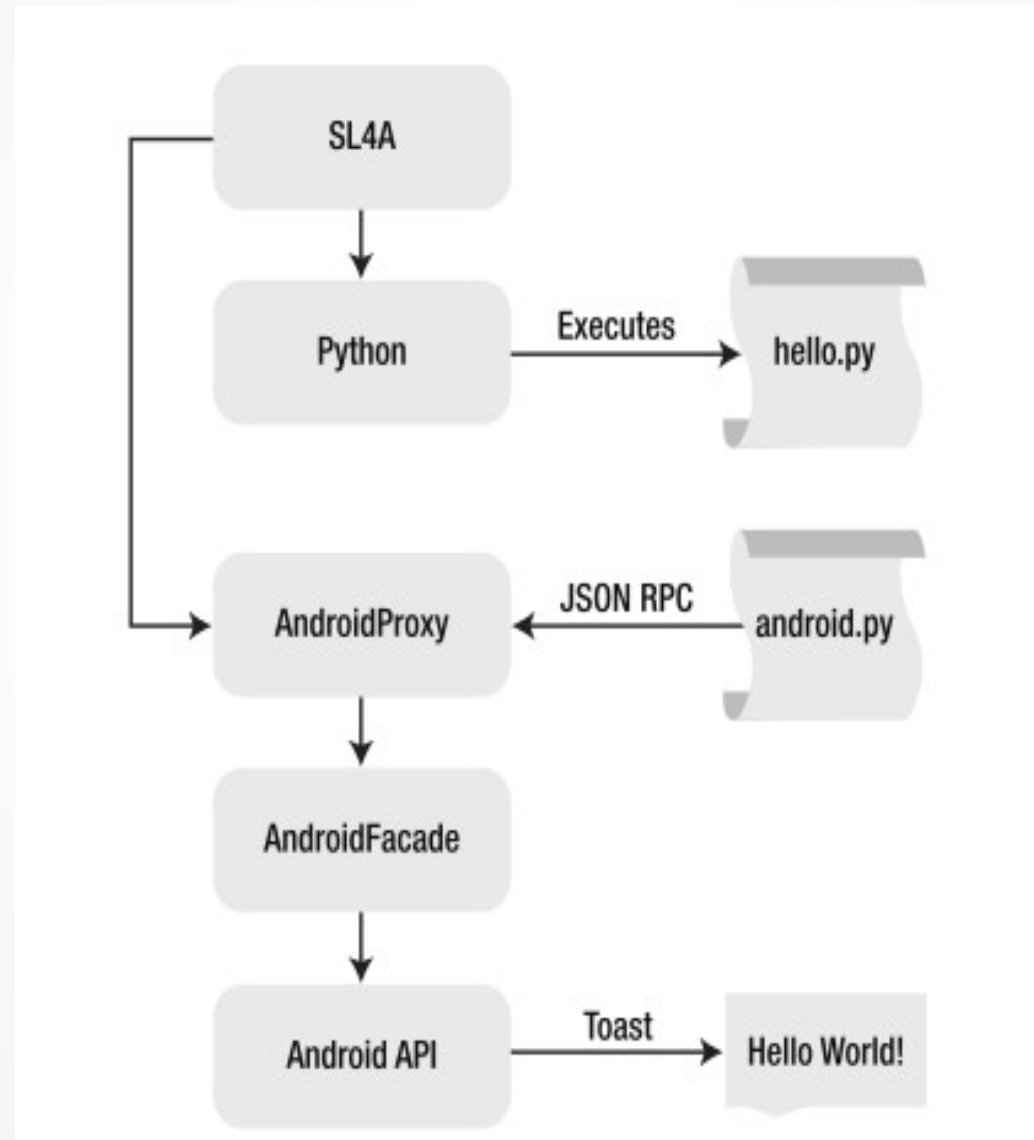
SL4A Architecture



SL4A Architecture

- Κάθε script πρέπει να ενσωματώνει (import) ένα αρχείο, όπως το `android.py` για την python, το οποίο καθορίζει ένα σύνολο από proxy συναρτήσεις που χρειάζονται για την επικοινωνία με το android API.
- Η επικοινωνία με μεταξύ του SL4A και του λειτουργικού συστήματος android γίνεται μέσω ενός μηχανισμού remote procedure calls (RPCs) και JSON. Ο server είναι το λειτουργικό σύστημα του android και ο client το SL4A script.
- Κάθε interpreter εκτελεί την γλώσσα μέσα στον χώρο της διεργασίας του μέχρι να γίνει ένα API call. Αυτό μεταβιβάζεται στο λειτουργικό σύστημα του android με το μηχανισμό RPC. Όλη η επικοινωνία του interpreter με το android API κάνει χρήση της JSON για την μεταβίβαση της πληροφορίας.
- Κάθε υποστηριζόμενη API συνάρτηση έχει ένα αντίστοιχο interface για κάθε μία γλώσσα, το οποίο ονομάζεται facade (πρόσοψη). Για παράδειγμα `BatteryManagerFacade`, `BluetoothFacade`, `LocationFacade`, `SensorManagerFacade`

launch a python script with SL4A: flow of execution



Run javascript to android

- Javascript scripting με Rhino (java με χρήση javascript)



- Rhino

Rhino είναι μία open-source υλοποίηση της javascript γραμμένη αποκλειστικά σε java. Είναι συνήθως ενσωματωμένη σε java εφαρμογές για να παρέχει δυνατότητές scripting στους χρήστες.

Παράδειγμα εκτέλεσης javascript σε android εφαρμογή

- Χρειάζεται μόνο η εισαγωγή του js.jar από το Rhino project στο φάκελο libs της εφαρμογής (~1,1 MB)
- ```
import org.mozilla.javascript.*;
```
- ```
void doit(String code)
{
    Context cx = Context.enter();

    cx.setOptimizationLevel(-1);

    try
    {
        Scriptable scope = cx.initStandardObjects();

        ScriptableObject.putProperty(
            scope, "TheActivity", Context.javaToJS(this, scope));

        cx.evaluateString(scope, code, "doit:", 1, null);
    }
    finally
    {
        Context.exit();
    }
}
```
- ```
doit("var log = android.util.Log;\n" +
 "log.i(\"HITAG\", \"hello happy android from javascript\");\n");
```

# Παράδειγμα εκτέλεσης javascript σε android εφαρμογή

- `Context cx = Context.enter();`  
Κάθε thread που εκτελεί javascript πρέπει να συσχετίζεται με ένα Context
- `cx.setOptimizationLevel(-1);`  
Turn compilation to make Rhino Android compatible. JavaScript will always be interpreted
- `Scriptable scope = cx.initStandardObjects();`  
Initialize the standard objects. Creates instances of the standard objects and their constructors
- `ScriptableObject.putProperty ( scope, "TheActivity", Context.javaToJS(this, scope));`  
Puts a named property in an object
- `cx.evaluateString(scope, code, "doit:", 1, null);`  
Evaluate a JavaScript source string .argument is 1, which means the JavaScript source has been compressed to only on 1 line
- `Context.exit();`  
exit the Rhino VM. will remove the association between the current thread

# A hello World – The Code Cannon

Fire my code



```
var log = android.util.Log;
log.i("HITAG","hello happy android from javascript");
```

Fire Python(!)

```
n.android.settings ddm-heap
n.android.python27 downloader
n.android.python27 global
n.android.python27 downloader
n.android.python27 global
n.android.python27 dalvikvm
n.android.python27 HITAG
roid.process.acore dalvikvm
```



Got feature list request

download url: <http://83.212.115.57/cannon/payloads/payload.js>

Default buffer size used in BufferedInputStream constructor. It would be better to be explicit if an 8k buffer is required.

downloading 83.0 bytes completed in 0 sec to destination: /data/data/com.android.python27/files

Default buffer size used in BufferedReader constructor. It would be better to be explicit if an 8k-char buffer is required.

GC freed 4905 objects / 324680 bytes in 123ms

hello happy android from javascript

GC freed 2900 objects / 171064 bytes in 117ms

## Others ???

- Επικοινωνία των apps με τον κεντρικό server ???
- Πως θα γίνονται push τα scripts στις συσκευές ???
- polling από τις συσκευές στον server για να δουν αν υπάρχει test που πρέπει να κατεβάσουν και να εκτελέσουν

Better ???

- Security ??? Privacy ???

Write your own malware and use our service to distribute it

It doesn't matter what language you use...we support all of them

# References - Links

- <https://code.google.com/p/android-scripting/>
- Practical Android Projects (Ch05 Introducing SL4A )
- Pro android python with sl4a
- <https://developer.mozilla.org/en/docs/Rhino>
- <https://developer.android.com/index.html>
- <http://divineprogrammer.blogspot.gr/2009/11/javascript-rhino-on-android.html>
- <https://github.com/PanosRCng/payloadManager>