

# Google Android

Sean C. Sullivan  
Portland Java User Group  
June 2008

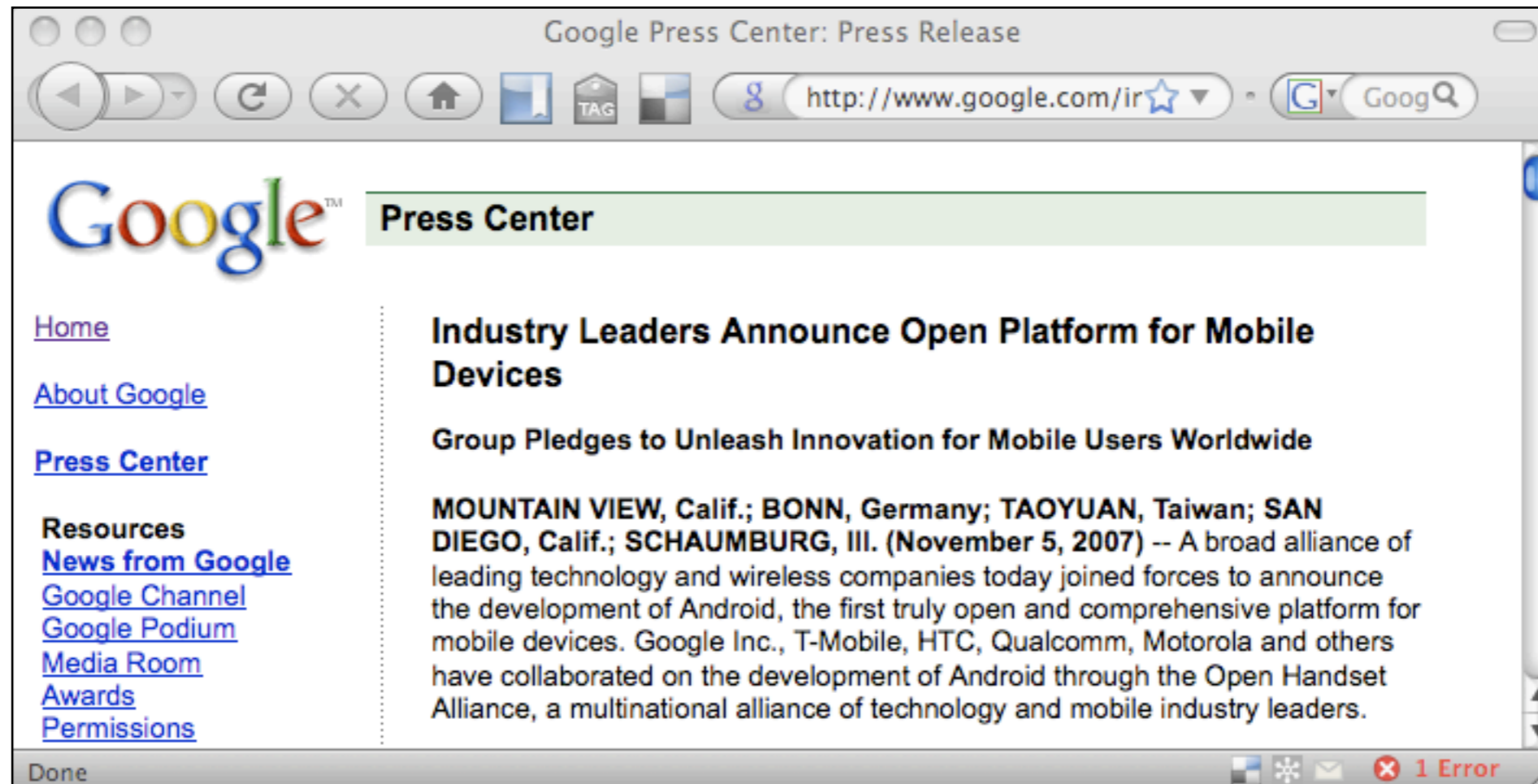


# Topics

- What is Android?
- Android platform
- Developer tools
- Android applications



# Android



November 5, 2007



# Android

## Google Enters the Wireless World

By MIGUEL HELFT and JOHN MARKOFF

Published: November 6, 2007

SAN FRANCISCO, Nov. 5 — What [Apple](#) began with its [iPhone](#), [Google](#) is hoping to accelerate, with an ambitious plan to transform the software at the heart of cellphones.

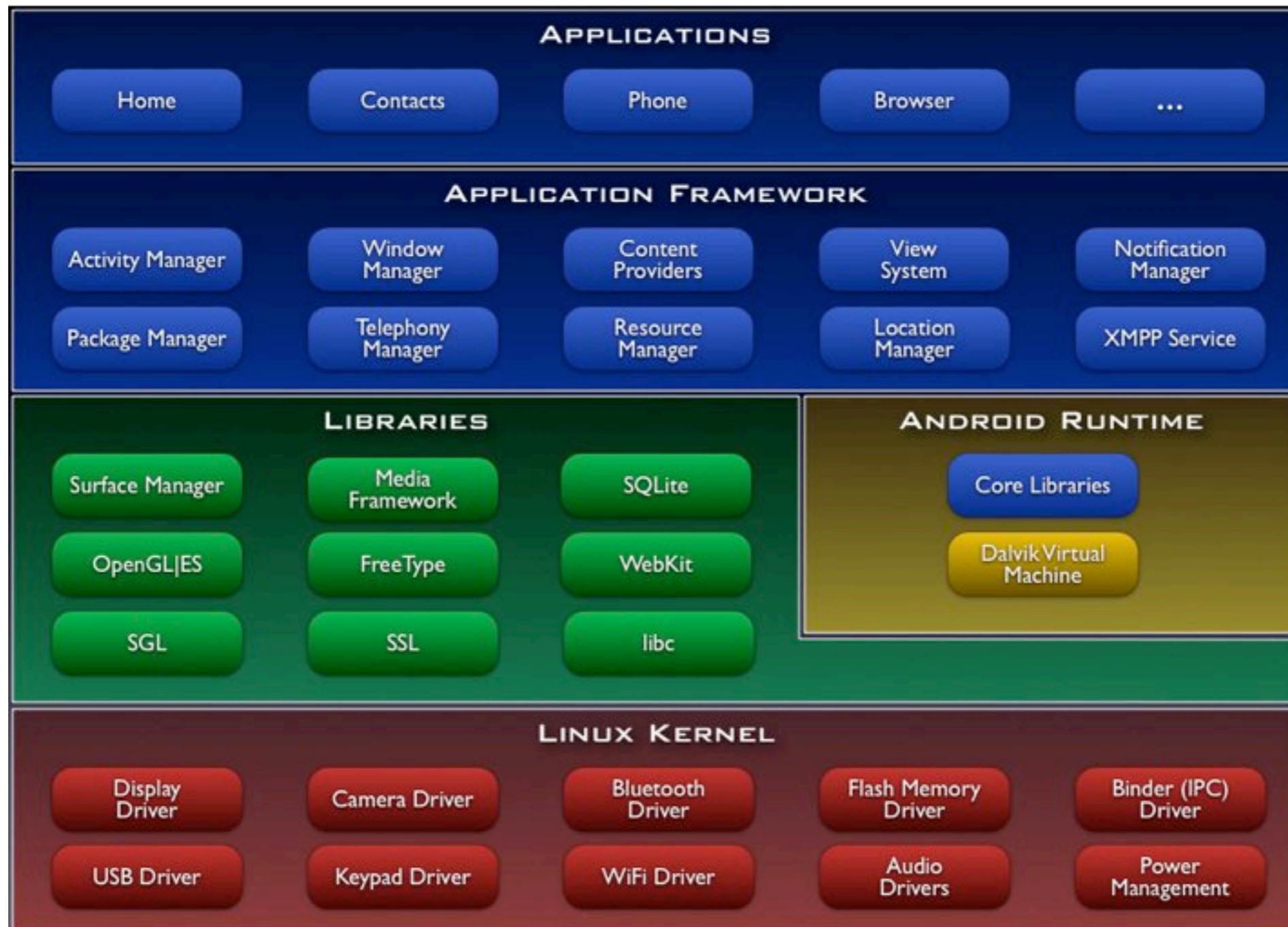
**The New York Times**



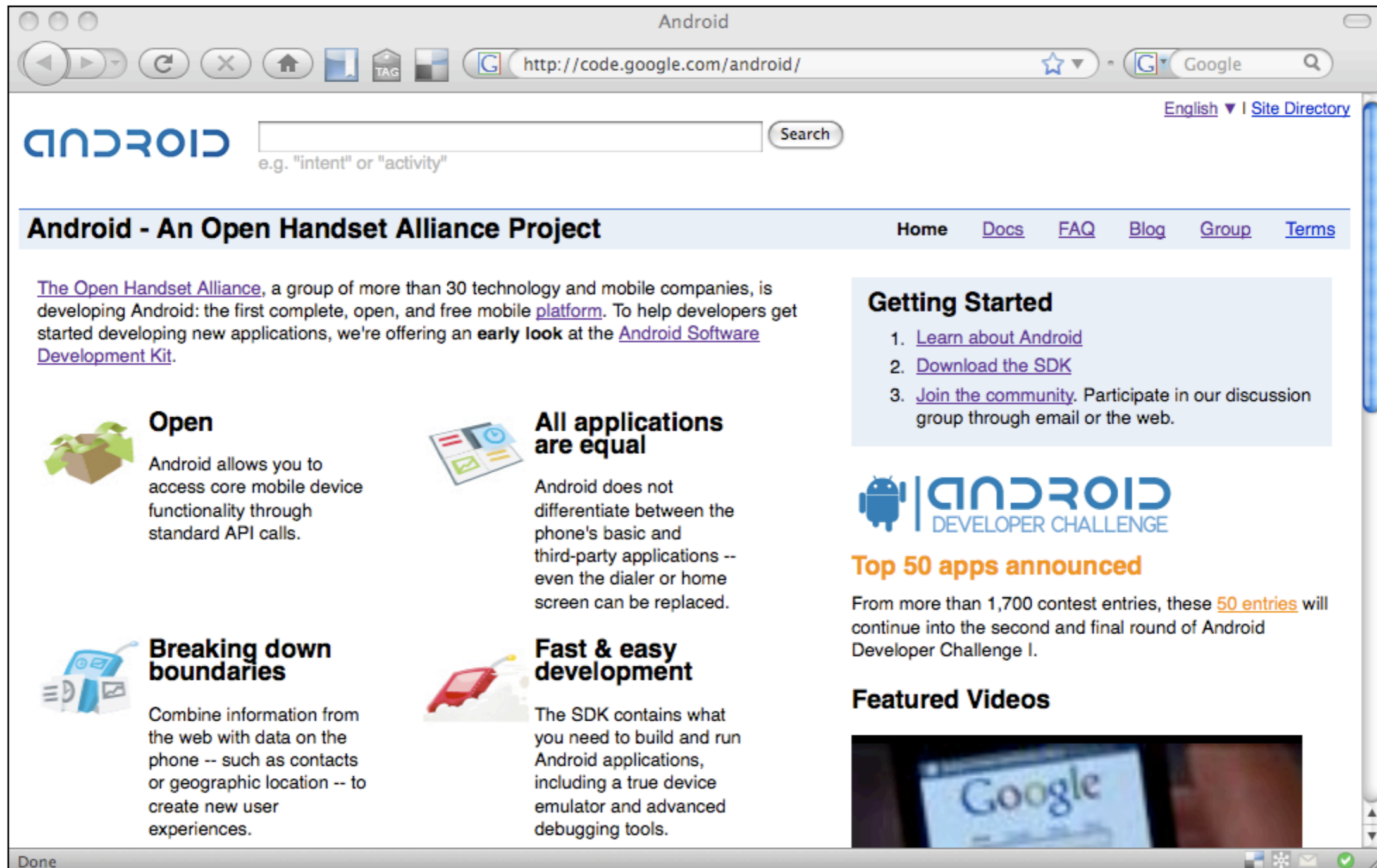
# What is Android?

*“Android is a software stack for mobile devices that includes an operating system, middleware and key applications”*





# Getting started



The screenshot shows a browser window with the URL <http://code.google.com/android/>. The page features the Android logo, a search bar, and a navigation menu with links for Home, Docs, FAQ, Blog, Group, and Terms. The main content area is titled "Android - An Open Handset Alliance Project" and includes a paragraph about the Open Handset Alliance. Below this are four feature highlights: "Open", "All applications are equal", "Breaking down boundaries", and "Fast & easy development". On the right side, there is a "Getting Started" section with a list of three steps: "Learn about Android", "Download the SDK", and "Join the community". Below that is a section for the "Android Developer Challenge" with a sub-section for "Top 50 apps announced". At the bottom right, there is a "Featured Videos" section with a video player showing a Google search on a mobile device.

Android

<http://code.google.com/android/> Google

English | Site Directory


**ANDROID** Search

e.g. "intent" or "activity"

## Android - An Open Handset Alliance Project


Home Docs FAQ Blog Group Terms

The [Open Handset Alliance](#), a group of more than 30 technology and mobile companies, is developing Android: the first complete, open, and free mobile [platform](#). To help developers get started developing new applications, we're offering an **early look** at the [Android Software Development Kit](#).




### Open

Android allows you to access core mobile device functionality through standard API calls.




### All applications are equal

Android does not differentiate between the phone's basic and third-party applications -- even the dialer or home screen can be replaced.



### Breaking down boundaries

Combine information from the web with data on the phone -- such as contacts or geographic location -- to create new user experiences.




### Fast & easy development

The SDK contains what you need to build and run Android applications, including a true device emulator and advanced debugging tools.

## Getting Started

1. [Learn about Android](#)
2. [Download the SDK](#)
3. [Join the community](#). Participate in our discussion group through email or the web.

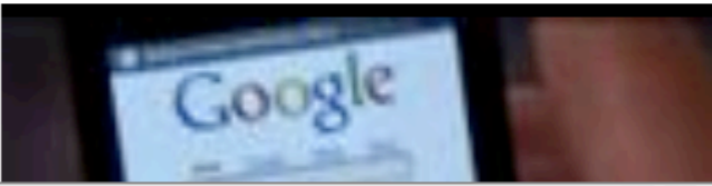


## Android Developer Challenge

### Top 50 apps announced

From more than 1,700 contest entries, these [50 entries](#) will continue into the second and final round of Android Developer Challenge I.

### Featured Videos



<http://code.google.com/android>



# Development tools

- Android SDK
- Eclipse plugin





# Android SDK

- Android emulator
- command line tools
- documentation
- example applications

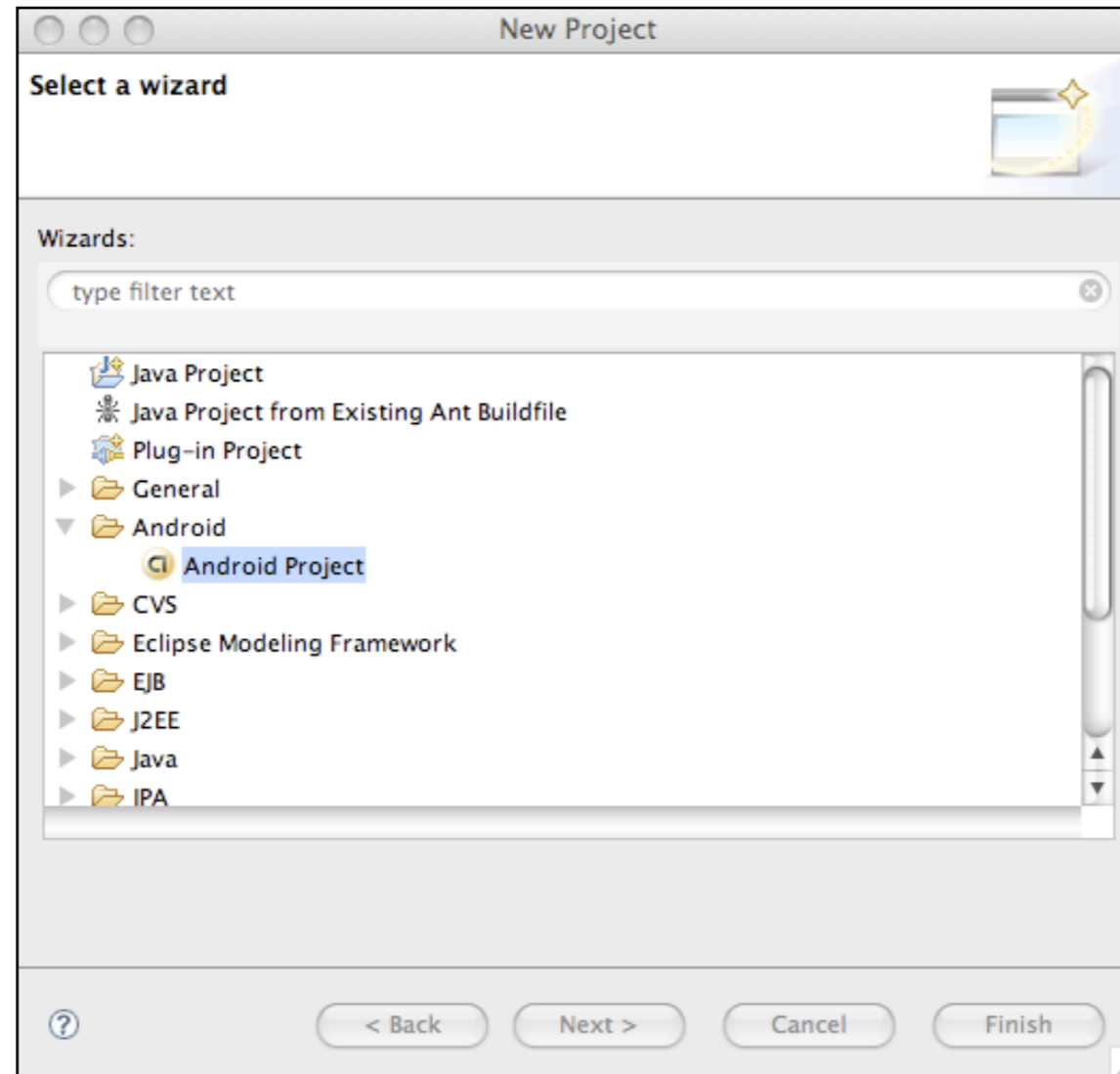


# Command line tools

- *aapt* - Android asset packaging tool
- *adb* - Android debug bridge
- *aidl* - Android IDL compiler
- *emulator* - Android emulator



# Eclipse plugin



<https://dl-ssl.google.com/android/eclipse/>



# Android applications

- application package file: *myapp.apk*
- an application is composed of one or more *activities*



# Activity

- an activity is usually a single screen in your application
- however, activities can also be faceless
- one activity is designated as the entry point for your application



# android.app.Activity

```
import android.app.Activity;

public class MyActivity extends Activity
{
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```



# Application building blocks

- Activities
- AndroidManifest.xml
- Views
- Layouts
- Intents & IntentReceivers
- Services
- Notifications
- ContentProviders



# Implementing your application UI

- Java code
- XML





# Android UI

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Hello World"
    />
</LinearLayout>
```



# AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.my_domain.app.helloactivity">
    <application android:label="@string/app_name">
        <activity android:name=".HelloActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
    </application>
</manifest>
```



# Intents

- “an Intent is a simple message object that represents an ‘intention’ to do something”
- “an intent is an abstract description of an operation to be performed”



# android.content.Intent

- VIEW\_ACTION
- EDIT\_ACTION
- PICK\_ACTION
- WEB\_SEARCH\_ACTION
- SYNC\_ACTION
- ...



# Views

- an object that knows how to draw itself on the screen
- examples: `ListView`, `DatePicker`, `Button`



# Web browser

- based on WebKit
- <http://www.webkit.org>
- android.webkit.WebView



# Local data store

- SQLite
- <http://www.sqlite.org>
- `android.database.sqlite`



# Application Context

android.app.ApplicationContext

- *startActivity(Intent)*
- *getSystemService*
- *createDatabase*
- *openDatabase*
- *deleteDatabase*
- ...





# Additional topics

- Threading
- Security model
- Internationalization
- Power management
- AIDL - Android IDL
- API for Bluetooth
- API for WiFi



# Android resources

- <http://code.google.com/android/>
- <http://android-developers.blogspot.com>
- <http://code.google.com/p/apps-for-android/>
- <http://sites.google.com/site/io/>
- <http://www.openhandsetalliance.com/>
- <http://git.android.com>

