



Android WebKit Development: A cautionary tale

Joe Bowser

Nitobi

E-Mail: joe.bowser@nitobi.com

About this talk

- This talk is not explicitly about PhoneGap
- This is a technical talk
 - It is expected that you have looked at the Android SDK (ApiDemos, Hello World)
 - This will make more sense if you have gone to the other PhoneGap talks today as well
- There will be screenshots
- This won't be just a rant! (I promise)
- This won't be an Apple Fanboy Talk

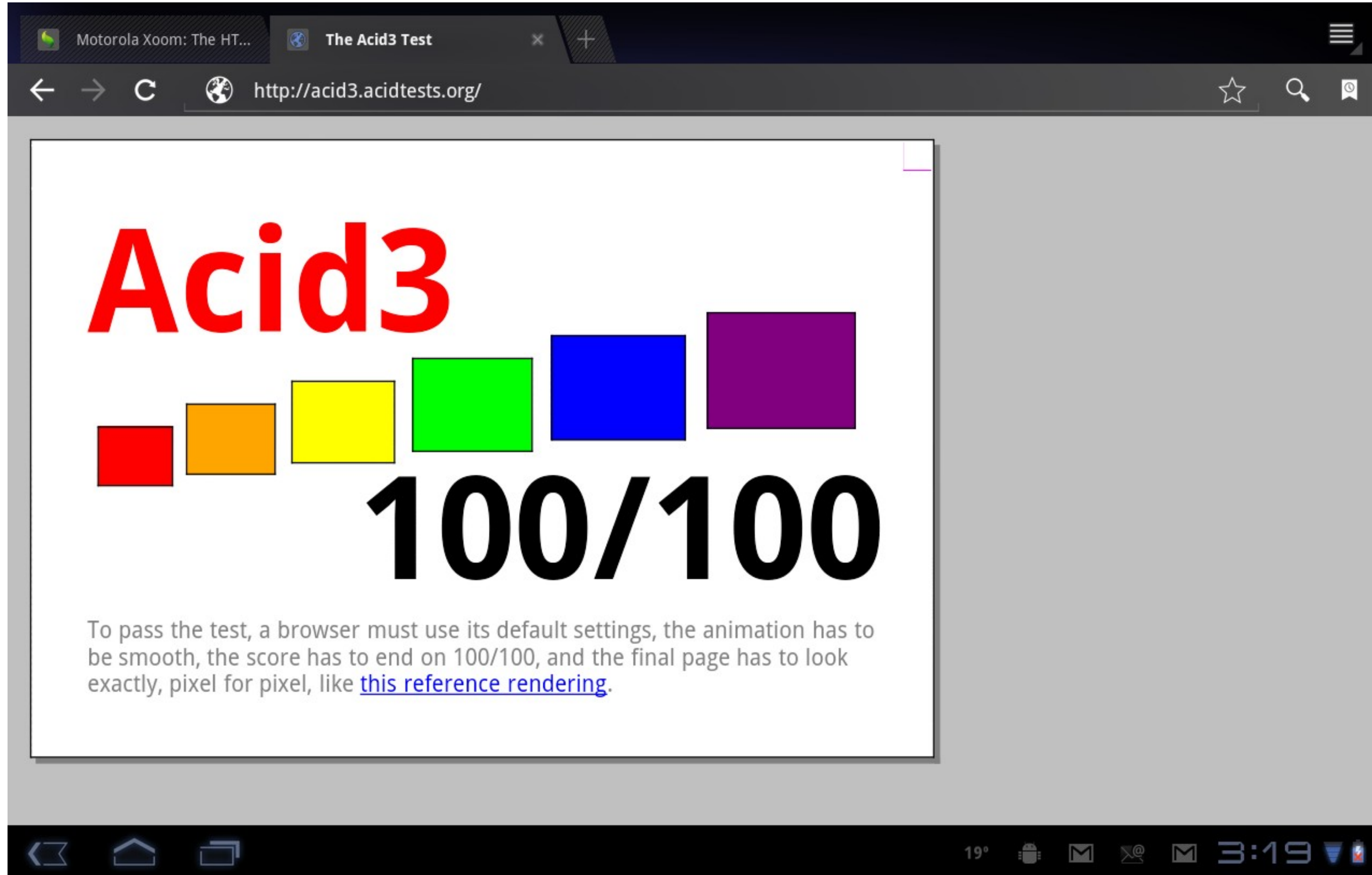
About the Speaker

- First commit on PhoneGap-Android
- Did minor work on other Open Source Projects
- Worked with Android since 2008 (Version 1.0, T-Mobile G1 release days)
- Co-founded Vancouver Hack Space (hackspace.ca)
- Currently still work with PhoneGap (I do dogfood my stuff), but also work with the ADK and other Hardware-Related stuff on my spare time
- I definitely do NOT work for Google, I just get to hear people's complaints about the Android Browser
- All my experiences with this are from OUTSIDE Google
- I feel your pain!!!

Android Web Development

- When someone develops a mobile web app, they will have to deal with the Android Browser
- Android Browser is an application that is different on EVERY SINGLE device released by a manufacturer
- Android Browser handles copy/paste functionality, dialog boxes, text boxes, “Native-Style” UI
- Android Browser is written in Java and is the visible app for users

FAIL



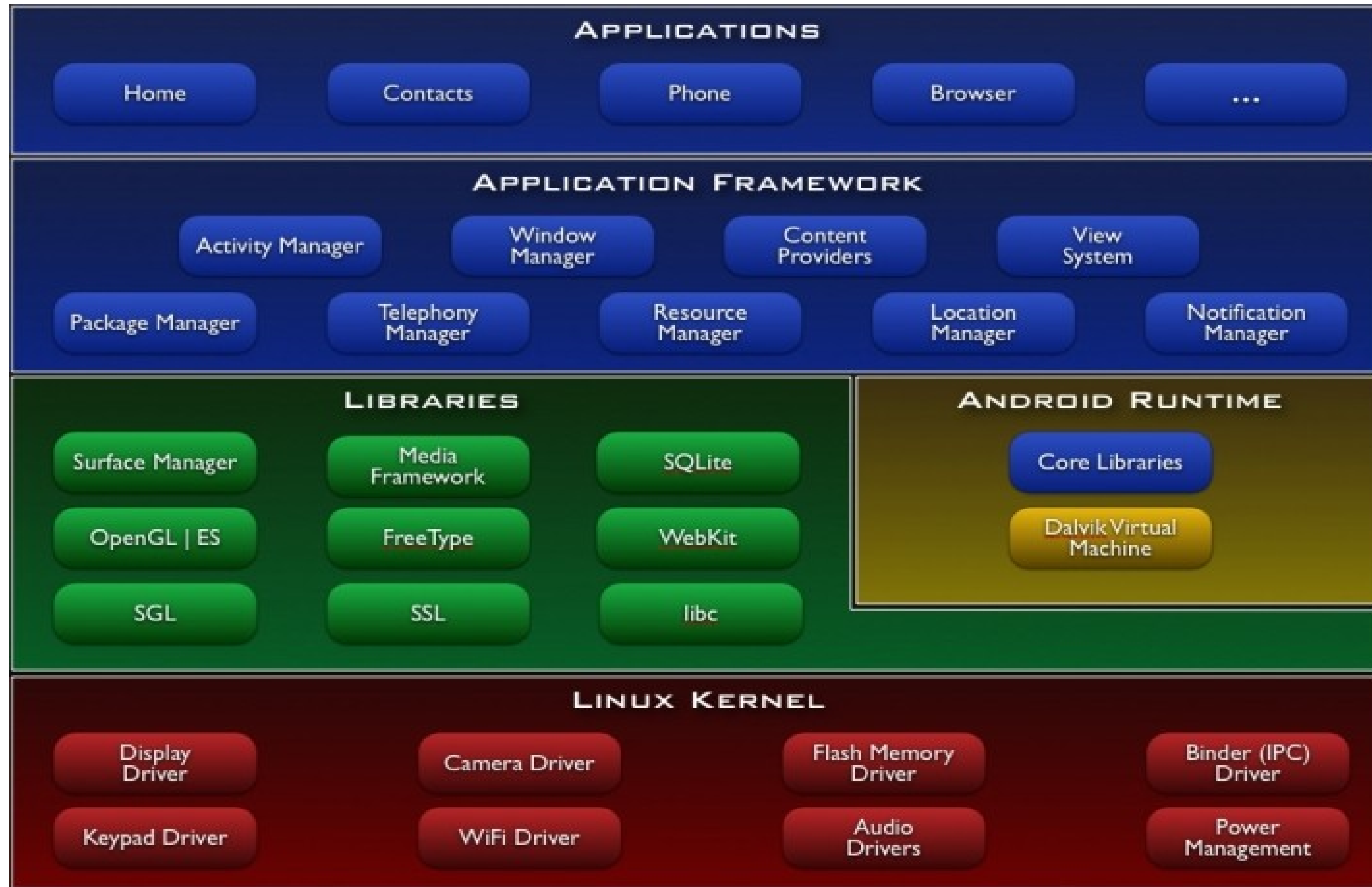
But doesn't Android use WebKit

- Android DOES use WebKit, through Android Browser
- WebKit is NOT a browser
- WebKit is NOT written in Java
- WebKit can't create Dialog Boxes or any of the other Native UI elements
 - Webkit can render HTML and Javascript, but these are NOT consistent with the look and feel of Android's UI
- WebKit renders graphics in a completely different way than the rest of Android, and these graphics are pushed onto a surface for Java applications to layer components over top of

What is WebKit?

- From webkit.org:
 - Webkit is a modular open source browser engine, but it is NOT A BROWSER
 - WebKit is a fork of KHTML, and is used everywhere
 - Android, iOS, Blackberry OS 6, Chrome, Safari, etc
 - WebKit is written in C++
 - WebKit is NOT the solution to every problem

Android Layer Cake



Android Browser's Cake

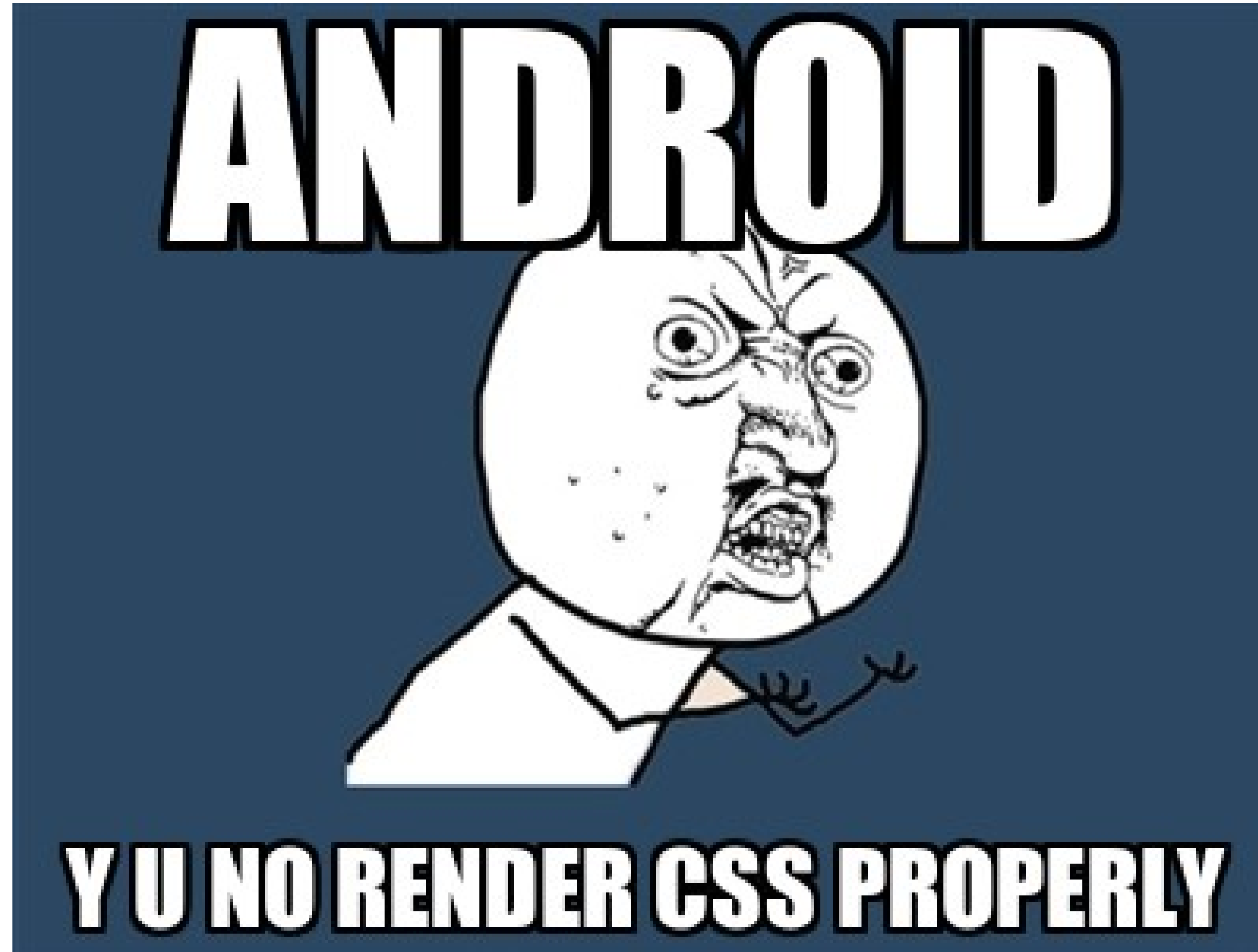


- The world of Android WebKit looks something like this
- WebKit displays pages using the Android WebView
- WebView is a complex view that consists of other views
- The implementation of WebView directly influences how WebKit interacts with the Android UI
- It is often not clear where the differences between WebView end and WebKit begin, making issues with Android WebView and Android WebKit harder to track down
- Android WebView is part of the Android Apache source, Android WebKit is LGPL/MIT

But I'm writing Web Apps?

- Writing Web Applications using HTML, CSS and Javascript can get most applications done very quickly
- The applications have to use whatever version of WebKit is on the Android Device
- Every Manufacturer has a different implementation of webkit that are optimized for their device
- Every device released has a slightly different version as well
- Buying every device on the market is prohibitively expensive and many companies have cashed in on this fact with their own solutions
 - Some companies provide Device Rental
 - Device Anywhere

Common complaints about Android and WebKit



Serious Differences

- Android Versions
 - Old Android 1.x issues
 - Touch Events don't exist
 - No Storage by default
 - Android 2.1+
 - No REAL Multi-Touch (very glitchy, works in serial, crashes device)
 - Android 2.3 doesn't have a working addJavascriptInterface
 - Android 3.0+ can't handle “#” or “?” in some URLs (Seriously)
- Manufacturer Differences
 - HTC phones don't have a working console.log
 - HTC, Samsung and Motorola all have their own copy-paste behavior
- Different screen sizes between devices

What about testing?

- Android has the Compatibility Test Suite (CTS)
- The CTS covers Android, but does not cover WebKit
- The CTS tests don't appear to make or break whether a device is released, but do determine whether it's a Google Branded Device
- CTS may be why there's no low-end Android 2.3 devices
 - Low-RAM devices uses the JSC Javascript Engine instead of V8, which breaks the WebKit bridge, which allows developers to hook Java objects as Javascript Globals
 - CTS 2.3 has a test for the method addJavascriptInterface, which uses the WebKit bridge

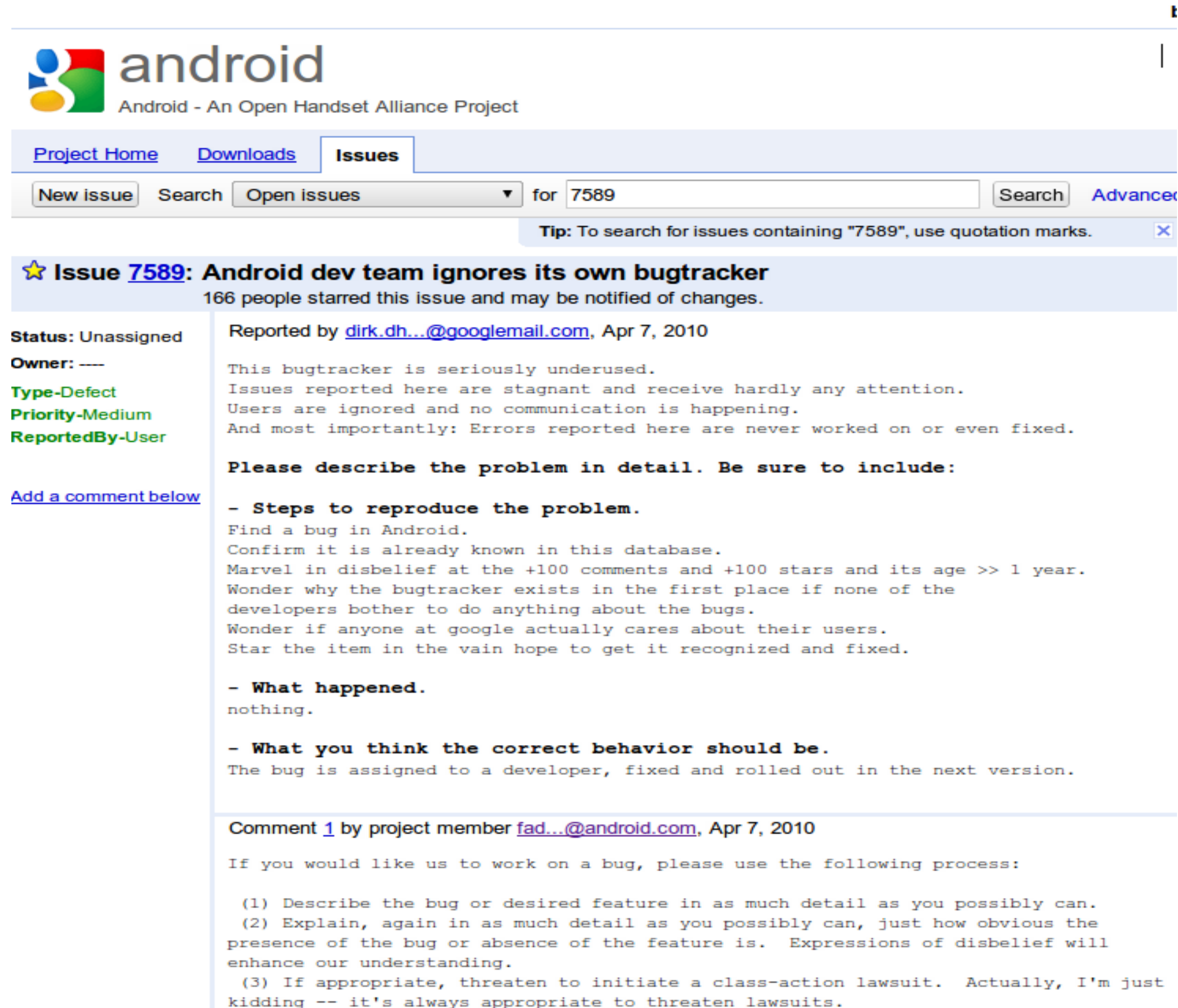
Android is open source, how about fixing it

- Android Open Source Development is HARD
- Android does have a way to commit bug fixes (gerrit)
- WebKit is WebKit
 - Very different than Mozilla and other more developer-friendly projects (no good “First Bug” tags)
 - It IS possible to fix bugs in WebKit, but it's very difficult to track WebKit development
- A good place to start if you go down this path are the Google Groups
 - Android-Building : How to build the AOSP
 - Android-Platform : Discussions on what should be changed in the Android Platform

How do we, as Application Developers deal with all of this?

- Test on as many different devices as possible
- The Android Emulator (as slow as it may be) is your friend with these bugs
 - Sony Ericsson (developer.sonyericsson.com)
 - Samsung (Galaxy Tab only, through the Android manager)
 - Motorola (developer.motorola.com)
 - LG (through Android Manager)
 - Kyocera (through Android Manager)
- Use tools like `weinre`
- Avoid CSS 3D Transforms – They **STILL** don't work (as of Android 3.1)

Will it get better?



The screenshot shows the Android issue tracker interface. At the top, the Android logo and the text "android - An Open Handset Alliance Project" are visible. Below this, there are navigation tabs for "Project Home", "Downloads", and "Issues". A search bar contains the text "7589" and a "Search" button. A tip below the search bar reads: "Tip: To search for issues containing '7589', use quotation marks." The main content area displays the details for "Issue 7589: Android dev team ignores its own bugtracker", which has 166 stars. The issue is reported by "dirk.dh...@gmail.com" on April 7, 2010. The status is "Unassigned" and the type is "Defect". The description of the issue is as follows:

This bugtracker is seriously underused. Issues reported here are stagnant and receive hardly any attention. Users are ignored and no communication is happening. And most importantly: Errors reported here are never worked on or even fixed.

Please describe the problem in detail. Be sure to include:

- **Steps to reproduce the problem.**
Find a bug in Android.
Confirm it is already known in this database.
Marvel in disbelief at the +100 comments and +100 stars and its age >> 1 year.
Wonder why the bugtracker exists in the first place if none of the developers bother to do anything about the bugs.
Wonder if anyone at google actually cares about their users.
Star the item in the vain hope to get it recognized and fixed.
- **What happened.**
nothing.
- **What you think the correct behavior should be.**
The bug is assigned to a developer, fixed and rolled out in the next version.

Below the description, there is a comment from a project member "fad...@android.com" dated April 7, 2010. The comment provides a process for reporting bugs:

If you would like us to work on a bug, please use the following process:

- (1) Describe the bug or desired feature in as much detail as you possibly can.
- (2) Explain, again in as much detail as you possibly can, just how obvious the presence of the bug or absence of the feature is. Expressions of disbelief will enhance our understanding.
- (3) If appropriate, threaten to initiate a class-action lawsuit. Actually, I'm just kidding -- it's always appropriate to threaten lawsuits.

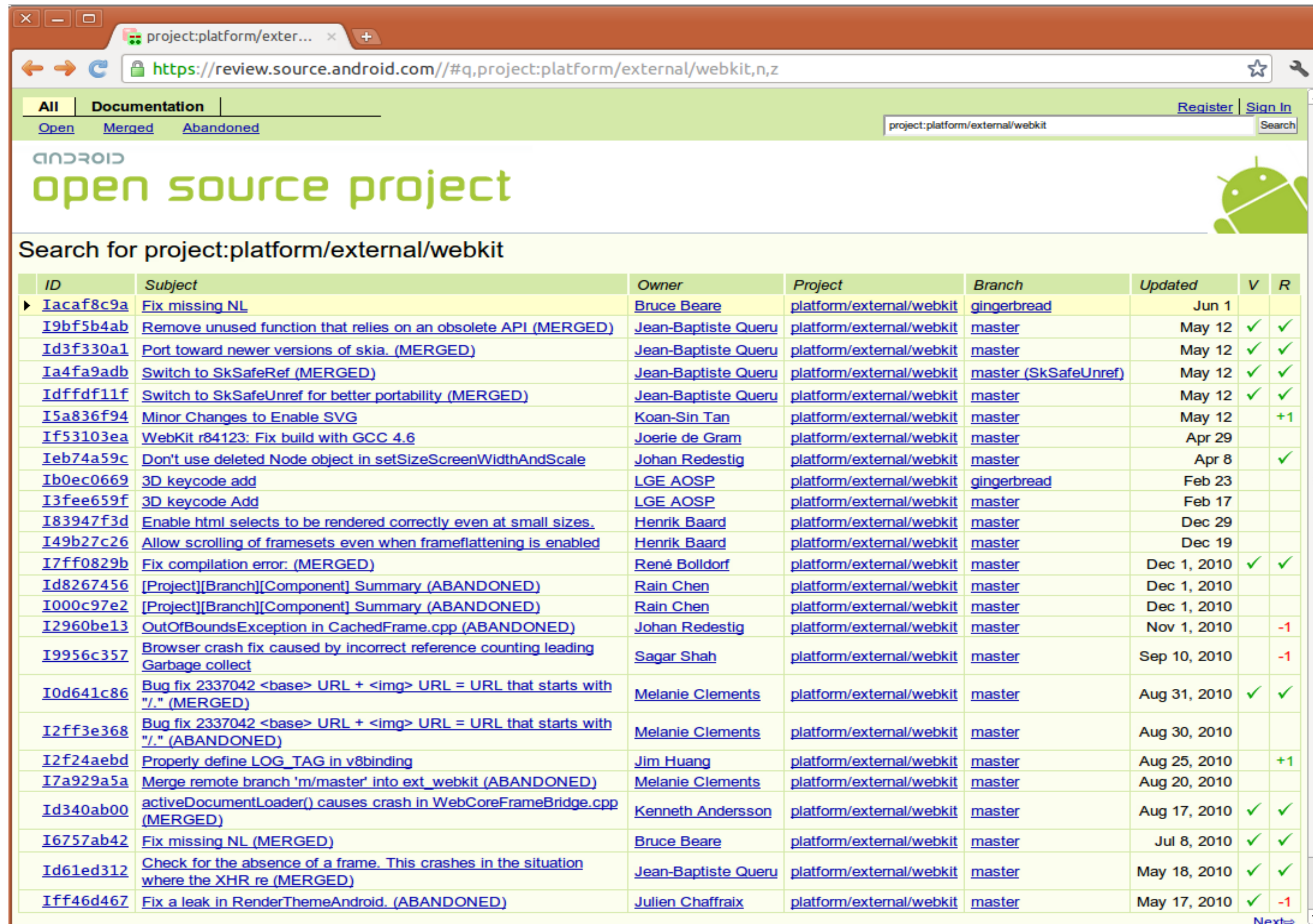
I believe that one day it will

- There are various bugs in the tracker that I think should be fixed:
 - 17485 – Poor performance of WebView (3D CSS Transforms)
 - 12987 – Javascript to Java Bridge throws Exceptions on Android 2.3
 - 16312 – addJavascriptInterface: no type checking
 - URLs should be rock-solid (should not break the browser if you use a ? or a # in it)
- There are features that I would like to see
 - Running Javascript from Java (`webView.loadUrl("javascript: foo()")`) is considered harmful)

Success

- Success from outside developer influence
 - Android Web Applications section added due to large number of PhoneGap applications
 - Android SQLite Storage Added in Android 2.1
 - Android DOM storage added shortly after
- Successful patches to WebKit
 - Android accepts patches from 3rd Parties
 - Android WebKit development is out in the open
 - Android Browser is still a part of the Apache-Licensed closed dev model

WebKit Gerrit



The screenshot shows a web browser window displaying the Gerrit review interface for the project:platform/external/webkit. The page header includes the Android Open Source Project logo and navigation links like 'All', 'Documentation', 'Open', 'Merged', and 'Abandoned'. A search bar is present with the text 'project:platform/external/webkit'. Below the header, a table lists various code reviews with columns for ID, Subject, Owner, Project, Branch, Updated, V (Verified), and R (Reviewed).

ID	Subject	Owner	Project	Branch	Updated	V	R
Iacaf8c9a	Fix missing NL	Bruce Beare	platform/external/webkit	gingerbread	Jun 1		
I9bf5b4ab	Remove unused function that relies on an obsolete API (MERGED)	Jean-Baptiste Quere	platform/external/webkit	master	May 12	✓	✓
Id3f330a1	Port toward newer versions of skia. (MERGED)	Jean-Baptiste Quere	platform/external/webkit	master	May 12	✓	✓
Ia4fa9adb	Switch to SkSafeRef (MERGED)	Jean-Baptiste Quere	platform/external/webkit	master (SkSafeUnref)	May 12	✓	✓
Idffdf11f	Switch to SkSafeUnref for better portability (MERGED)	Jean-Baptiste Quere	platform/external/webkit	master	May 12	✓	✓
I5a836f94	Minor Changes to Enable SVG	Koan-Sin Tan	platform/external/webkit	master	May 12		+1
If53103ea	WebKit r84123: Fix build with GCC 4.6	Joerie de Gram	platform/external/webkit	master	Apr 29		
Ieb74a59c	Don't use deleted Node object in setSizeScreenWidthAndScale	Johan Redestig	platform/external/webkit	master	Apr 8		✓
Ib0ec0669	3D keycode add	LGE AOSP	platform/external/webkit	gingerbread	Feb 23		
I3fee659f	3D keycode Add	LGE AOSP	platform/external/webkit	master	Feb 17		
I83947f3d	Enable html selects to be rendered correctly even at small sizes.	Henrik Baard	platform/external/webkit	master	Dec 29		
I49b27c26	Allow scrolling of framesets even when frameflattening is enabled	Henrik Baard	platform/external/webkit	master	Dec 19		
I7ff0829b	Fix compilation error. (MERGED)	René Bolldorf	platform/external/webkit	master	Dec 1, 2010	✓	✓
Id8267456	[Project][Branch][Component] Summary (ABANDONED)	Rain Chen	platform/external/webkit	master	Dec 1, 2010		
I000c97e2	[Project][Branch][Component] Summary (ABANDONED)	Rain Chen	platform/external/webkit	master	Dec 1, 2010		
I2960be13	OutOfBoundsException in CachedFrame.cpp (ABANDONED)	Johan Redestig	platform/external/webkit	master	Nov 1, 2010		-1
I9956c357	Browser crash fix caused by incorrect reference counting leading Garbage collect	Sagar Shah	platform/external/webkit	master	Sep 10, 2010		-1
I0d641c86	Bug fix 2337042 <base> URL + URL = URL that starts with "/." (MERGED)	Melanie Clements	platform/external/webkit	master	Aug 31, 2010	✓	✓
I2ff3e368	Bug fix 2337042 <base> URL + URL = URL that starts with "/." (ABANDONED)	Melanie Clements	platform/external/webkit	master	Aug 30, 2010		
I2f24aebd	Properly define LOG_TAG in v8binding	Jim Huang	platform/external/webkit	master	Aug 25, 2010		+1
I7a929a5a	Merge remote branch 'm/master' into ext_webkit (ABANDONED)	Melanie Clements	platform/external/webkit	master	Aug 20, 2010		
Id340ab00	activeDocumentLoader() causes crash in WebCoreFrameBridge.cpp (MERGED)	Kenneth Andersson	platform/external/webkit	master	Aug 17, 2010	✓	✓
I6757ab42	Fix missing NL (MERGED)	Bruce Beare	platform/external/webkit	master	Jul 8, 2010	✓	✓
Id61ed312	Check for the absence of a frame. This crashes in the situation where the XHR re (MERGED)	Jean-Baptiste Quere	platform/external/webkit	master	May 18, 2010	✓	✓
If46d467	Fix a leak in RenderThemeAndroid. (ABANDONED)	Julien Chaffraix	platform/external/webkit	master	May 17, 2010	✓	-1

How do I make the Mobile Web better

- Write more code – Work on PhoneGap, write Apps, work on WebKit directly
- Use PhoneGap and other Web Technologies when appropriate
- Write tests and show them to people. Writing comments like “This doesn't work” isn't as helpful as explaining why and showing a stack trace
- Remember that while some parts of HTML5 code for the iPhone won't work on Android, that code will work properly on the iPhone
- Don't Give Up

If I wanted to hack on Android WebKit, how would I do it

- **DISCLAIMER:** I have not committed ANY code to WebKit, I only got this working a few months ago
- This will be a step-by-step guide to debugging WebKit on the AOSP
- This will be based on Screenshots, because I can't guarantee that the AOSP will compile, or that it will run (I do not work for Google)
- There are probably far better ways to do this, this is the way that I've pieced together from reading the Google Groups
- Make sure that you have a proper build setup first before doing this, otherwise you will spend hours chasing down build dependencies
- See source.android.com for more information

Step One: Get the Source code for AOSP

- Setup build environment (Ubuntu 10.04 works best)
- Install Dependencies
- See source.android.com for more info

```
repo init -u git://android.git.kernel.org/platform/manifest.git  
repo sync
```

Step Two: Edit the buildspec.mk

- Copy the file from build/buildspec.mk.default to buildspec.mk (in the root directory of your AOSP build)

- Edit the file and add the following:

```
DEBUG_MODULE_libwebcore:=true
```

```
DEBUG_MODULE_libxml2:=true
```

```
TARGET_CUSTOM_DEBUG_CFLAGS:=-O0 -mlong-calls
```

```
ADDITIONAL_BUILD_PROPERTIES += debug.db.uid=100000
```

- This allows WebKit to be built in DEBUG mode

Step Three: Edit the Android.mk in external/webkit

- This is VERY IMPORTANT
- Your AOSP build will break if this is not done, since libwebkit will be huge with debugger symbols
- Uncomment this line:

```
LOCAL_PRELINK_MODULE := false
```

Step Four: Build the Project, take a nap

- Run the following in the root directory:

```
source build/envsetup.sh
```

```
lunch full-eng
```

```
make -j2 # Note: The more cores you have, the faster this will run
```

- This will take a very long time on most systems

Step Five: Start the Emulator

- Set the ANDROID_PRODUCT_OUT variable

```
export ANDROID_PRODUCT_OUT="/home/(username)/aosp-official/out/product/generic/"
```

- Start the emulator

```
emulator
```

Step Six: Run a WebKit Application

- I like running PhoneGap
- I run PhoneGap in Eclipse because I'm a n00b who can't use Command-Line debugger tools for Java
- This is identical to the standard workflow for debugging a Java application
 - See developer.android.com for more info on how to do this
- To check out PhoneGap Android, use git and check this out:

```
git clone git://github.com/phonegap/phonegap-android.git
```

Step Seven: Hook up a Java Debugger to the Java Application

- This is the same thing as debugging a Java Application in Android
- Set your breakpoints in Eclipse
- Then instead of running your Android Application, Debug your Android Application
- See <http://developer.android.com> for more details on how to do this

Step Eight: Hook up a C Debugger to WebKit

- Run the following command on your computer:

```
adb forward tcp:5039 tcp:5039
```

- Run the following command on your running emulator:

```
gdbserver 10.0.2.2:5039 --attach pid
```

- Run the following commands in gdb/ddd:

```
set solib-absolute-prefix /home/(yourdir)/aosp-  
official/out/target/product/generic/symbols
```

```
set solib-search-path /home/(yourdir)/aosp-  
official/out/target/product/generic/symbols/system/lib
```

```
file /home/(yourdir)/aosp-  
official/out/target/product/generic/symbols/system/app_process
```

If you're successful, and you use old tools like DDD....

Applications Places System Thu 17 Mar, 5:27 PM browserj

DDD: /home/browserj/aosp-official/external/webkit/WebKit/android/jni/WebCoreFrameBridge.cpp

File Edit View Program Commands Status Source Data Help

(): /home/browserj/aosp-official/external/webkit/WebKit/android/jni/WebCoreFrameBridge.cpp:1188

[JNIENV *) OXITQIBS

9: script
(Disabled)

```
1180
1181 static jobject StringByEvaluatingJavaScriptFromString(JNIEnv *env, jobject obj, jstring script)
1182 {
1183 #ifdef ANDROID_INSTRUMENT
1184     TimeCounterAuto counter(TimeCounter::NativeCallbackTimeCounter);
1185 #endif
1186     WebCore::Frame* pFrame = GET_NATIVE_FRAME(env, obj);
1187     LOG_ASSERT(pFrame, "stringByEvaluatingJavaScriptFromString must take a valid frame pointer!");
1188
1189     WebCore::ScriptValue value =
1190     pFrame->script()->executeScript(to_string(env, script), true);
1191     WebCore::String result = WebCore::String();
1192     ScriptState* scriptState = mainWorldScriptState(pFrame);
1193     if (!value.getString(scriptState, result))
```

DDD

Run

Interrupt

Step StepI

Next NextI

Until Finish

Cont Kill

Up Down

Undo Redo

Edit Make

```
0x809af5ba <StringByEvaluatingJavaScriptFromString+54>: ldr    r0, [sp, #20]
0x809af5bc <StringByEvaluatingJavaScriptFromString+56>: bl     0x8089f4bc <void>
WTF::dereferIfNotNull<WebCore::StringImpl>(WebCore::StringImpl*)>
0x809af5c0 <StringByEvaluatingJavaScriptFromString+60>: movs   r3, #0
```

(gdb) graph display pFrame
[No symbol "pFrame" in current context.]
(gdb) graph display pFrame
[No symbol "pFrame" in current context.]
(gdb)

[No symbol "pFrame" in current context.]

Inbox - bows... browserj@par... Commit Hist... [Buddy List] [colin@keyb... DDD: WebCo... Java - Phone... 5554:<build>

Obviously, most people won't bother to do this, but..

- This can give you a solid understanding of how WebKit works on the phone
- There are easier ways to do this, but showing a debug screen of WebKit impresses people
- This is how to debug WebKit, not V8 or any of the other libraries you can plug into WebKit
- This only works on a stable AOSP branch (Gingerbread), edge has WebKit built against proprietary Honeycomb source
- Even though I can do this, I don't really have any idea what is going on past this point
 - This could probably be made easier by using an Eclipse C/C++ debugger
 - I'd rather learn more gdb/ddd skills

Summary

- Web Development is easy
- Developing the fundamental tools required for the Web Development Ecosystem to exist is HARD
- Haters need to stop hating! Every time you get frustrated with WebKit bugs, attempt this process
- Android WebKit has different versions, but phone manufacturers know this and often provide emulators of the builds that they ship
- It's not always WebKit's fault on Android (in fact, it is very rarely WebKit's fault, it could be JSC, or Android Browser or the other components of WebKit)
- Keep Calm and Carry On

Questions?