Google's Opening Statement
Key Points of Evidence

- Sun gave the Java language to the public
- Google built Android using free and open technologies
- Sun publicly approved Android's use of Java
- Google made fair use of the Java language APIs in Android
Key Points of Evidence

- Sun gave the Java language to the public
- Google built Android using free and open technologies
- Sun publicly approved Android's use of Java
- Google made fair use of the Java language APIs in Android
The Java programming language is open and free for anyone to use.
"Q. And you understand that nobody owns the Java programming language; right?

A. That's correct.

Q. Anybody can use that without any royalty at all?

A. Correct."
The names of the various items appearing in the disputed API package specifications, such as names of API files, packages, classes, and methods, are not protected.
### Java API Package Names Are Free for All to Use

<table>
<thead>
<tr>
<th>java.lang.reflect</th>
<th>java.security.spec</th>
<th>javax.crypto.interfaces</th>
</tr>
</thead>
<tbody>
<tr>
<td>java.net</td>
<td>java.sql</td>
<td>javax.crypto.spec</td>
</tr>
<tr>
<td>java.nio</td>
<td>java.text</td>
<td>javax.net</td>
</tr>
<tr>
<td>java.nio.channels</td>
<td>java.util</td>
<td>javax.net.ssl</td>
</tr>
<tr>
<td>java.nio.channels.spi</td>
<td>java.util.jar</td>
<td>javax.security.auth</td>
</tr>
<tr>
<td>java.nio.charset</td>
<td>java.util.logging</td>
<td>javax.security.auth.callback</td>
</tr>
<tr>
<td>java.nio.charset.spi</td>
<td>java.util.prefs</td>
<td>javax.security.auth.login</td>
</tr>
<tr>
<td>java.lang.ref</td>
<td>java.util.regex</td>
<td>javax.security.auth.x500</td>
</tr>
<tr>
<td>java.security</td>
<td>java.util.zip</td>
<td>javax.security.cert</td>
</tr>
<tr>
<td>java.security.acl</td>
<td>java.lang</td>
<td>javax.sql</td>
</tr>
<tr>
<td>java.security.cert</td>
<td>java.lang.annotation</td>
<td>java.awt.font</td>
</tr>
<tr>
<td>java.security.interfaces</td>
<td>javax.crypto</td>
<td>java.beans</td>
</tr>
<tr>
<td></td>
<td></td>
<td>java.io</td>
</tr>
</tbody>
</table>
Java API Class Names Are Free for All to Use

<table>
<thead>
<tr>
<th>Class Name</th>
<th>Class Name</th>
<th>Class Name</th>
<th>Class Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BufferedOutputStream</td>
<td>BufferedReader</td>
<td>BufferedWriter</td>
<td>ByteArrayInputStream</td>
</tr>
<tr>
<td>ByteArrayOutputStream</td>
<td>CharArrayReader</td>
<td>CharArrayWriter</td>
<td>DataInputStream</td>
</tr>
<tr>
<td>DataOutputStream</td>
<td>File</td>
<td>FileDescriptor</td>
<td>FileInputStream</td>
</tr>
<tr>
<td>FileOutputStream</td>
<td>FilePermission</td>
<td>FileReader</td>
<td>FileWriter</td>
</tr>
<tr>
<td>FilterInputStream</td>
<td>FilterOutputStream</td>
<td>FilterReader</td>
<td>FilterWriter</td>
</tr>
<tr>
<td>InputStream</td>
<td>InputStreamReader</td>
<td>LineNumberInputStream</td>
<td>LineNumberReader</td>
</tr>
<tr>
<td>ObjectInputStream</td>
<td>ObjectInputStream.getField</td>
<td>ObjectOutputStream</td>
<td>ObjectOutputStream.PutField</td>
</tr>
<tr>
<td>ObjectStreamClass</td>
<td>ObjectStreamField</td>
<td>OutputStream</td>
<td>OutputStreamWriter</td>
</tr>
<tr>
<td>PipedInputStream</td>
<td>PipedOutputStream</td>
<td>PipedReader</td>
<td>PipedWriter</td>
</tr>
<tr>
<td>PrintStream</td>
<td>PrintWriter</td>
<td>PushbackInputStream</td>
<td>PushbackReader</td>
</tr>
<tr>
<td>RandomAccessFile</td>
<td>Reader</td>
<td>SequenceInputStream</td>
<td>SerializablePermission</td>
</tr>
<tr>
<td>StreamTokenizer</td>
<td>StringBufferInputStream</td>
<td>InputStreamReader</td>
<td>StringReader</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>StringWriter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Writer</td>
</tr>
</tbody>
</table>
Java API Method Names Are Free for All to Use

<table>
<thead>
<tr>
<th>static double abs()</th>
<th>static float abs()</th>
<th>static int abs()</th>
<th>static long abs()</th>
</tr>
</thead>
<tbody>
<tr>
<td>static double acos()</td>
<td>static double asin()</td>
<td>static double atan()</td>
<td>static double atan2()</td>
</tr>
<tr>
<td>static double cbrt()</td>
<td>static double ceil()</td>
<td>static double cos()</td>
<td>static double cosh()</td>
</tr>
<tr>
<td>static double exp()</td>
<td>static double expm1()</td>
<td>static double floor()</td>
<td>static double hypot()</td>
</tr>
<tr>
<td>static double IEEERemainder()</td>
<td>static double log()</td>
<td>static double log10()</td>
<td>static double log1p()</td>
</tr>
<tr>
<td>static double max()</td>
<td>static float max()</td>
<td>static int max()</td>
<td>static long max()</td>
</tr>
<tr>
<td>static double min()</td>
<td>static float min()</td>
<td>static int min()</td>
<td>static long min()</td>
</tr>
<tr>
<td>static double pow()</td>
<td>static double random()</td>
<td>static double rint()</td>
<td>static long round()</td>
</tr>
<tr>
<td>static int round()</td>
<td>static double signum()</td>
<td>static float signum()</td>
<td>static double sin()</td>
</tr>
<tr>
<td>static double sinh()</td>
<td>static double sqrt()</td>
<td>static double tan()</td>
<td>static double tanh()</td>
</tr>
<tr>
<td>static double toDegrees()</td>
<td>static double toRadians()</td>
<td>static double ulp()</td>
<td>static float ulp()</td>
</tr>
</tbody>
</table>
Package java.lang

Provides classes that are fundamental to the design of the Java programming language.

http://download.oracle.com/javase/6/docs/api/java/lang/package-summary.html (as of September 26, 2011)
Volume 1, Core Packages, describes the libraries that are the foundation of the Java language. These libraries include java.lang, java.io, java.util, and java.net. These are the general-purpose libraries fundamental to every Java program.
"I mean, in a sense, these elements here are like parts of speech. So nouns, verbs, adjectives, and so on. And using those parts of speech, package names, class and interface names, relationships between them, the methods associated with each class, methods associated with interface . . . you form an API using those parts of speech in the same way I'd write a technical paper using, you know, nouns, verbs, objects, so on."

Mitchell Deposition., p. 121:1-10
"Q. Then why does Java have APIs?
A. Well, if there were no APIs, we would only have a language. You would be able to write basic computations that never did any IO, had any communication with the outside world or the underlying platform. You could write -- you know, you could do computations on numbers and strings and generate them, but you wouldn't be able to do anything with them."
GNU Classpath

Open source based on Java programming language, including Java language APIs

1998
Apache Harmony

Open source based on Java programming language, including Java language APIs

2005
"Jonathan Schwartz, CEO at Sun, said in a press conference, 'there is no reason that Apache cannot ship Harmony today....'"
IBM's Use of Apache Harmony

Apache Harmony

Lotus Notes, SameTime, Symphony, and Expeditor
Sun gave the Java language to the public

Google built Android using free and open technologies

Sun publicly approved Android's use of Java

Google made fair use of the Java language APIs in Android
Google Is a Leader in Technology Innovation

Google search

Google Maps

Gmail
Google Invests in Research & Development

- 13,000 Google employees in California
- 4 out of 10 employees are engineers in Research and Development
The Google Team

Eric Schmidt
Chairman of the Board

Andy Rubin
Senior Vice President of Mobile
Key Android Engineers

Bob Lee

Dan Bornstein
Android Supports

Handset Manufacturers

Developers

Applications

Android
The History of Android

July 2005: Google acquires Android

Mid-2005 to 2008: Google and OHA build Android

August 2005 to mid-2006: Google and Sun discuss partnership

2005  2006  2007  2008  2009  2010

Fall 2008: Google launches first Android phone

May 2008: Sun praises Android

Nov. 2007: Sun welcomes Android

Nov. 2007: Google releases Android Kit

October 2008: HTC releases HTC Dream

April 2009: Oracle purchases Sun

June 2009: Oracle endorses Android
Samsung releases Galaxy

November 2009: Motorola releases Droid

March 2010: Oracle solicits Android sales deal

August 2010: Lawsuit filed
Eric Schmidt wrote:

> Scott.. I'm in a product review and we are looking at a very interesting partnership proposal with Sun. Basically, Andy Rubin runs our mobile os/search engineering group; he is talking with Alan Brenner, VP Consumer & Mobile Systems Group of Sun.
> Google has engaged with Sun's Java team in an effort to form an alliance around our Open Handset Platform. It is an opportunity for our two companies to work together to define the de-facto standard software stack for mobile phones. Google has adopted a completely open source model as a way to solve some intricate distribution dependencies. It makes sense to me that Sun and Google should do this together; can you check and hopefully dedicate the resources necessary to move this ahead at an accelerated pace.
> I wanted to make sure you know I will do the same on my side.
> Anyway, this is very exciting and hopefully a good idea for both of us!
> Thanks and take care.. Eric
Android Phone
The Android Platform
The Android Platform
The Android Platform
The Android Platform
The Android Platform

Applications

Activity Manager
Window Manager
Content Providers
View System
PackageManager
Telephony Manager
ResourceManager
Location Manager
Notification Manager

Libraries

Surface Manager
OpenGL ES
Display Driver
Keypad Driver
WiFi Driver
Audio Drivers

Android Runtime

Binder (IPC) Driver
Power Management
The Android Platform
The Android Platform

Applications
- Home
- Contacts
- Phone
- Browser
- ...

Application Framework
- Activity Manager
- Window Manager
- Content Providers
- View System
- Notification Manager
- Package Manager
- Telephony Manager
- Resource Manager
- Location Manager

Libraries
- Surface Manager
- OpenGL ES
- SGL
- Media Framework
- Rich audio and video

home video
The Android Platform
The Android Platform

- **Applications**
  - Home
  - Contacts
  - Phone
  - Browser
  - ...

- **Application Framework**
  - Activity Manager
  - Window Manager
  - Content Providers
  - View System
  - Package Manager
  - Telephony Manager
  - Resource Manager
  - Location Manager
  - Notification Manager

- **Libraries**
  - Surface Manager
  - Media Framework
  - SQLite
  - OpenGL ES
  - FreeType
  - WebKit
  - SGL
  - SSL
  - libc

- **Android Runtime**
  - Core Libraries
  - Dalvik Virtual Machine

- **Linux Kernel**
  - Display Driver
  - Camera Driver
  - Flash Memory Driver
  - Binder (IPC) Driver
  - Keypad Driver
  - WiFi Driver
  - Audio Drivers
  - Power Management
The Android Platform

<table>
<thead>
<tr>
<th>APPLICATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
</tr>
<tr>
<td>Contacts</td>
</tr>
<tr>
<td>Phone</td>
</tr>
<tr>
<td>Browser</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>APPLICATION FRAMEWORK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity Manager</td>
</tr>
<tr>
<td>Window Manager</td>
</tr>
<tr>
<td>Content Providers</td>
</tr>
<tr>
<td>View System</td>
</tr>
<tr>
<td>Package Manager</td>
</tr>
<tr>
<td>Telephony Manager</td>
</tr>
<tr>
<td>Resource Manager</td>
</tr>
<tr>
<td>Location Manager</td>
</tr>
<tr>
<td>Notification Manager</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LIBRARIES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surface Manager</td>
</tr>
<tr>
<td>Media Framework</td>
</tr>
<tr>
<td>OpenGL</td>
</tr>
<tr>
<td>FreeType</td>
</tr>
<tr>
<td>WebKit</td>
</tr>
<tr>
<td>SQLite</td>
</tr>
<tr>
<td>SGL</td>
</tr>
<tr>
<td>SSL</td>
</tr>
<tr>
<td>libc</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ANDROID RUNTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core Libraries</td>
</tr>
<tr>
<td>Dalvik Virtual Machine</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LINUX KERNEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Driver</td>
</tr>
<tr>
<td>Camera Driver</td>
</tr>
<tr>
<td>Flash Memory Driver</td>
</tr>
<tr>
<td>Binder (IPC) Driver</td>
</tr>
<tr>
<td>Keypad Driver</td>
</tr>
<tr>
<td>WiFi Driver</td>
</tr>
<tr>
<td>Audio Drivers</td>
</tr>
<tr>
<td>Power Management</td>
</tr>
</tbody>
</table>
The APIs at Issue

37 Java API Packages
The APIs at Issue

"structure, selection, organization"
Applications

Application Developer
Application Developer
max()
An API Provides Access to Source Code

**API**

choose greater of two numbers

```java
max()
```

tells the computer what to do

```java
public static float max(float f1, float f2) {
    if (f1 > f2) {
        return f1;
    }
    if (f1 < f2) {
        return f2;
    }
    /* if either arg is NaN, return NaN */
    if (f1 != f2) {
        return Float.NaN;
    }
    /* max(+0.0,-0.0) == +0.0 */
    /* Float.floatToRawIntBits(0.0f) == 0*/
    if (Float.floatToRawIntBits(f1) != 0) {
        return f2;
    }
    return 0.0f;
}
```
choose greater of two numbers

max() tells the computer what to do

```java
public static float max(float a, float b)
{
    // this check for NaN, from JLS 15.21.1, saves a method call
    if (a != a)
        return a;
    // no need to check if b is NaN; > will work correctly
    // recall that -0.0 == 0.0, but [+]0.0 - [+]0.0 behaves special
    if (a == 0 && b == 0)
        return a - b;
    return (a > b) ? a : b;
}
```

GNU Classpath 0.9.8
An API Provides Access to Source Code

**API**

choose greater of two numbers

\[ \text{max}() \]

tells the computer what to do

```java
public static float max(float a, float b) {
    if (a != a) return a;  // a is NaN
    if ((a == 0.0f) && (b == 0.0f)
        && (Float.floatToIntBits(a) ==
            negativeZeroFloatBits)) {
        return b;
    }
    return (a >= b) ? a : b;
}
```

**OpenJDK 6**
public static float max(float a, float b) {
    if (a != a) return a;  // a is NaN
    if ((a == 0.0f) && (b == 0.0f)
        && (Float.floatToIntBits(a) ==
            negativeZeroFloatBits)) {
        return b;
    }
    return (a >= b) ? a : b;
}
Java Language API: A System of Organization

java.lang.Math.max()
Software Developer Finds Source Code in API Packages

java.lang.Math.max()
APIs are Organized in Packages

java.lang.Math.max()
APIs are Organized in Packages

```java
java.lang.Math.max()

java.io.FileInputStream.read()
```
APIs are Organized in Packages

\texttt{java.lang.Math.max()}
Android Source Code/Implementations Differ from Java

Oracle JDK 1.5
String.compareTo

```
public int compareTo(String anotherString) {
    int len1 = count;
    int len2 = anotherString.count;
    int n = Math.min(len1, len2);
    char v1[] = value;
    char v2[] = anotherString.value;
    int i = offset;
    int j = anotherString.offset;
    if (i == j) {
        int k = 1;
        int lim = n + 1;
        while (k < lim) {
            char c1 = v1[k];
            char c2 = v2[k];
            if (c1 == c2) {
                return c1 - c2;
            }
            k++;
        }
    } else {
        while (i < 0) {
            char c1 = v1[i++];
            char c2 = v2[i++];
            if (c1 == c2) {
                return c1 - c2;
            }
        }
    }
    return len1 - len2;
}
```

Android
String.compareTo

```
public int compareTo(String anotherString) {
    // Code adapted from K&R, pg 101
    int o1 = offset, o2 = string.offset,
    result;
    int end = offset + count < anotherString.count ?
    count + string.count;
    char[] target = string.value;
    while (o1 < end) {
        if (result = value[o1++] - target[o2++])
            i = 0;
            return result;
    }
    return count - anotherString.count;
```
Android Source Code/Implementations Differ from Java

Oracle JDK 1.5
ZipFile.getInputStream
(fragment)

public InputStream getInputStream(ZipEntry entry) throws IOException {
    return getInputStream(entry.name);
}

private InputStream getInputStream(String name) throws IOException {
    if (name == null) {
        throw new
            NullPointerException("name");
    } long jentry = 0;
    ZipFileInputStream in = null;
    synchronized (this) {
        ensureOpen();
        jentry = getEntry(jzfile, name, false);
        if (jentry == 0) {
            return null;
        } if (mappedBuffer != null) {
            in = new
                MappedZipFileInputStream(jentry, name);
        } else {
            in = new
                ZipFileInputStream(jentry);
    }

    ...

Android
ZipFile.getInputStream
(fragment)

public InputStream getInputStream(ZipEntry entry) throws IOException {
    entry = getEntry(entry.getName());
    if (entry == null) {
        return null;
    }

    ...

TX 2392
Rebuttal Expert Report of Dr. Owen Astrachan, ¶30
rangeCheck(): 9 Lines of Code

9 lines of code

924 lines of code in the file

15 million lines of code in Android
rangeCheck(): 9 Lines of Code

15 million lines of code in Android
Iain Cockburn
Oracle Expert Witness

$0
Key Points of Evidence

- Sun gave the Java language to the public
- Google built Android using free and open technologies
- Sun publicly approved Android's use of Java
- Google made fair use of the Java language APIs in Android
MONDAY NOV 05, 2007

Congratulations Google, Red Hat and the Java Community!

I just wanted to add my voice to the chorus of others from Sun in offering my heartfelt congratulations to Google on the announcement of their new Java/Linux phone platform, Android. Congratulations!

I'd also like Sun to be the first platform software company to commit to a complete developer environment around the platform, as we throw Sun's NetBeans developer platform for mobile devices behind the effort. We've obviously done a ton of work to support developers on all Java based platforms, and were pleased to add Google's Android to the list.

Jonathan Schwartz
Sun's Chief Executive Officer

TX 2352
Late-2007: Sun Publicly Applauds Android

And needless to say, Google and the Open Handset Alliance just strapped another set of rockets to the community's momentum - and to the vision defining opportunity across our (and other) planets.

And in the spirit of offering congratulations, I'd like to offer a similar shout out to our friends at Red Hat Linux - who today announced their support for the OpenJDK project. With friends like Google and Red Hat, it sure seems like the momentum behind Java's on the rise...

Jonathan Schwartz
Sun's Chief Executive Officer
Late-2007: Sun Offers its Support for Android

Jonathan Schwartz
Sun's Chief Executive Officer

-----Original Message-----
From: Jonathan.Schwartz@sun.com [mailto:Jonathan.Schwartz@sun.com]
On Behalf
Of Jonathan Schwartz
Sent: Friday, November 09, 2007 10:57 AM
To: Eric Schmidt
Subject: android


Let us know how we can help support your announcements next week -
we're
happy to do so.

Jonathan
"Sun, has done a fantastic job, uh, invent-inventing Java, uh, expanding Java, opening up Java, giving Java to the world. And we're going to do more of the same."

Larry Ellison
Oracle CEO
"I'm reading a lot, uh, in the newspaper about, uh, you know, uh, devices based on Java that are Android devices, which is very exciting. You know, we're flat uh, I guess everyone should be flattered. ...I think we can see lots and lots of Java devices, some coming from our friends at Google."

Larry Ellison
Oracle CEO
Key Points of Evidence

- Sun gave the Java language to the public
- Google built Android using free and open technologies
- Sun publicly approved Android's use of Java
- Google made fair use of the Java language APIs in Android
And Participate In The Java Community Process

The Java Community Process
Community Development of Java Technology Specifications

 manufactures logos:

- Fujitsu
- Nordstrom
- Oracle
- HP
- Red Hat
- Intel
- SAP
- Nortel
- American Express
- Bank of America
- Sony
- IBM
- Nokia
- Sony Ericsson
- Time Warner Cable
- eBay
- IBM
- Adobe
- VMware
- Symantec
- Panasonic
- Credit Suisse
- UBS
Sun's Failed Efforts to Build a Java Platform for Smartphones

TX 2200

TX 2199

TX 2969
April 2009: Enter Oracle

Larry Ellison
From: Lawrence Ellison <larry.ellison@oracle.com>
Sent: Friday, May 22, 2009 9:53 PM
To: Scott McNealy <scott.mcnealy@sun.com>
Cc: larry.ellison@oracle.com; safra.catz@oracle.com; charles.phillips@oracle.com
Subject: RE: [Fwd: Status of Apple JVM]

Scott,

I think the best way to increase Java revenue is to build a mobile phone applications suite on top of JavaFX which will make Apple's iPhone a direct competitor of ours. We would then license the Java Phone software to carriers like Verizon.
Problem 3: Very limited internal expertise to make smart decisions
"Q. What was the business proposition? Did you want Google to replace its virtual machine with a Java machine? Is that what you said?

A. The prime — yeah, we wanted — we wanted to demonstrate — we thought we could demonstrate that our JVM would boot faster, run faster, use less power, do all of those things, and we would then have a joint project with Google where we could save them money and deliver better technology, you know, if — and there would be one and only one Java."
Key Points of Evidence

- Sun gave the Java language to the public
- Google built Android using free and open technologies
- Sun publicly approved Android's use of Java
- Google made fair use of the Java language APIs in Android