Android 101

Mercedes Wyss
@itrjwyss

Community Leader
Devs+502 & JDuchess Chapter Guatemala

Ex-JUG Member
Guatemala Java Users Group (GuateJUG)

Chief Technology Officer (CTO) at Produactivity
Full Stack Developer

Auth0 Ambassador &
Oracle Developer Champion

Duke's Choice Award
2016 Winner
Java

AMBASSADOR

{ } ORACLE®
Developer
Champion

⟨Devs⟩ +502
Comunidad Desarrolladores en Tecnologías
< Google > en Guatemala

JDuchess
Guatemala

Occidente, julio 7
7:30 a 18:00
Universidad Mesoamericana,
(3ra calle 14-03 y 14-36
zona 3, Quetzaltenango)

¿Qué es Android?

Android

- Sistema Operativo Basado en el Kernel de Linux

- Es un sistema Linux Multi-usuario
  (Cada aplicación es un usuario diferente)

- Cada aplicación tiene su propio "Equipo Virtual"
  (Cada aplicación se ejecuta de forma independiente a las otras)

ORACLE
Developer
Champion

¿Qué lenguaje utilizamos para programar Android?

Ambassador

Apps Android no son Apps Java

Java App

Android App

Program logic

Java libraries (common)

Java
User Interface

Java-specific
libraries

Android
User Interface

Android-specific
libraries

Image from https://www.extremetech.com/

ORACLE
Developer
Champion
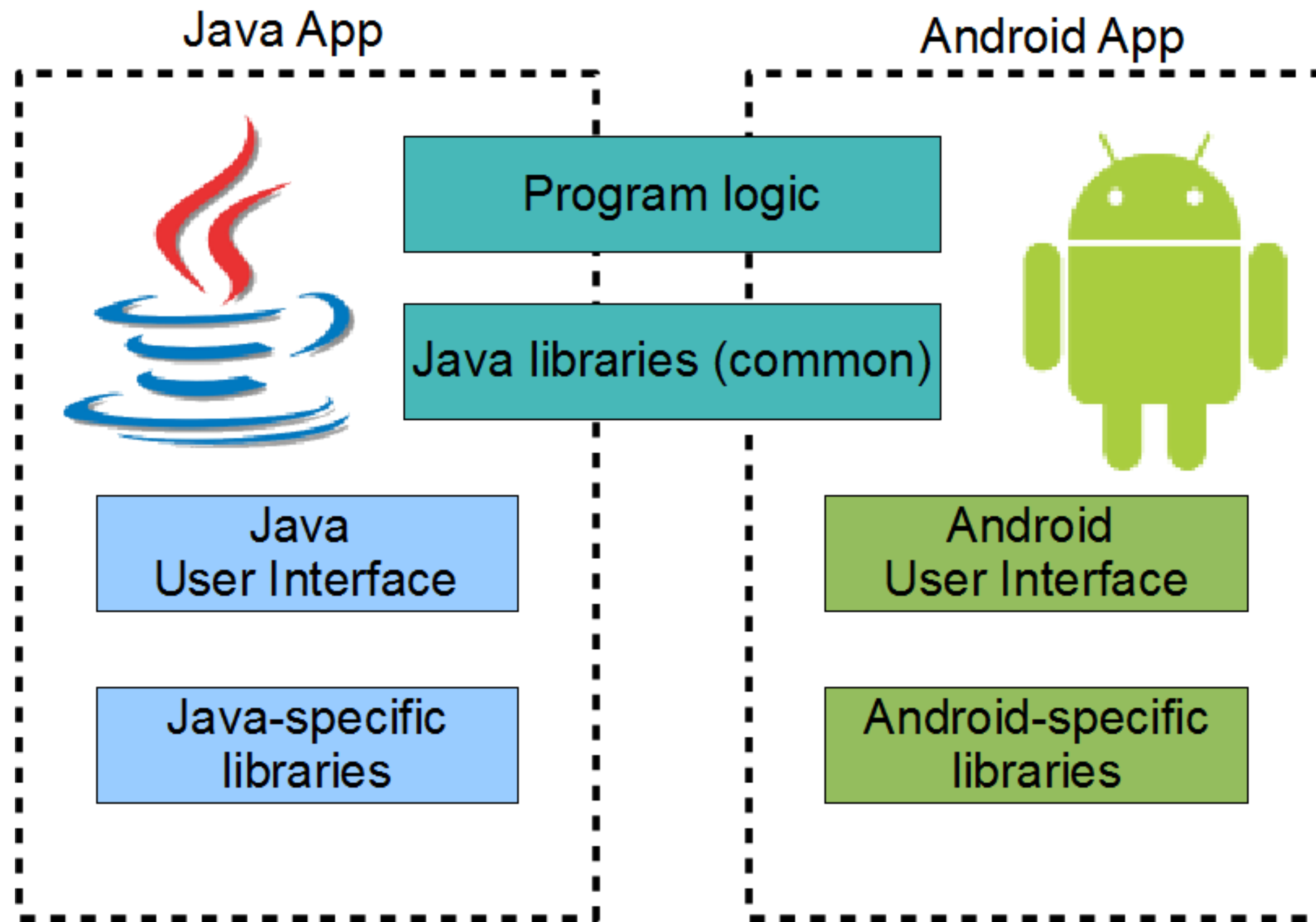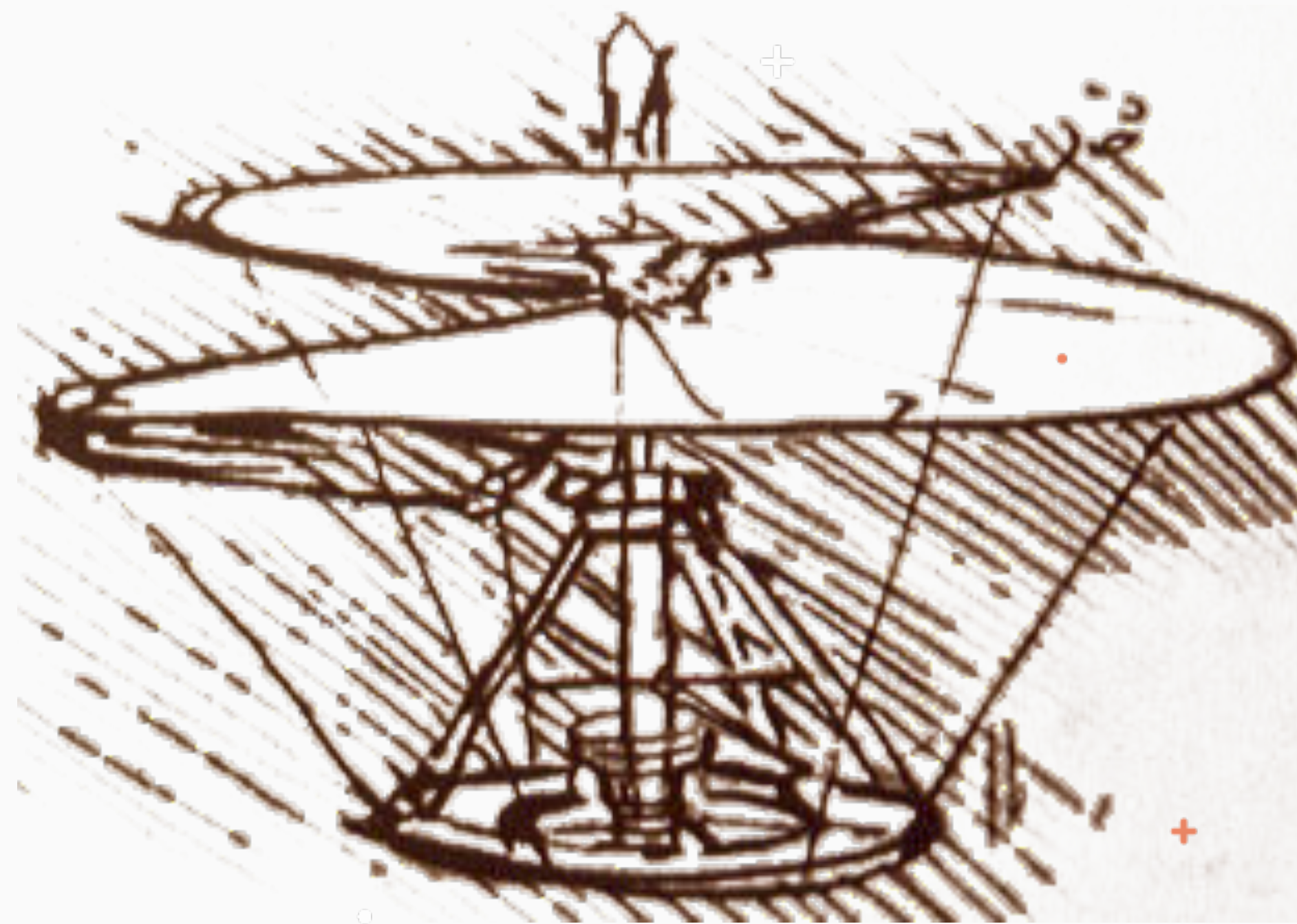
Dalvik & ART

No Standard Java Virtual Machines
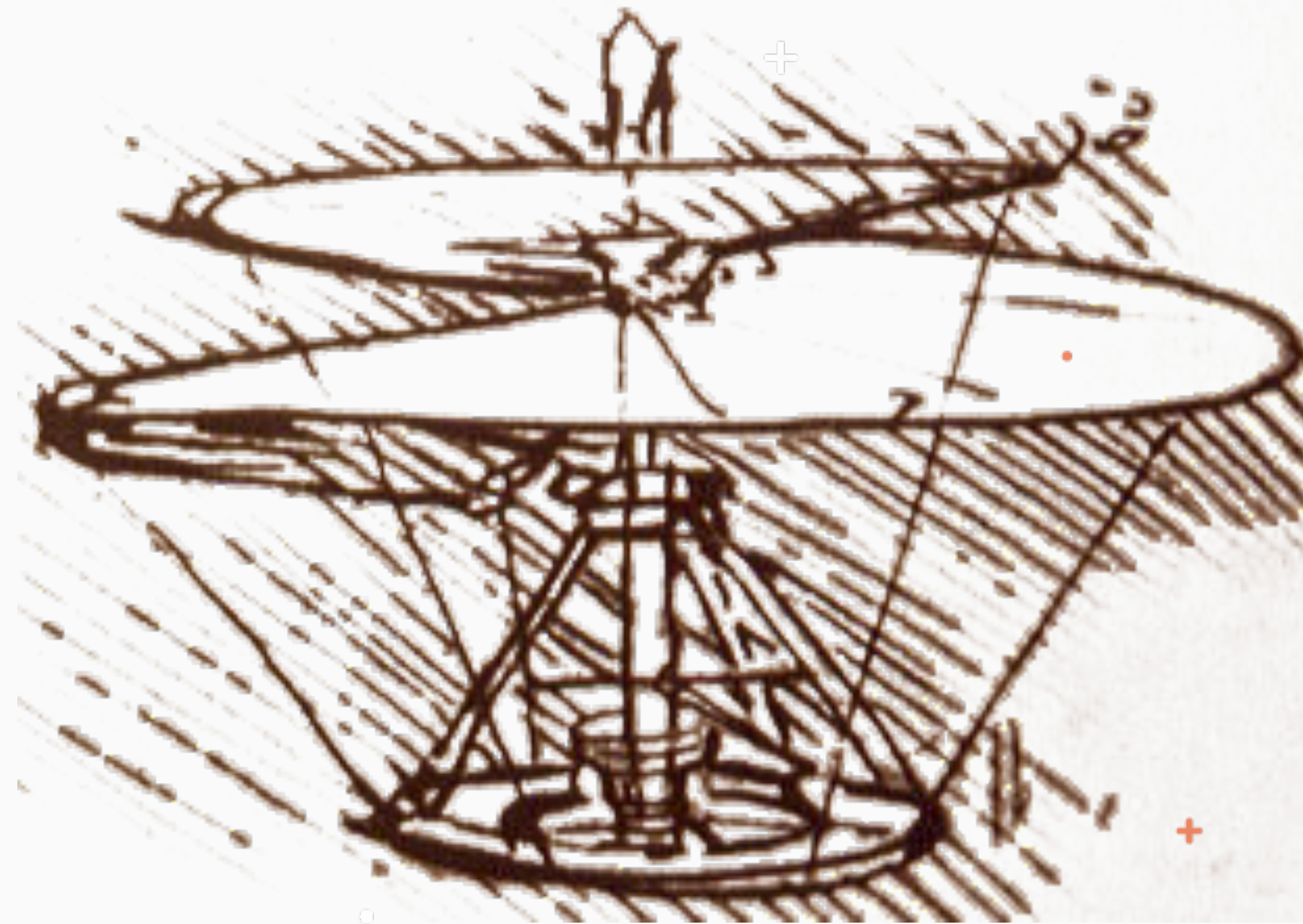
Dalvik & ART

we run .dx files and not .class ones

dx tool
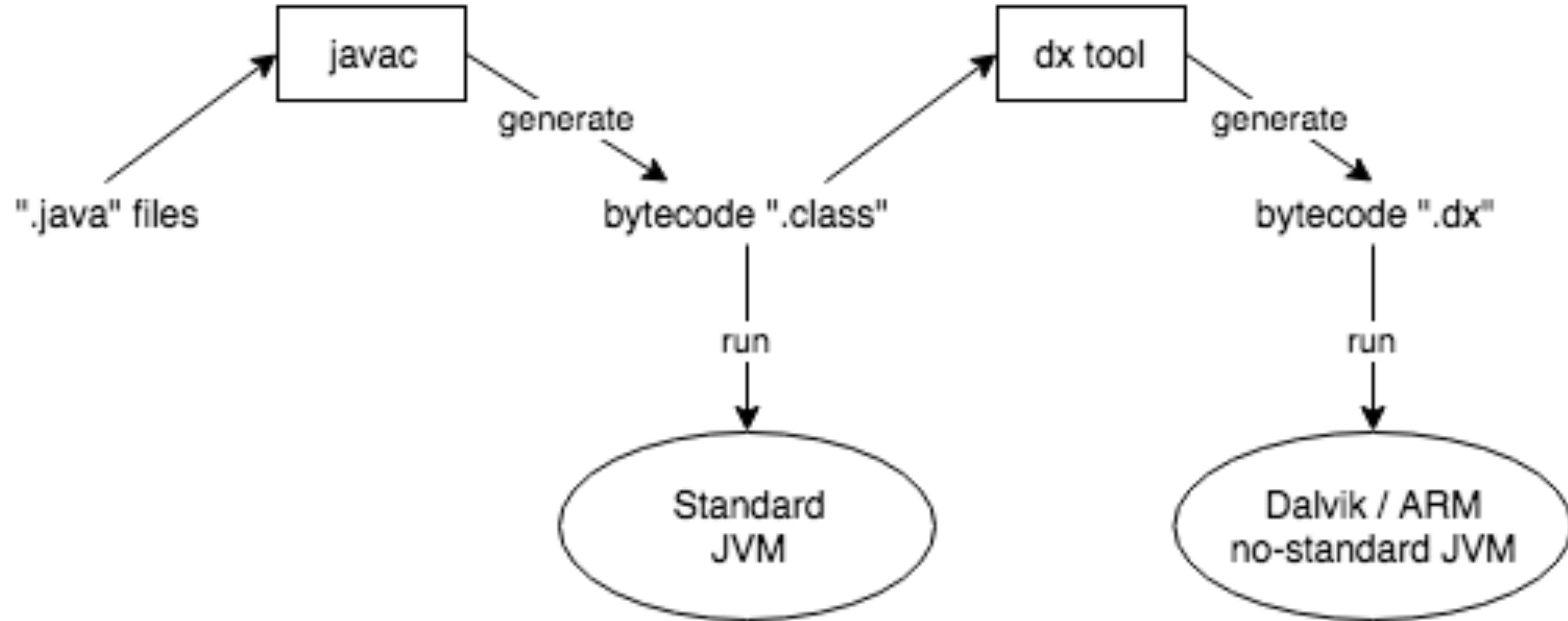
¿Qué <u>versión</u> de Java utiliza Android?

JavaSE6   JavaSE7   JavaSE8

ORACLE
Developer
Champion

¿Qué distribución de Java utiliza Android
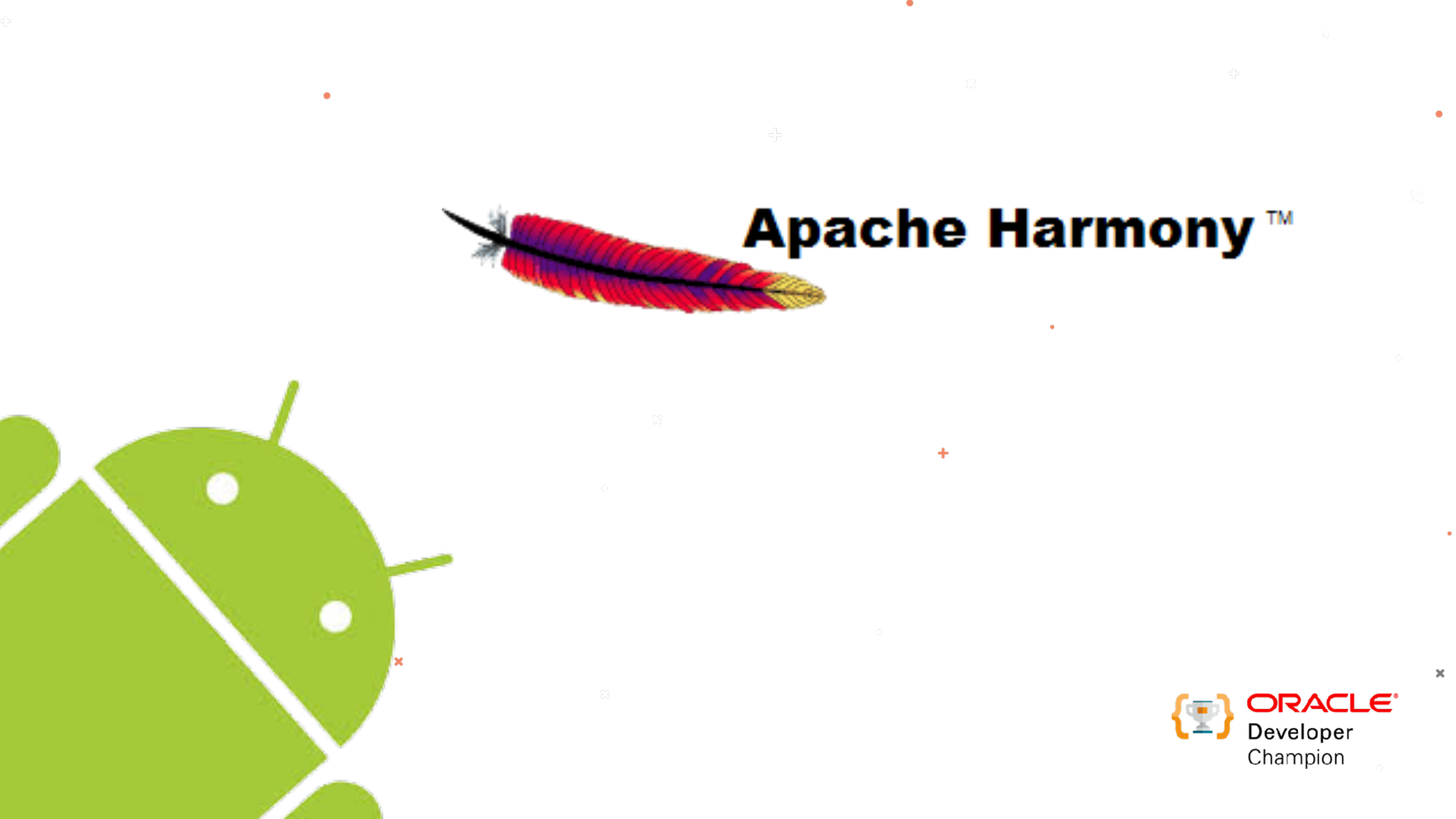
Apache Harmony ™

Apache Harmony™

- Open Source version of Java developed by the Apache Software Foundation

- 99% of compatibility with Java SE 5

- 97% of compatibility with Java SE 6

- The project was discontinued on November 2011

¿Cómo podemos utilizar JavaSE7?

ORACLE®
Developer
Champion

Magia Oscura
Google Transforma JavaSE7 en JavaSE8

¿Cómo logra Google hacer que Android sea Compatible con JavaSE8?

ANDROID

Alpha	Beta	Cupcake	Donut	Eclair	Froyo	Gingerbread	Honeycomb	Ice Cream Sandwich	Jelly Bean	KitKat
09/2008	02/2009	04/2009	09/2009	10/2009	05/2010	12/2010	02/2011	10/2011	08/2012	09/2013

@ijooh

Cupcake
Android 1.5

Donut
Android 1.6

Eclair
Android 2.0/2.1

Froyo
Android 2.2.x

Gingerbread
Android 2.3.x

Honeycomb
Android 3.x

Ice Cream Sandwich
Android 4.0.x

Jelly Bean
Android 4.1.x

KitKat
Android 4.4.x

Lollipop
Android 5.0

Marshmallow
android 6.0

Nougat
android 7.0

Los Features de JavaSE8 suelen ser compatibles solo con Android 7 - Nougat Lanzado en el 2016

Fragmentación

Una barrera entre la innovación y el desarrollo

Fundamentos de Programación en Android

App Fundamentals

- App Components

- Manifest File

- App Resources

App Components

- Activities

- Services

- Broadcast receivers

- Content providers

Manifest - Declaring Components

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest ... >
    <application android:icon="@drawable/app_icon.png" ... >
        <activity android:name="com.example.project.ExampleActivity"
                  android:label="@string/example_label" ... >
        </activity>
        ...
    </application>
</manifest>
```

Declaring Components Capabilities

```xml
<manifest ... >
    ...
    <application ... >
        <activity android:name="com.example.project.ComposeEmailActivity">
            <intent-filter>
                <action android:name="android.intent.action.SEND" />
                <data android:type="*/*" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

ORACLE
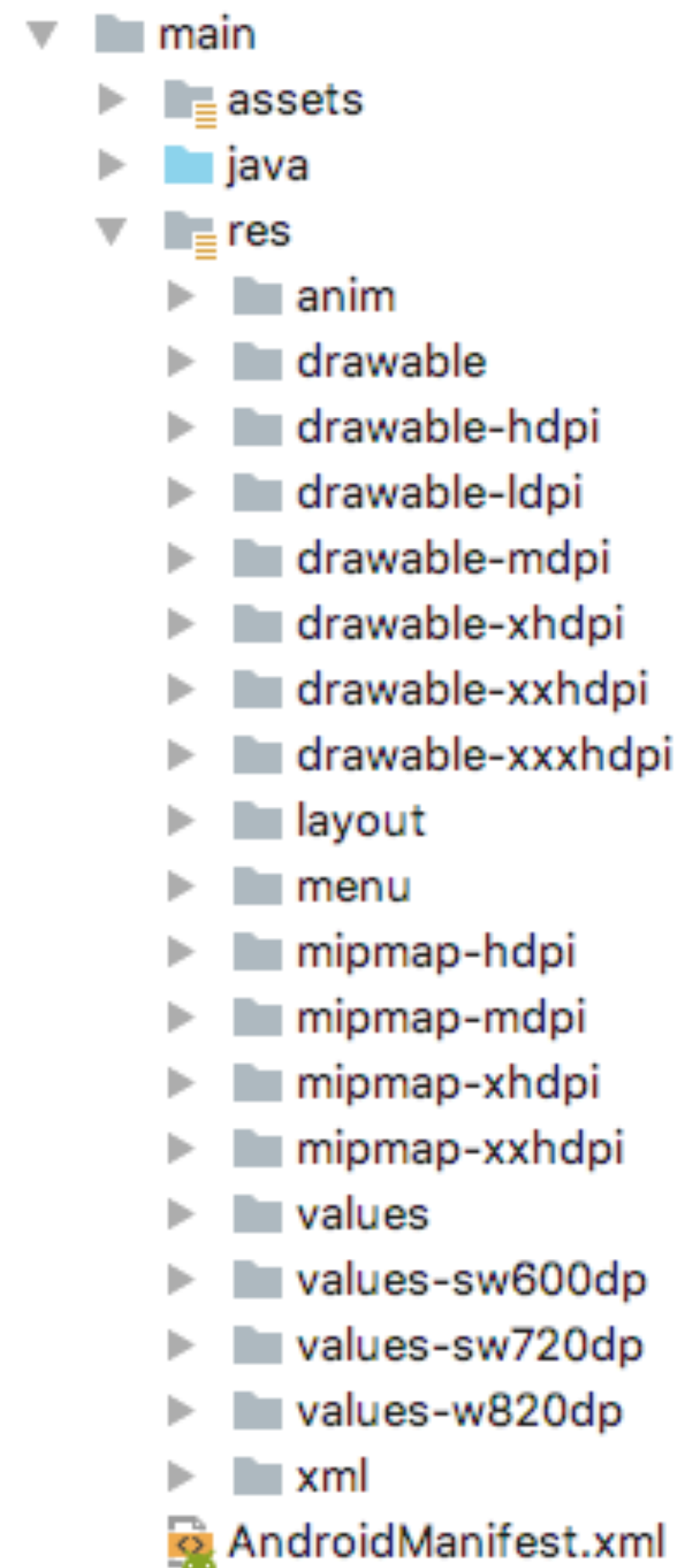Developer
Champion

Declaring App Requirements

```xml
<manifest ... >
    <uses-feature android:name="android.hardware.camera.any"
                  android:required="true" />
    <uses-sdk android:minSdkVersion="7" android:targetSdkVersion="19" />
    ...
</manifest>
```

App Resources

Lambda Calculus

This is a Function

$$squareSum(x, y) = x^2 + y^2$$

This is a Lambda

$$(x, y) \longrightarrow x^2 + y^2$$

ORACLE®
Developer
Champion

Lambda Expression Structure

Parameter(s) —> Body

Lambda Calculus

a —> a.doSomething()

a —> { a.doSomething(); }

Lambda Expression Structure

Always the **arrow**

a —> a.doSomething()

(type a) —> { a.doSomething(); }

Lambda Calculus

Parameter **name**

a —> a.doSomething()

(type a) —> { a.doSomething(); }

ORACLE
Developer
Champion

Lambda Expression Structure

Optional parameter <u>type</u>

a —> a.doSomething()

(type a) —> { a.doSomething(); }

Lambda Calculus

a —> a.doSomething()

(type a) —> { a.doSomething(); }

Optional Parenthesis when have only **one** <u>parameter</u> **without** <u>type</u>
Required when have **two or more** <u>parameters</u> or **have** <u>type</u>

ORACLE®
Developer
Champion

Lambda Expression Structure

Single statement

a —> **a.doSomething()**

(type a) —> { a.doSomething(); }

Lambda Calculus

a —> a.doSomething()

(type a) —> { a.doSomething(); }

For more than one statement use **curly brackets** (a code block)

ORACLE®
Developer
Champion

Lambda Expression Rules

```java
(int a, int b) -> {  return a + b; }

(a, b) -> {  return a + b; }

(a, b) -> a + b;

() -> System.out.println("Hello World");

(String s) -> { System.out.println(s); }

() -> 42

() -> { return 3.1415 };
```

Lambda Expressions in Action

```java
mHandler.postDelayed(new Runnable() {
    @Override
    public void run() {
        goToLogin();
    }
}, 5000);
```

```java
private Runnable stopAndGoToLogin = new Runnable() {
    @Override
    public void run() {
        goToLogin();
    }
}

mHandler.postDelayed(stopAndGoToLogin, 5000);
```

```java
private Runnable stopAndGoToLogin = new Runnable() {
    @Override
    public void run() {
        goToLogin();
    }
}

private Runnable stopAndGoToLogin = () -> goToLogin();

mHandler.postDelayed(stopAndGoToLogin, 5000);
```

```java
private Runnable stopAndGoToLogin = new Runnable() {
    @Override
    public void run() {
        goToLogin();
    }
}

private Runnable stopAndGoToLogin = () -> goToLogin();

mHandler.postDelayed(() -> goToLogin(), 5000);
```

ORACLE
Developer
Champion

we can pass to have a crazy(verbosity) code to other more simple
verbosity —> simple

```java
mHandler.postDelayed(new Runnable() {
    @Override
    public void run() {
        goToLogin();
    }
}, 5000);



mHandler.postDelayed(() -> goToLogin(), 5000);
```

Understanding Lambda Expressions

How we can define a Runnable?

Understanding Lambda Expressions

Runnable is an Interface

Functional Interface

Is any <u>interface</u> that contains only **one** **abstract method**.

```
java.lang.Runnable

void run()
```

Basic Functional Interface

```java
@FunctionalInterface
public interface Check<T> {

    boolean test(T a);

}
```

Example

```java
public class Animal {

    private String specie;
    private boolean canHop;
    private boolean canSwim;

    public Animal(String specieName, boolean hopper, boolean swimmer){
        specie = specieName;
        canHop = hopper;
        canSwim = swimmer;
    }

    public boolean canHop(){
        return canHop;
    }

    public boolean canSwim(){
        return canSwim;
    }

    public String toString(){
        return specie;
    }

}
```

ORACLE®
Developer
Champion

Example

```java
Check<Animal> checkHop = a -> a.canHop();
print(animals, checkHop);

Check<Animal> checkSwim = a -> a.canSwim();
print(animals, checkSwim);



private static void print(List<Animal> animals, Check checker) {
    for (Animal animal : animals) {
        if (checker.test(animal)) {
            System.out.print(animal + " ");
        }
    }
}
```

Android Fundamentals

- ~~Java language syntax~~ *Kotlin*
- ~~UI logic goes in the Activity~~ *View Models*
- ~~Services for background processing~~ *Job Scheduler*
- ~~Monitor system events with Receivers~~ *Job Scheduler*
- ~~Content Providers for database access~~ *Room database*

WORK MANAGER

#io18

Android Jetpack

Android Jetpack

Foundation

Architecture

Behavior

UI

ORACLE
Developer
Champion

https://github.com/itrjwyss/Journey18

https://www.facebook.com/itrjwyss

@itrjwyss