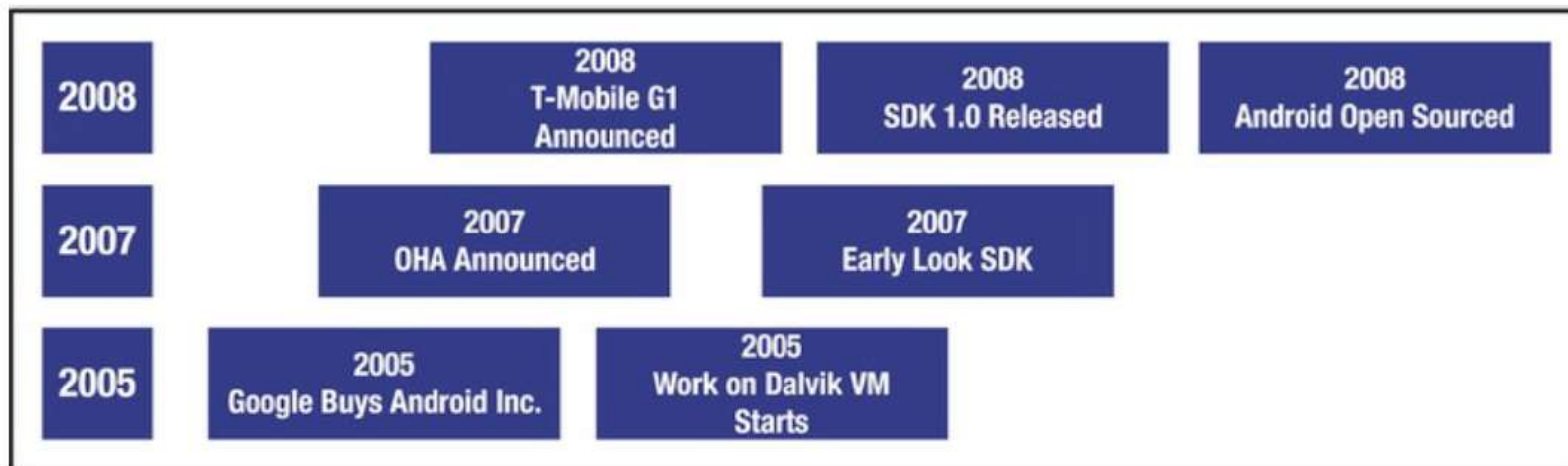


Programação para Dispositivos Móveis

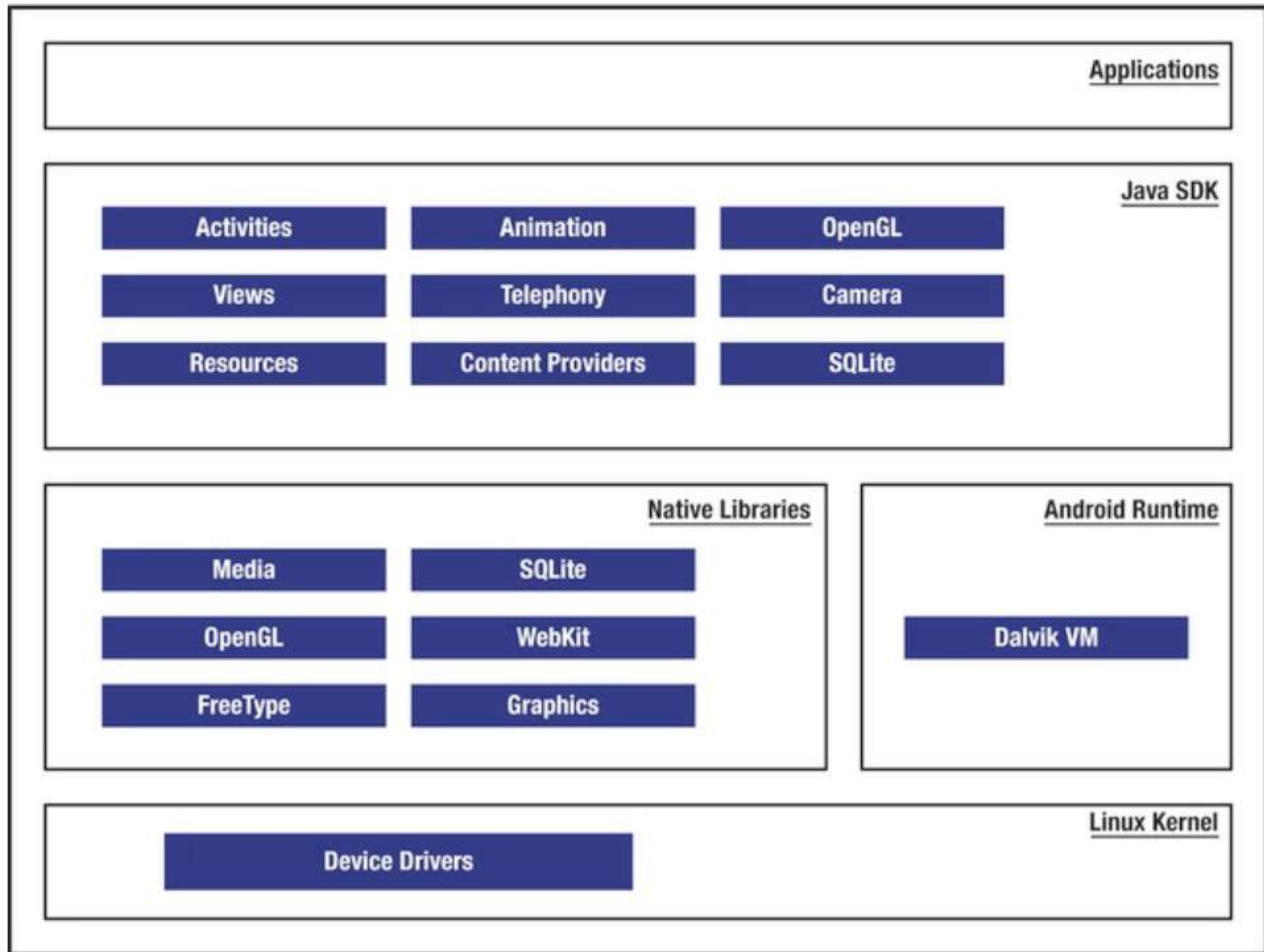
01 - Introdução



- 2005 Google adquire a Empresa Android Inc. e começa o desenvolvimento da plataforma Android



- Dalvik VM
 - Google gastou muito tempo e dinheiro em uma maneira de otimizar as aplicações para dispositivos móveis (pouca memória e processamento lento)
 - Dan Bornstein otimizou a JVM criando a Dalvik VM
 - Dalvik VM une várias classes java em um Dalvik Executable (*.dex)
 - O código executável final no Android é uma Dalvik VM e não bytecode
 - Você não pode executar diretamente bytecodes



- Consorcio de mais de 84 empresas com o intuito de desenvolver normas e padrões para dispositivos móveis
- Carro chefe é o Android

Open Handset Alliance(OHA)

Mobile Operators



open handset alliance

Handset Manufacturers



Semiconductor Companies



Software Companies



Commercialization Companies



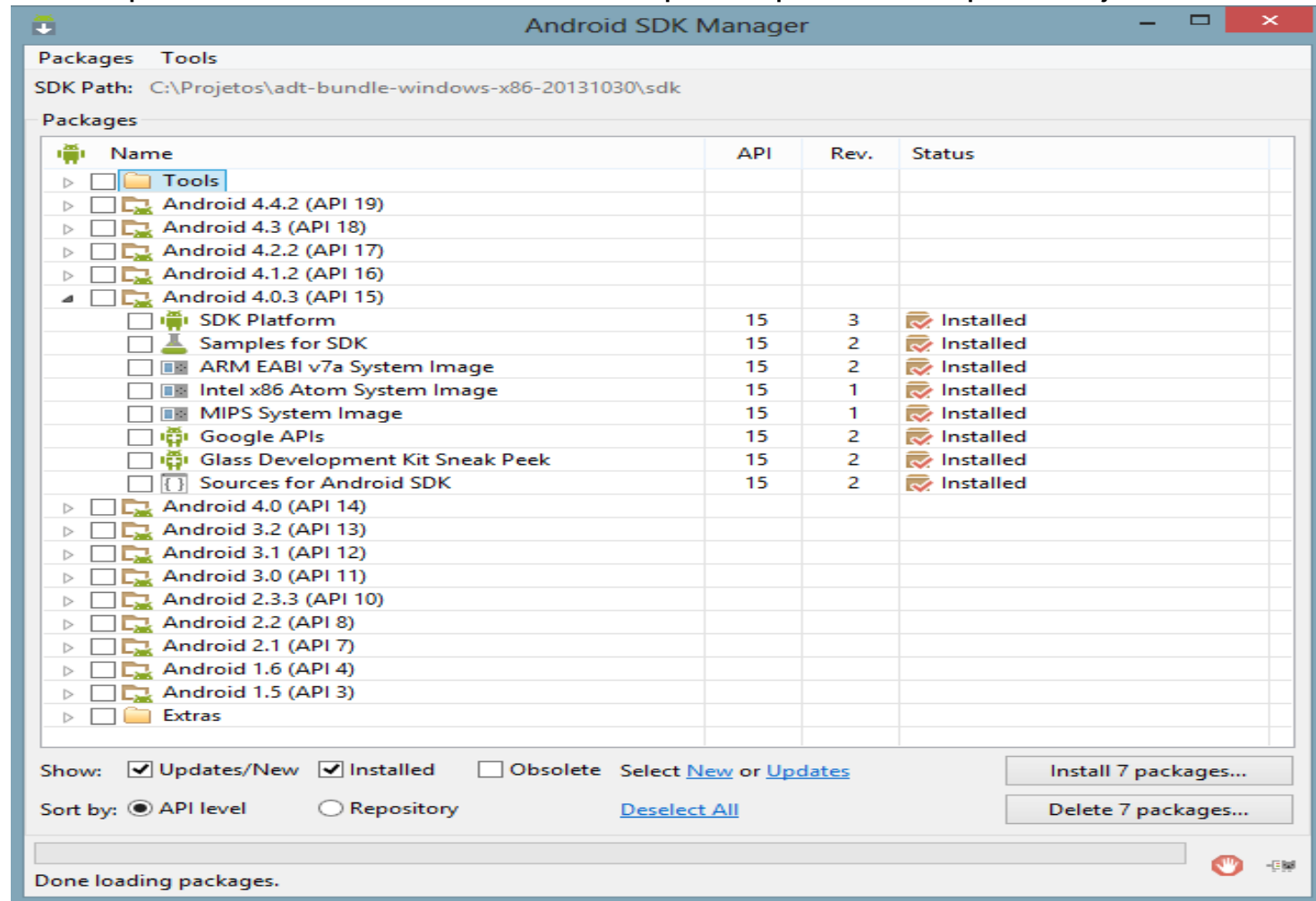
- 2010 temos cerca de 3 bilhões de celulares
- Até 2015 os acessos a internet por dispositivos móveis ultrapassaram os computadores pessoais
- O desenvolvimento é diferente, pois precisamos se preocupar com câmeras, microfone, tela sensível ao toque, sensores de ambiente, geolocalização, etc.
- Oportunidade de criar aplicações úteis ao dia a dia

Por que Android?

- Nenhuma certificação é exigida
- Google Play
- Não existe “processo de aprovação” das aplicações
- Total controle dos usuários

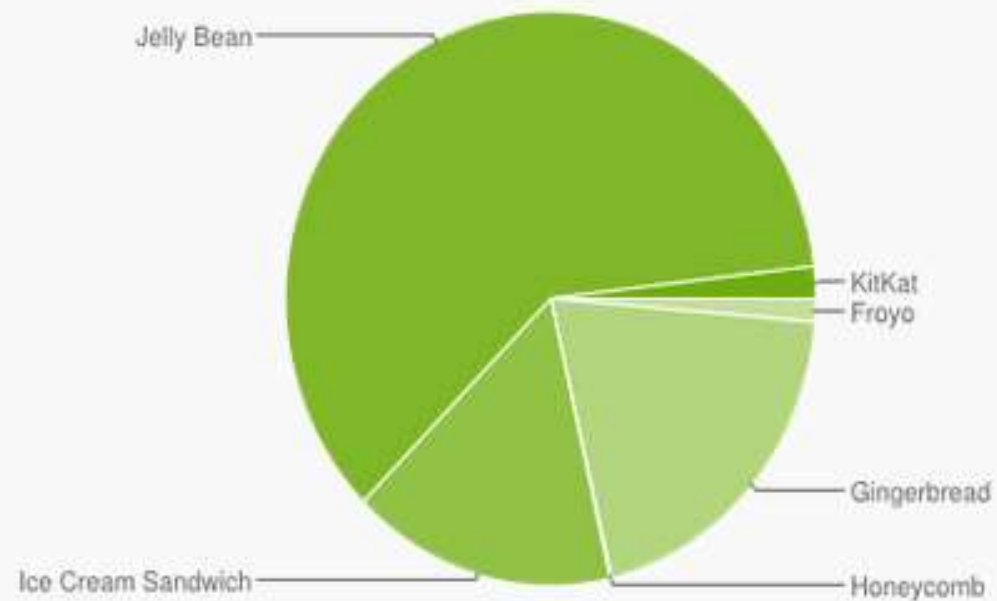
- Windows, Mac ou Linux
- JDK (Java Devel
- Download SDK (ADT Bundle for Windows)
 - (Plataforma completa)
 - Eclipse com ADT (Android Developer Toolkit)
 - Android SDK
 - Android API
 - Development tools
 - Android Virtual Device (AVD) usado pelo emulador
 - Documentação
 - Códigos Exemplos
 - Android Plataform-Tools
 - A última plataforma Android
 - Á última imagem do sistema Android do emulador
- Download via: <http://developer.android.com/sdk/index.html>
- Neste link também contam as instruções para instalação em outras IDEs existentes e outras plataformas

- Dentro do Eclipse
 - Menu Window – Android SDK Managment
- Escolher a plataforma e demais ferramentas para a plataforma que desejar desenvolver



<https://developer.android.com/about/dashboards/index.html>

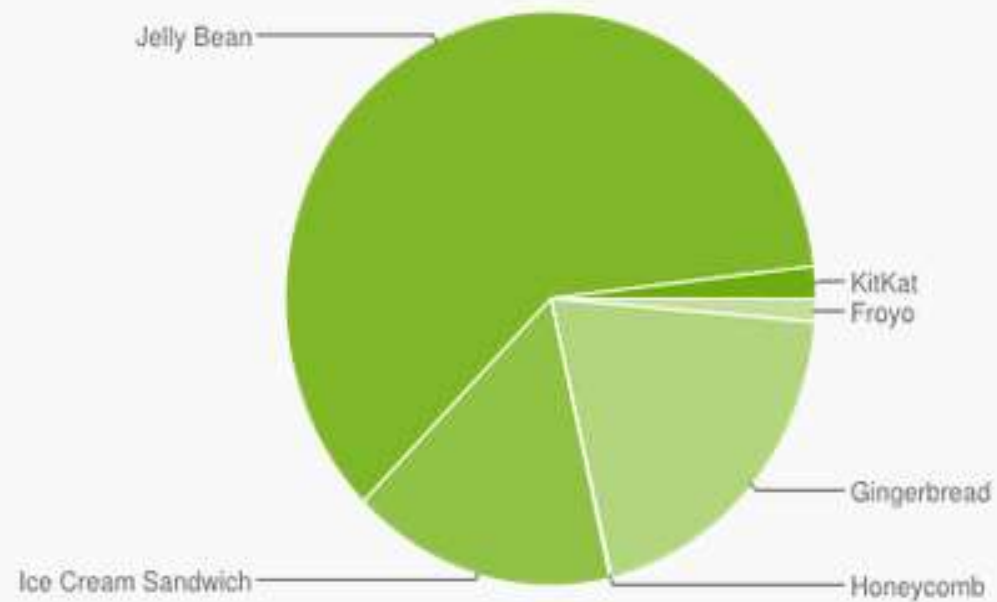
Version	Codename	API	Distribution
2.2	Froyo	8	1.3%
2.3.3 - 2.3.7	Gingerbread	10	20.0%
3.2	Honeycomb	13	0.1%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	16.1%
4.1.x	Jelly Bean	16	35.5%
4.2.x		17	16.3%
4.3		18	8.9%
4.4	KitKat	19	1.8%



*Data collected during a 7-day period ending on February 4, 2014.
Any versions with less than 0.1% distribution are not shown.*

<https://developer.android.com/about/dashboards/index.html>

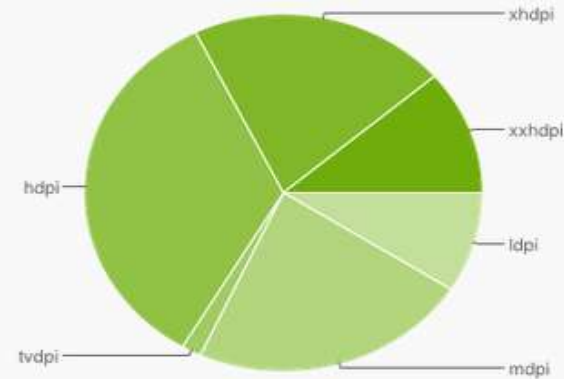
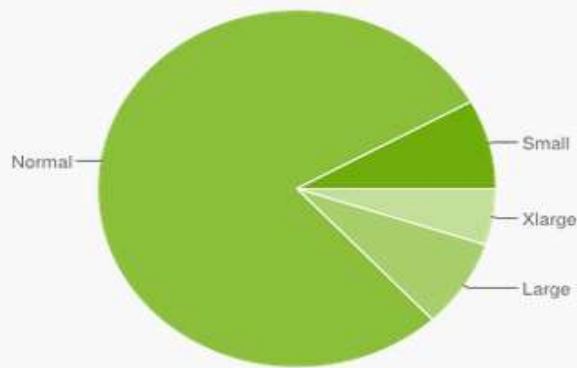
Version	Codename	API	Distribution
2.2	Froyo	8	1.3%
2.3.3 - 2.3.7	Gingerbread	10	20.0%
3.2	Honeycomb	13	0.1%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	16.1%
4.1.x	Jelly Bean	16	35.5%
4.2.x		17	16.3%
4.3		18	8.9%
4.4	KitKat	19	1.8%



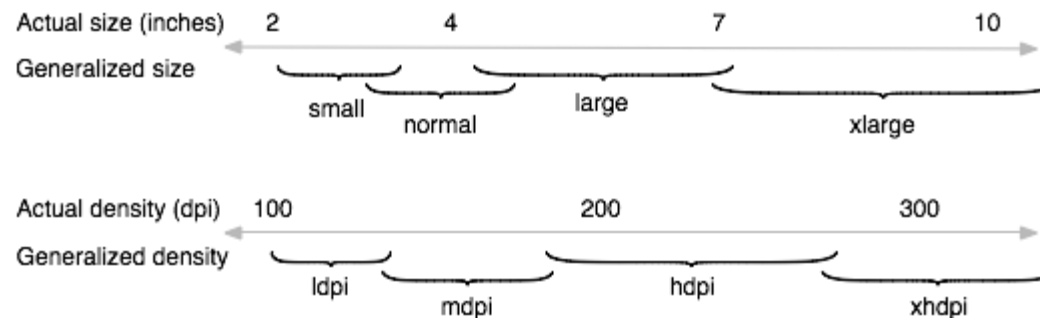
*Data collected during a 7-day period ending on February 4, 2014.
Any versions with less than 0.1% distribution are not shown.*

Tamanho da Tela e Densidade

	ldpi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
Small	8.1%						8.1%
Normal	0.1%	13.9%		33.3%	20.2%	11.3%	78.8%
Large	0.8%	4.4%	1.6%	0.6%	0.6%		8.0%
Xlarge	0.1%	4.5%		0.3%	0.2%		5.1%
Total	9.1%	22.8%	1.6%	34.2%	21.0%	11.3%	

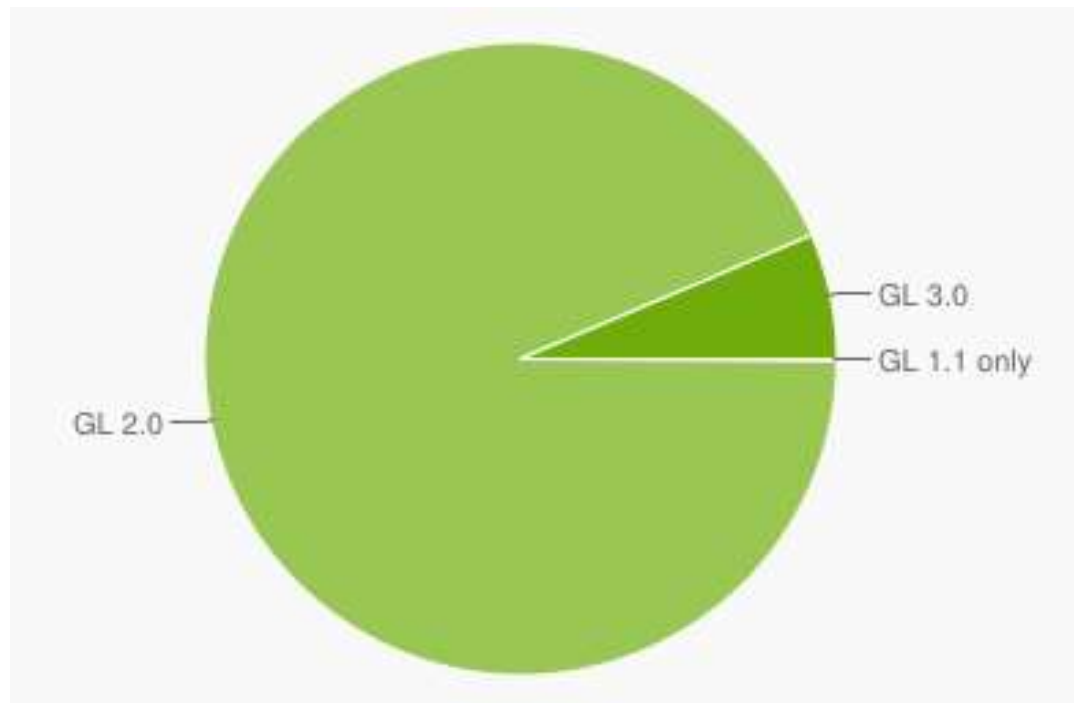


Data collected during a 7-day period ending on February 4, 2014.
Any screen configurations with less than 0.1% distribution are not shown.

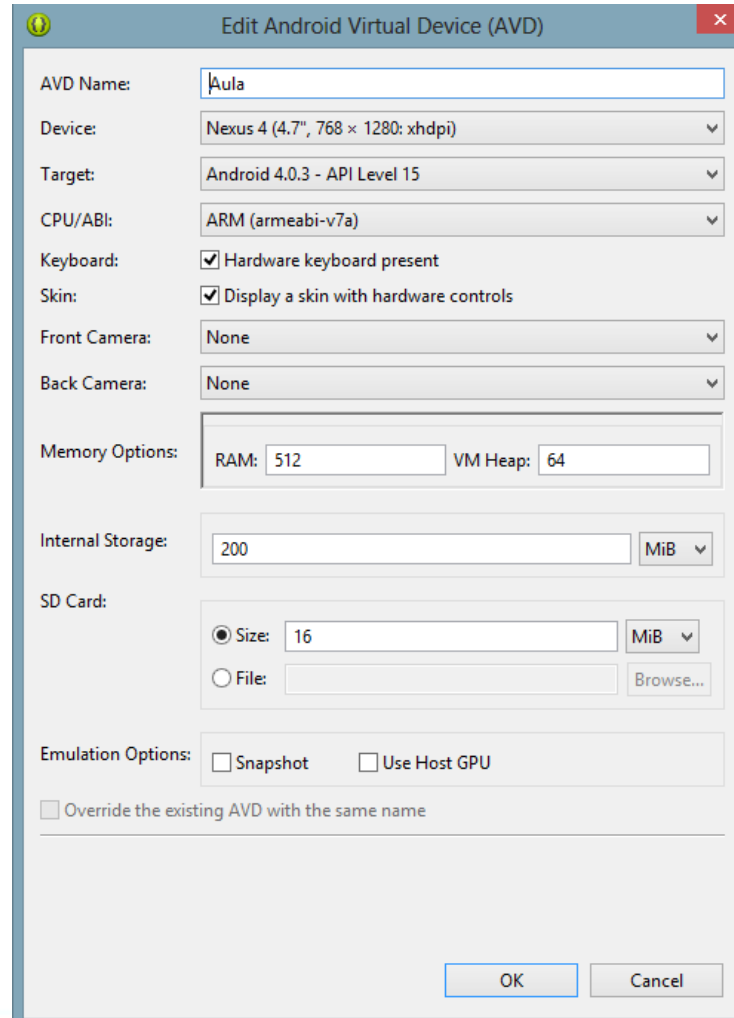


Versão do OpenGL

OpenGL ES Version	Distribution
1.1 only	0.1%
2.0	92.3%
3.0	7.6%



- Criação de um AVD (Android Virtual Device). Emulador para testar sua aplicação
- Clicar no menu Window – Android Virtual Device Manager (ou selecione o ícone na barra de ferramentas do Eclipse)
- Escolha a melhor configuração para testar sua aplicação



Dialog box titled "Edit Android Virtual Device (AVD)".

Fields and options:

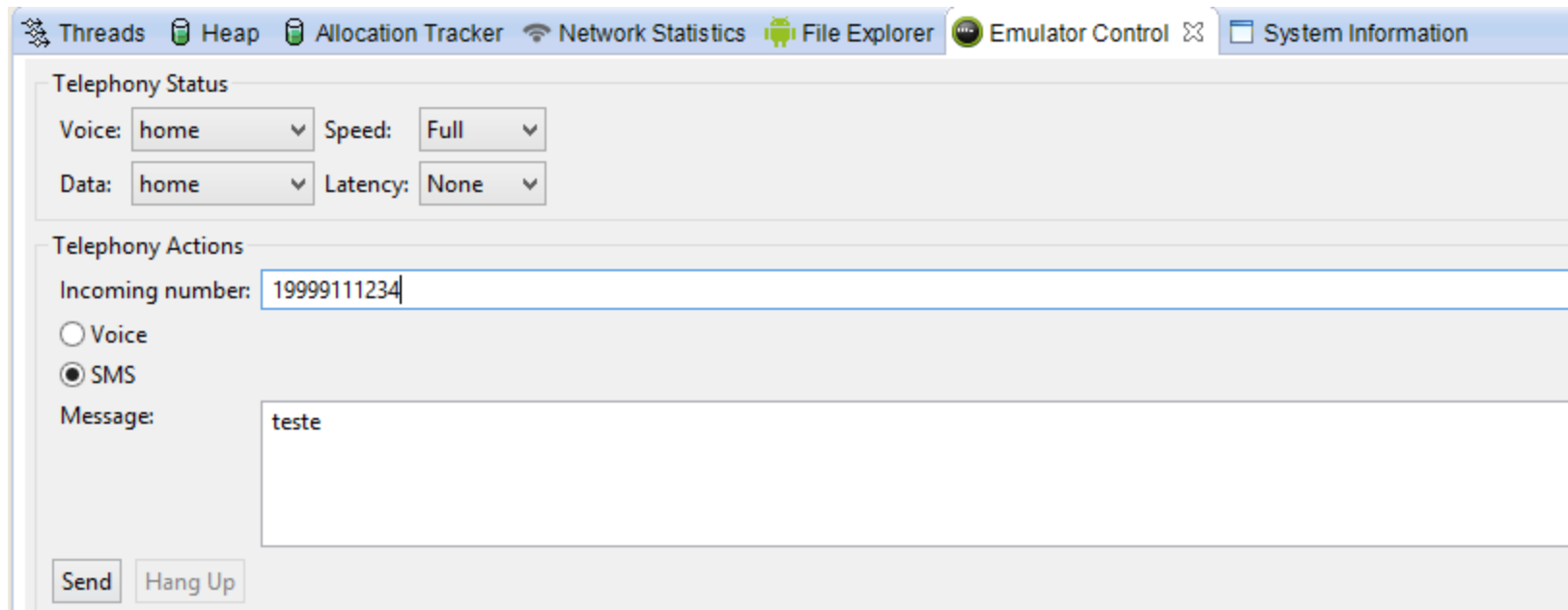
- AVD Name:
- Device:
- Target:
- CPU/ABI:
- Keyboard: ☒ Hardware keyboard present
- Skin: ☒ Display a skin with hardware controls
- Front Camera:
- Back Camera:
- Memory Options: RAM: VM Heap:
- Internal Storage:
- SD Card: ☒ Size: ☐ File:
- Emulation Options: ☐ Snapshot ☐ Use Host GPU
- ☐ Override the existing AVD with the same name

Buttons:

Dalvik Debug Monitor Server (DDMS)

Clique em Windows – Open Perspective – DDMS

- Além de debug, captura telas, simula recebimento de ligações e SMS, gerenciador de arquivos, analisador de tráfego na rede, entre outras.
- Exercício
- Abra o emulador e abra a perspectiva DDMS, clique na Aba Emulator Control e envie uma ligação e um SMS para o emulador.



- Android Debug Bridge (adb) é um aplicativo versátil de via linha de comando que possibilita a comunicação com uma instância do emulador ou dispositivo físico conectado.
- Ele se encontra no diretório do ADT (normalmente adt-bundle-Windows_x86\sdk\plataforma-tools)
- Alguns Exemplos:
 - adb devices // lista os dispositivos conectados
 - adb -s emulator-5556 install aplicacao.apk // instala a aplicação do emulador-5556
 - adb pull <remote> <local> // copia do emulador para o micro
 - adb push foo.txt /sdcard/foo.txt // copia do micro para o emulador
 - adb shell // entra no shell
 - adb shell ls // executa o comando "ls" no shell

DICA: Deixando o diretório sdcard com permissões RW (read e write) no emulador

```
$/adb shell
```

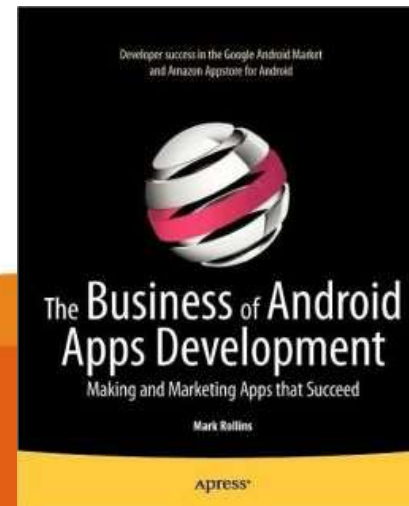
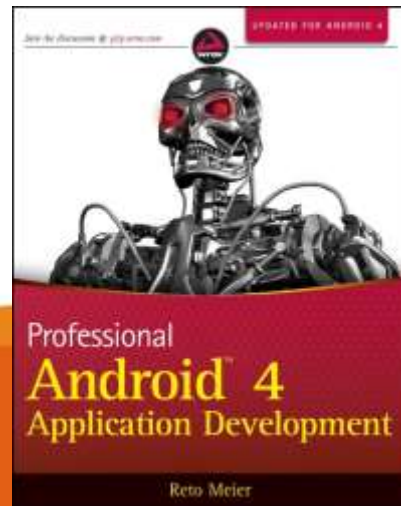
```
mount -o rw,remount rootfs /
```

```
chmod 777 /mnt/sdcard
```

Folha 01-Exercícios



Anhanguera



Lauren Dorsey
Shane Conder

