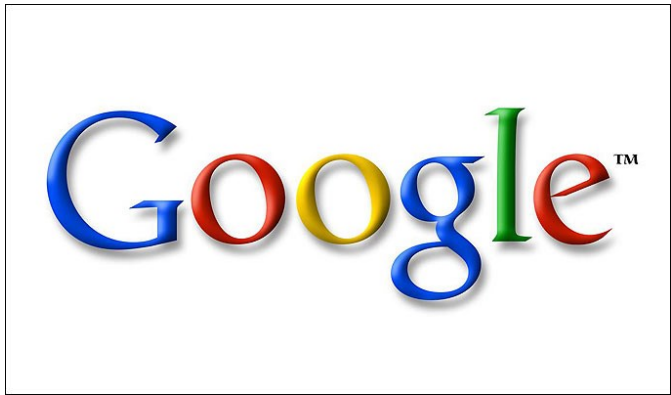
Introduction to Android Multimedia

CS 436 Software Development on Mobile

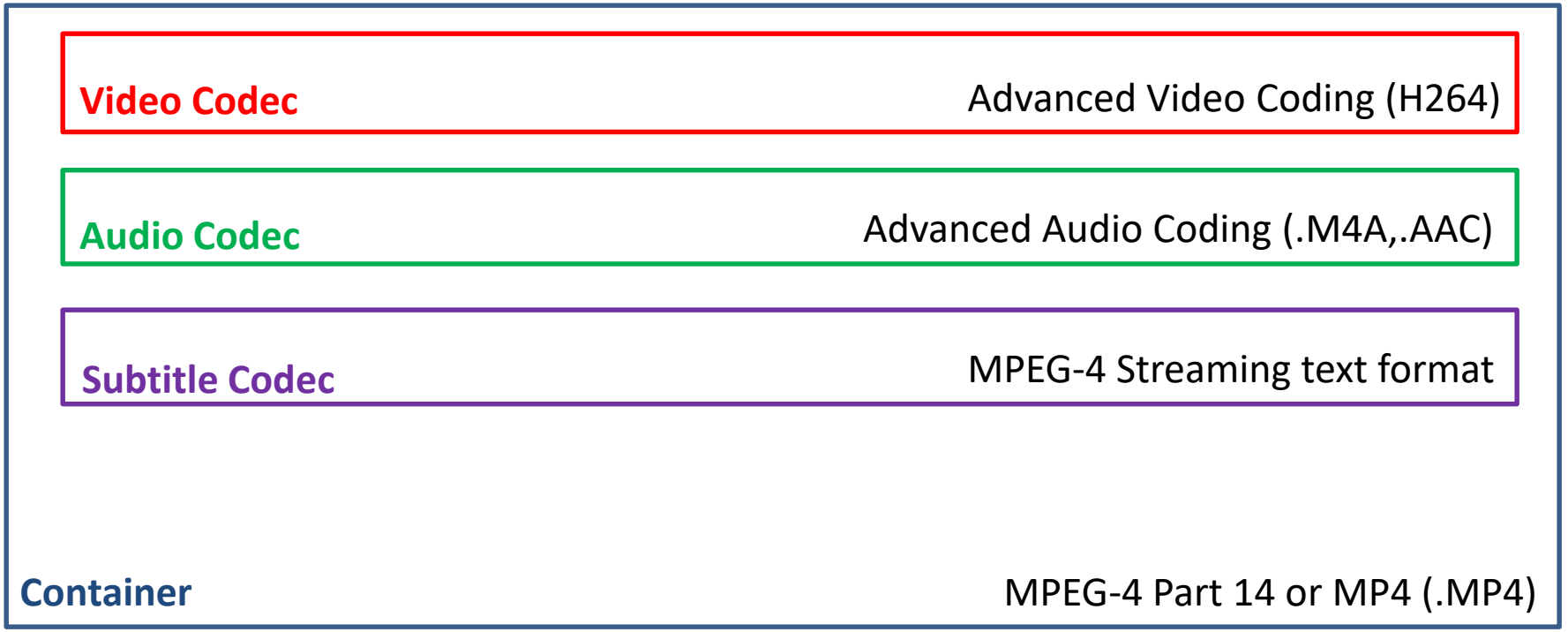
By Dr.Paween Khoenkaw



Introduction to Android Multimedia

Image
Sound and Video

Multimedia data



Android native audio codec support

Format / Codec		Encoder	Decoder	Container Formats
AAC LC		•	•	• 3GPP (.3gp) • MPEG-4 (.mp4, .m4a) • ADTS raw AAC (.aac, decode in Android 3.1+, encode in Android 4.0+, ADIF not supported) • MPEG-TS (.ts, not seekable, Android 3.0+)
HE-AACv1 (AAC+)		• (Android 4.1+)	•	
HE-AACv2 (enhanced AAC+)			•	
AAC ELD (enhanced low delay AAC)		• (Android 4.1+)	• (Android 4.1+)	
AMR-NB		•	•	3GPP (.3gp)
AMR-WB		•	•	3GPP (.3gp)
FLAC			• (Android 3.1+)	FLAC (.flac) only
MP3			•	MP3 (.mp3)
MIDI			•	• Type 0 and 1 (.mid, .xmf, .mxmf) • RTTTL/RTX (.rtttl, .rtx) • OTA (.ota) • iMelody (.imy)
Vorbis			•	• Ogg (.ogg) • Matroska (.mkv, Android 4.0+)
PCM/WAVE		• (Android 4.1+)	•	WAVE (.wav)
Opus		• (Android 5.0+)		Matroska (.mkv)

Android native image codec support

Format / Codec	Encoder	Decoder	Details	Container Formats
JPEG	•	•	Base+progressive	JPEG (.jpg)
GIF		•		GIF (.gif)
PNG	•	•		PNG (.png)
BMP		•		BMP (.bmp)
WEBP	• (Android 4.0+)	• (Android 4.0+)		WebP (.webp)

Android native video codec support

Format / Codec	Encoder	Decoder	Details	Container Formats
H.263	•	•		• 3GPP (.3gp) • MPEG-4 (.mp4)
H.264 AVC	• (Android 3.0+)	•	Baseline Profile (BP)	• 3GPP (.3gp) • MPEG-4 (.mp4) • MPEG-TS (.ts, AAC audio only, not seekable, Android 3.0+)
MPEG-4 SP		•		3GPP (.3gp)
VP8		• (Android 2.3.3+)	Streamable only in Android 4.0 and above	• WebM (.webm) • Matroska (.mkv, Android 4.0+)
H.265 HEVC		• (Android 5.0+)	Main Profile Level 3 for mobile devices and Main Profile Level 4.1 for Android TV	• MPEG-4 (.mp4)

Android Video Encoding Recommendations

	SD (Low quality)	SD (High quality)	HD (Not available on all devices)
Video codec	H.264 Baseline Profile	H.264 Baseline Profile	H.264 Baseline Profile
Video resolution	176 x 144 px	480 x 360 px	1280 x 720 px
Video frame rate	12 fps	30 fps	30 fps
Video bitrate	56 Kbps	500 Kbps	2 Mbps
Audio codec	AAC-LC	AAC-LC	AAC-LC
Audio channels	1 (mono)	2 (stereo)	2 (stereo)
Audio bitrate	24 Kbps	128 Kbps	192 Kbps

-Interlace / Progressive ?
-Keyframe ?
-W,H can be mod 4 ?

Interlaced Scan

Field 1 + Field 2 = Frame (Complete Image)

DigitalPhotographyWriter.com

Progressive

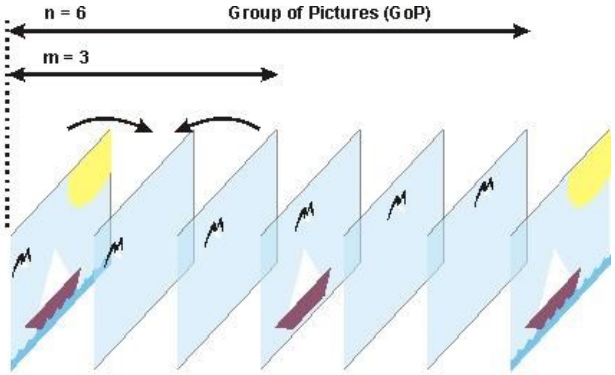
Frame 1
(All lines)

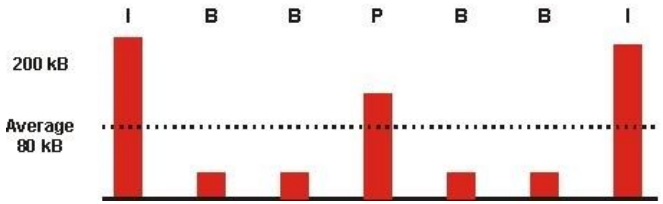
Frame 2
(All lines)

Display Rate: 60 frames per second (North America)

http://mycreate95.blogspot.com/2013/03/interlaced-scan-progressive-scan.html

MPEG Group of Pictures

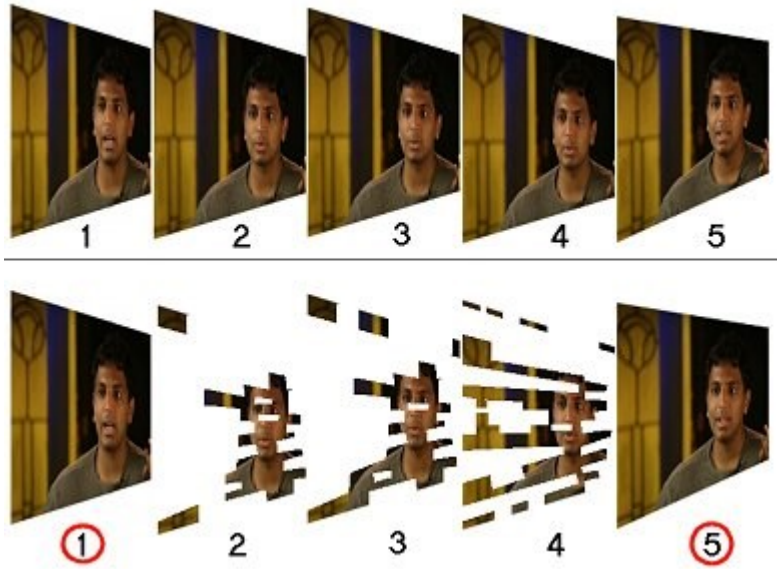




I frame P frame B frame

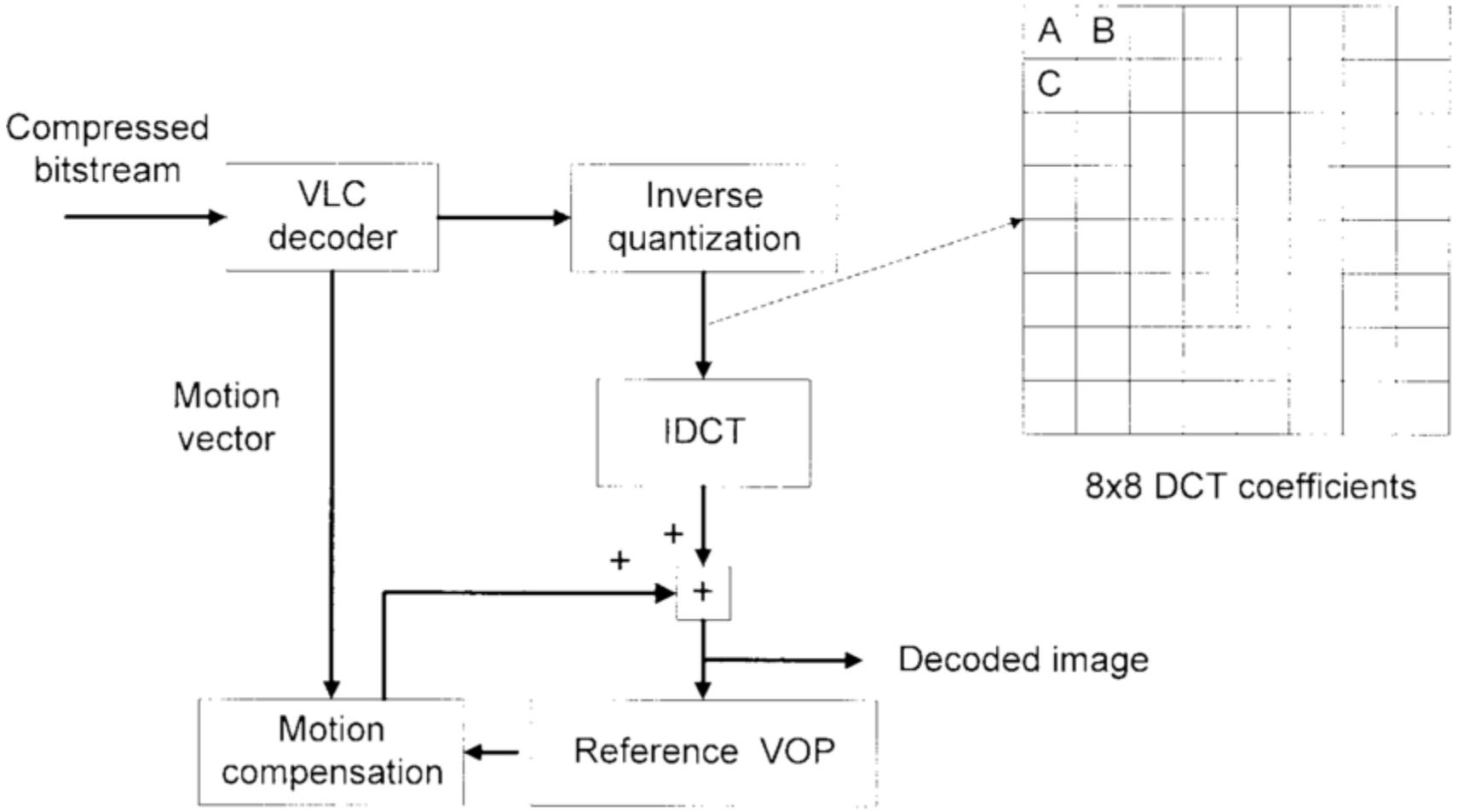
Intra
Inter prediction
Biprediction
Direct mode

- Abundance of Direct macroblock in B frames



http://www.divx-digest.com/articles/flask-method_br.html

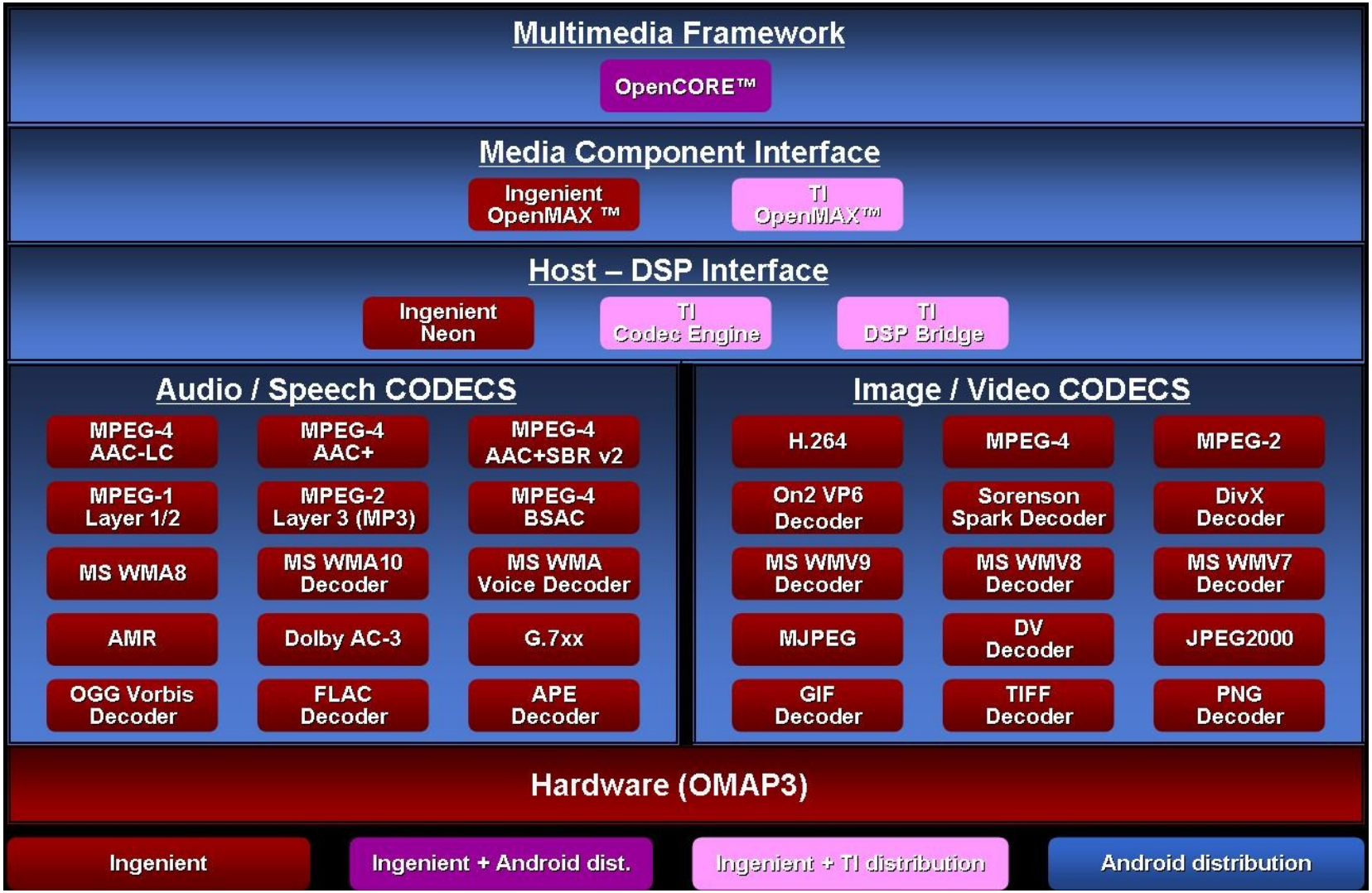
MPEG Decoder



https://www.researchgate.net/figure/Decoder-block-diagram-of-the-MPEG-4-and-DCT-coefficients-after-inverse-quantization_fig1_3307981

Android media player framework

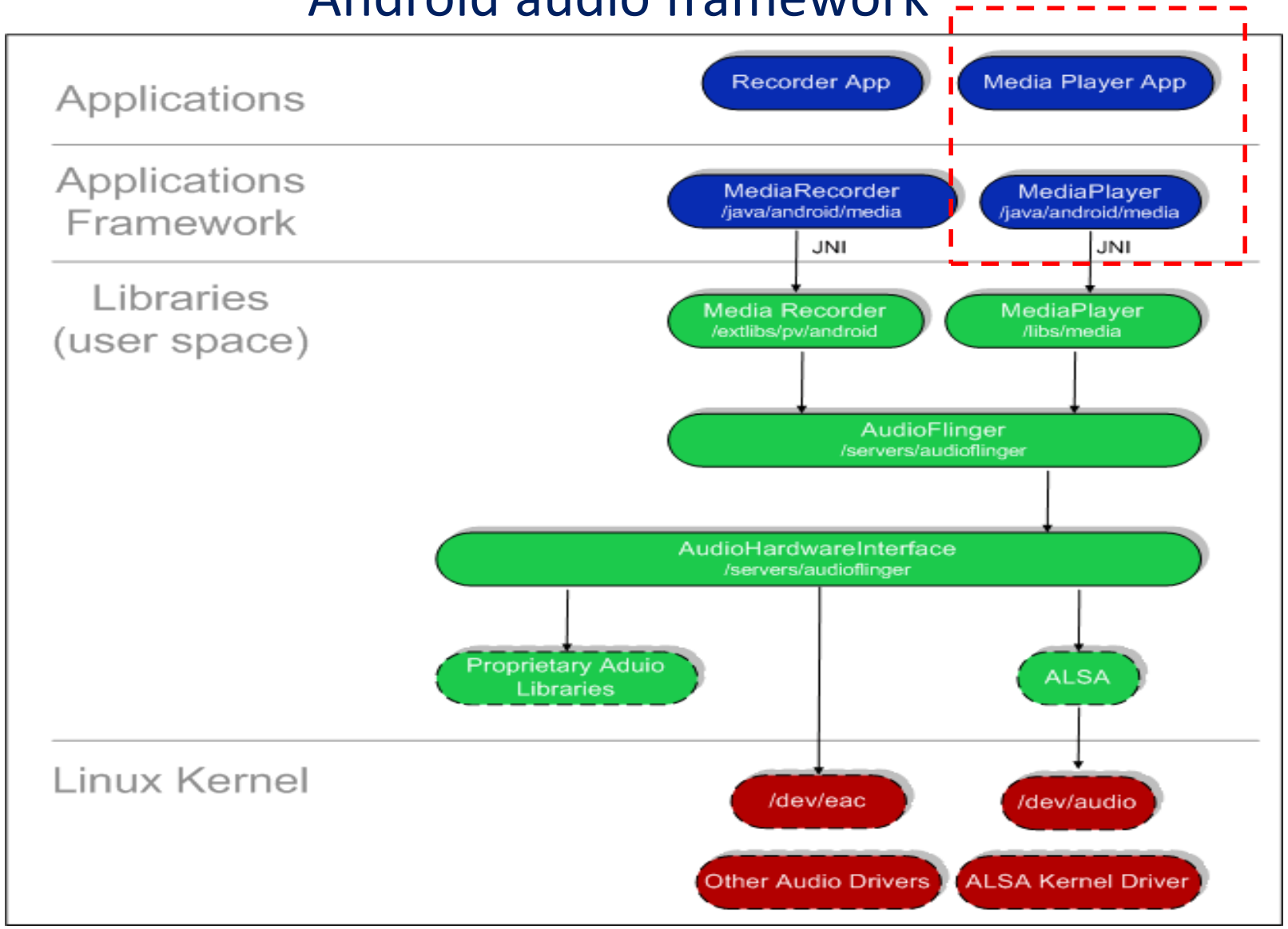
Ingenient Android Multimedia



http://www.arm.com/community/partners/display_product/rw/ProductId/4738/

Play sound

Android audio framework



Media player

Android multimedia player

-Play from resource or assets
-Play from SD CARD
-Play from online source

Play sound

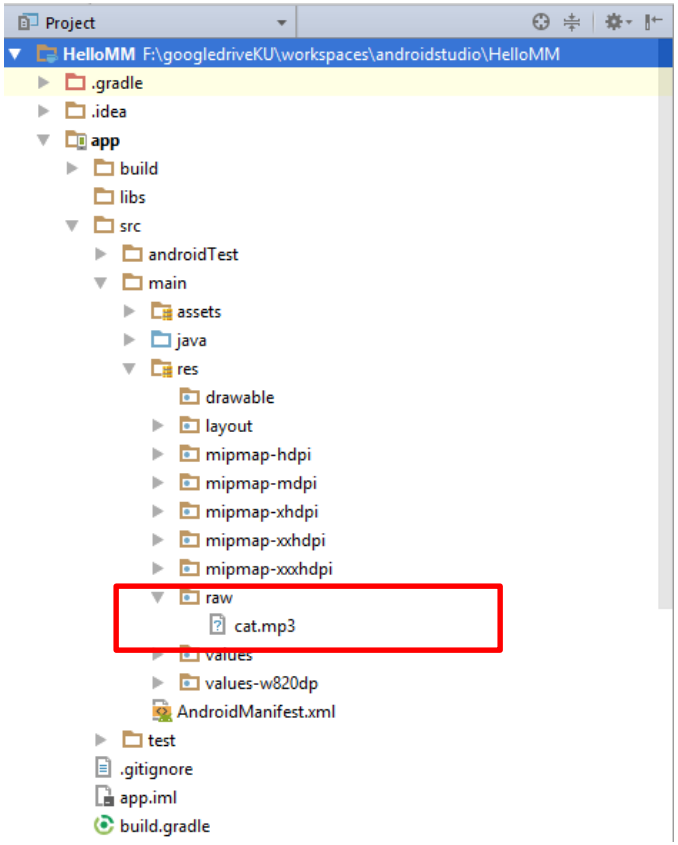
Play sound

Project: android_ex_audio

Play sound

-Play from resource or assets

Step 1: Create folder “raw” in resource folder and put audio file in that folder

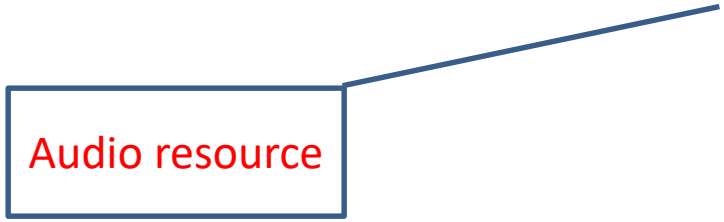


Play sound

-Play from resource or assets

Step 2: Instantiate the media player class


```
MediaPlayer player;
player = MediaPlayer.create(this, R.raw.cat);
```



Play sound

-Play from resource or assets

Step 3: Set loop property

```
player.setLooping(false);
```

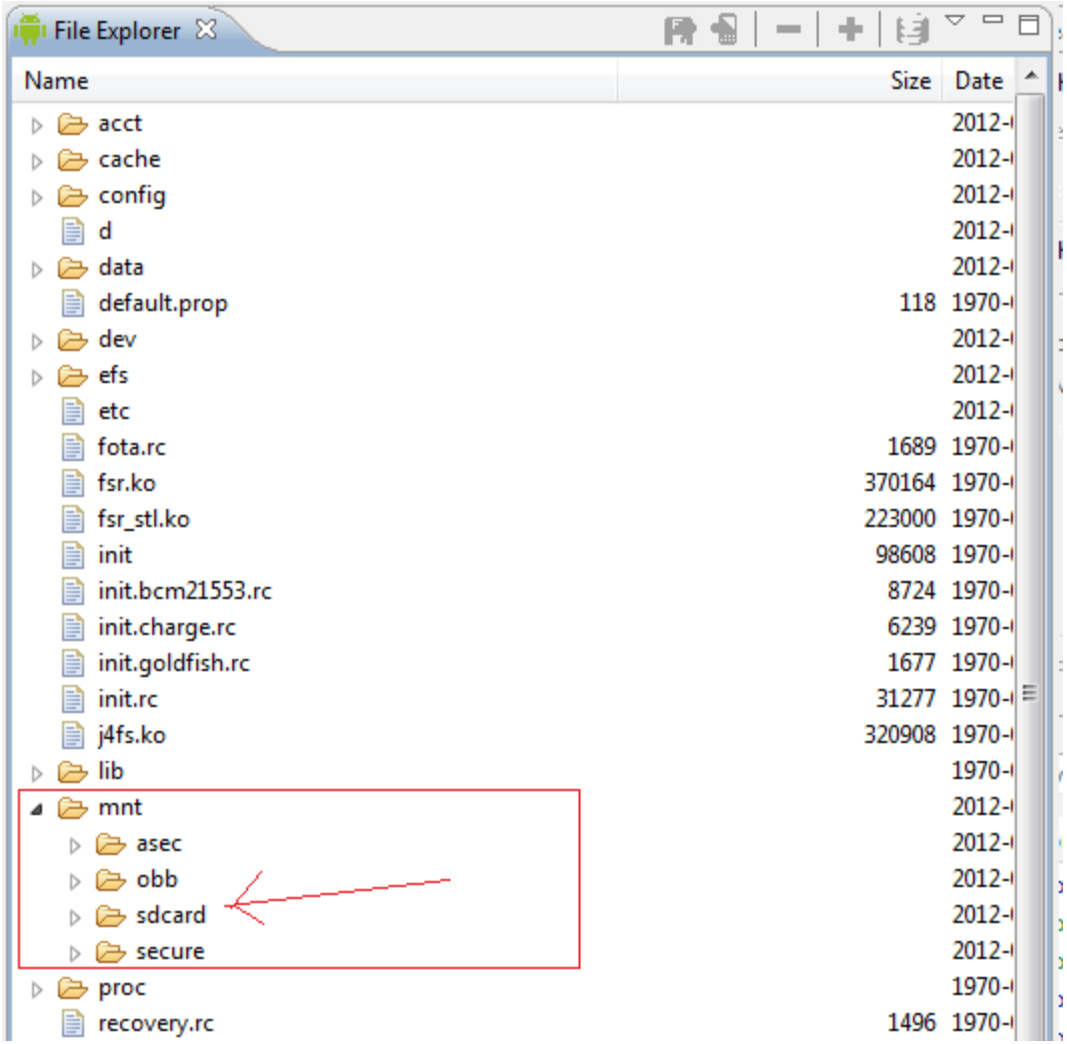
Step 4: Control media player

```
player.start();
player.pause();
player.stop();
```

Play sound

-Play from SD CARD

Step 1: Put audio file in folder /mnt/sdcard



Play sound

-Play from SD CARD

player = MediaPlayer.*create(getApplicationContext(), Uri.parse("/sdcard/song.mp3"));*

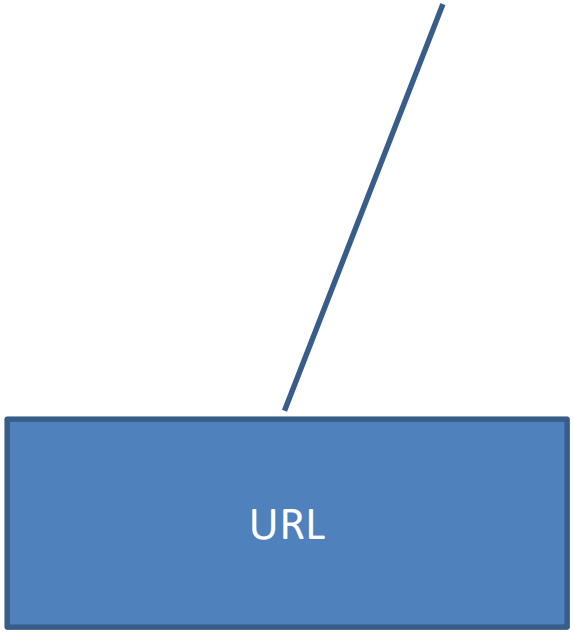
OR

player = MediaPlayer.*create(getApplicationContext(), Uri.parse(Environment.getExternalStorageDirectory().getPath()+"/song.mp3"));*

Play sound

-Play from URL

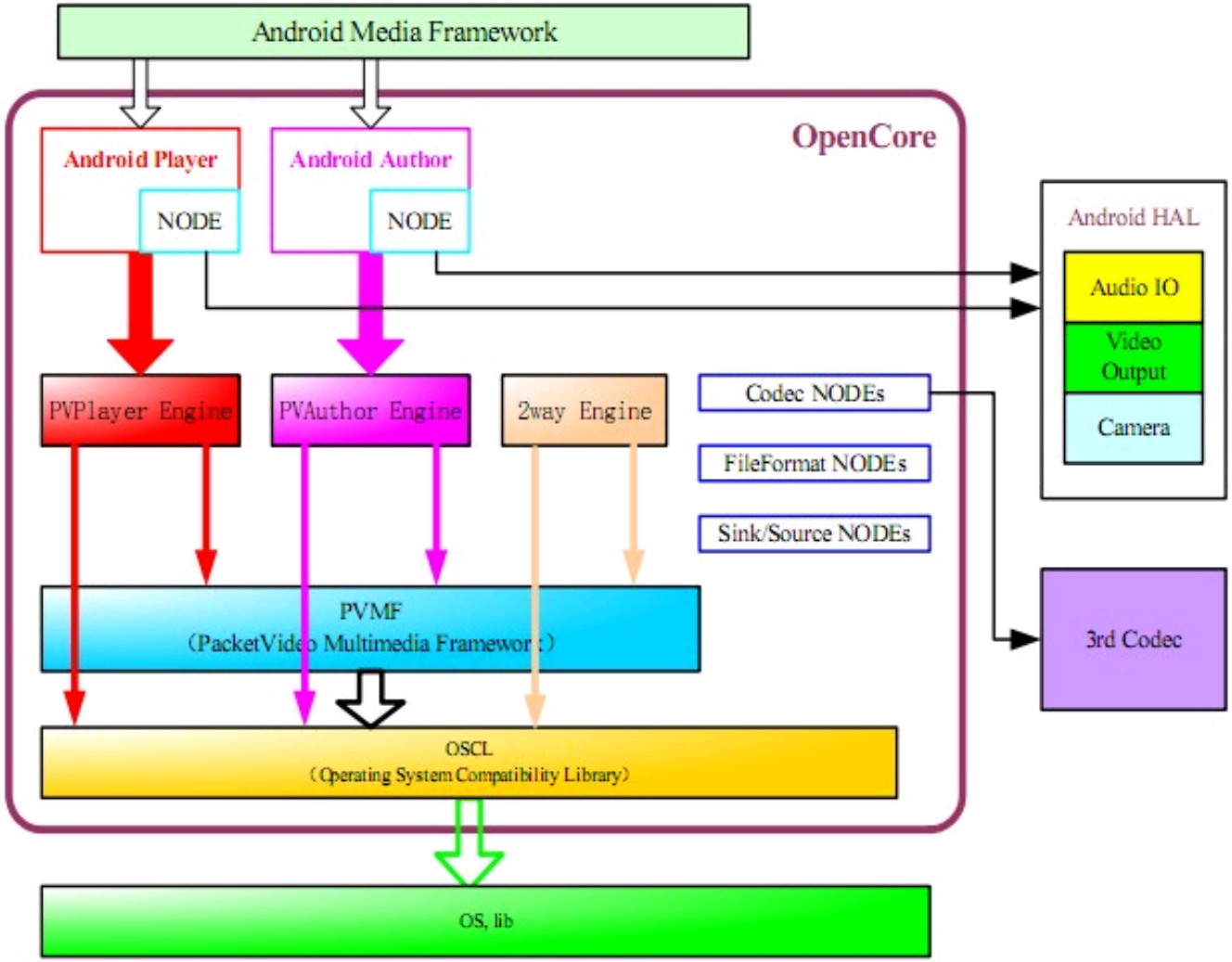
```
player = MediaPlayer.create(getApplicationContext(), Uri.parse("http://www.peoplecine.com/share/neko.mp3"));
```



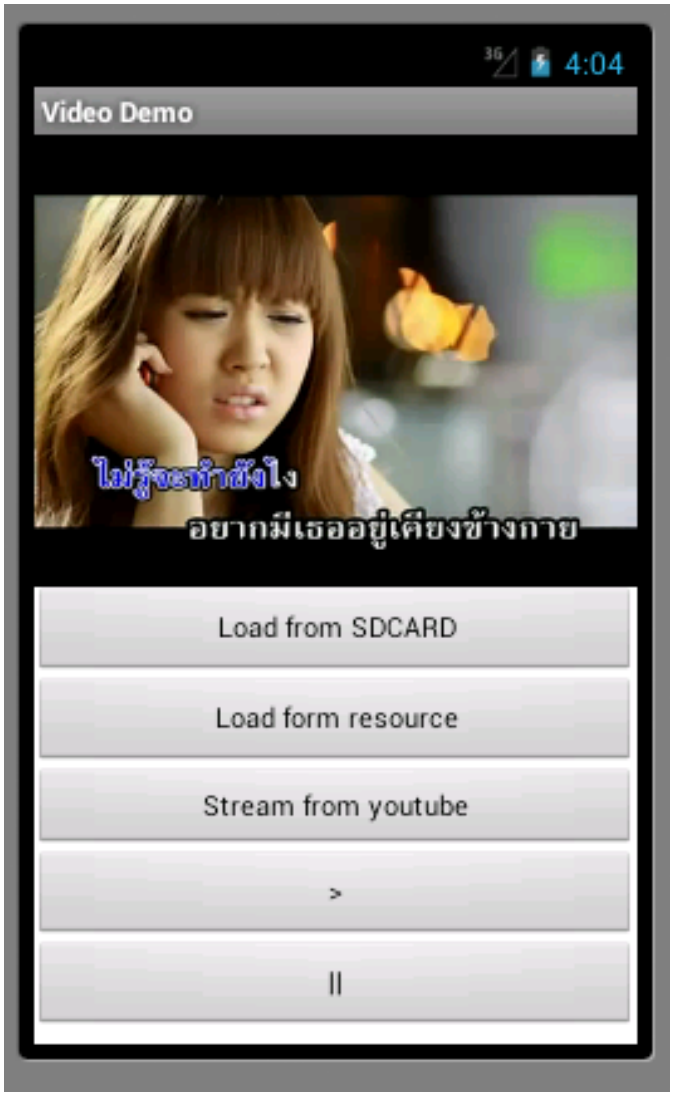
Play Video

Android media player framework

OpenCORE framework



Play Video

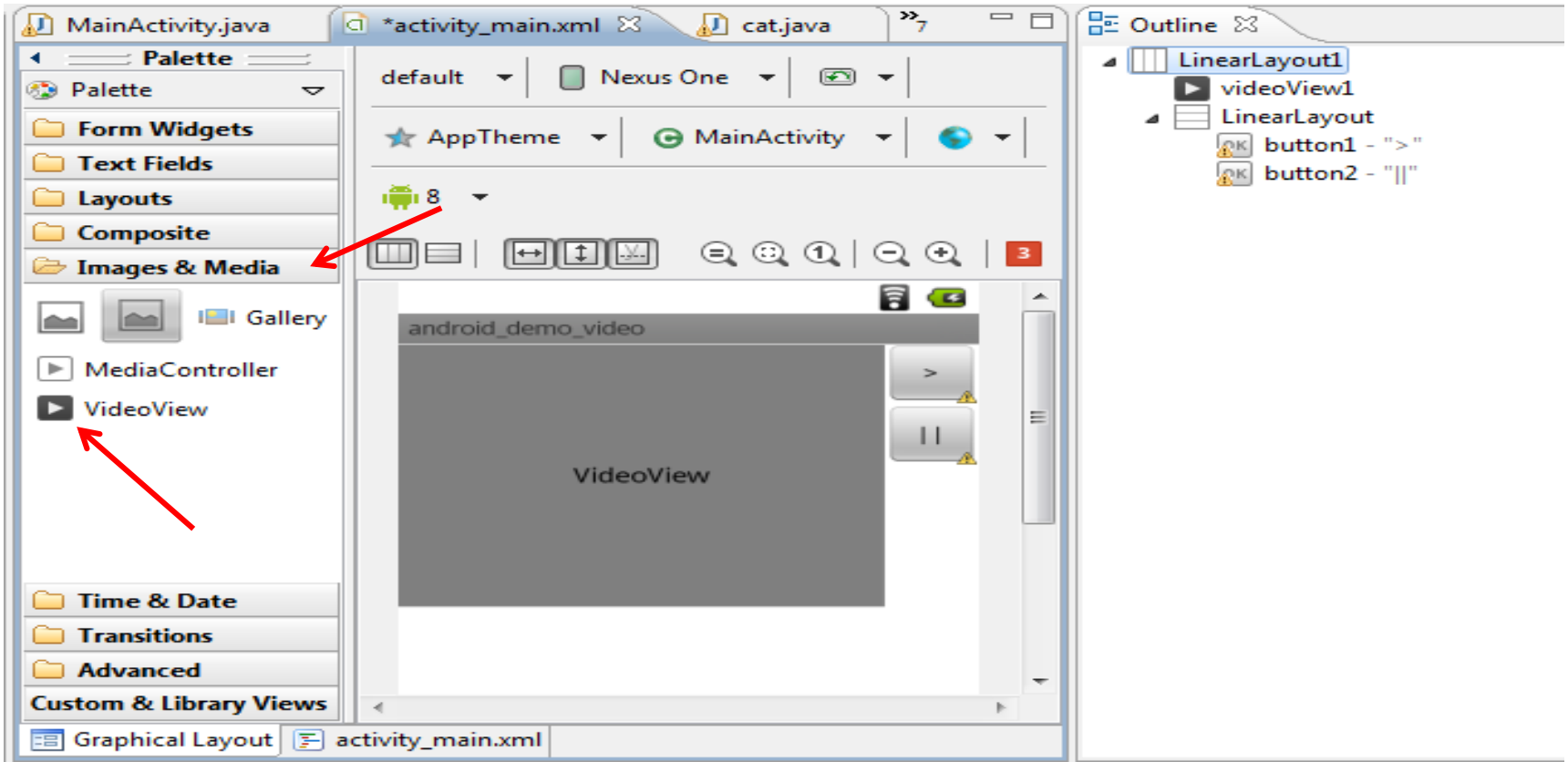


Project: android_demo_video

Play Video

-Play from resource or assets

Step 1: Put video file to resource folder (same as audio)
Step 2: Create VideoView Layout



Play Video

-Play from resource or assets

Step 3: Instantiate videoview

```
VideoView video1;
video1=(VideoView)this.findViewById(R.id.videoView1);
```

Step 4: Set target video

```
video1.setVideoURI(Uri.parse("android.resource://"+
getPackageName()+"/"+R.raw.video2));
```

Play Video

-Play from resource or assets

Step 5: Play video

```
video1.setMediaController(new MediaController(this));
 video1.requestFocus();
 video1.start();
```

Play Video

-Play from SD CARD

```
video1.setVideoURI(Uri.parse("/sdcard/test.3gp"));
```

-Play from Youtube

```
video1.setVideoURI(Uri.parse("rtsp://v2.cache8.c.youtube.com/CiILENy73wIaGQnuwiVlTL59QBMYDSANFEgGUgZ2aWRlb3MM/0/0/0/video.3gp"));
```

```
<uses-permission android:name="android.permission.INTERNET" />
```

URL to RTSP from http://www.drpaween.com/ohm/cs436/url2rtsp/

Thank you ☺