LiveVideo StackCon 音视频技术大会

技术开启新“视”界

Technology Bring New Vision

2018.10.19-20 北京丽亭华苑酒店

LiveVideoStackCon 2018 讲师热身分享会

01

6月28日19:30 准时开始

《FFmpeg加速那些事儿》

赵军 英特尔资深软件开发工程师

关注LiveVideoStack公众号

回复『热身』获得进群方式

LiveVideoStack 音视频技术社区

CSDN

Introduction to FFmpeg

Zhao, Jun@DCG/NPG

Agenda

- What is FFmpeg （ LibAV ）
- Components of FFmpeg
- Developing with FFmpeg
- FFmpeg HWACCEL
- Backup

What's FFmpeg

- Tools and Library
  - the **most popular open-source multimedia** manipulation tools with a library of plugins that can be applied to various parts of the audio and video processing pipelines and have achieved wide adoption across the world.
  - To Convert, manipulate and stream multimedia formats and protocols
- Written in C/assembly(yasm/nasm), Open Source, relies on external libraries (libx264,libmp3lame,…) **when it makes sense**
- Multiplatform: GNU/Linux, Mac OSX, Android, MS Windows, …
- License: GNU GPLv2 or GNU LGPLv2.1.
  - Refer to https://ffmpeg.org/legal.html

History

- **2000**: **Fabrice Bellard** starts the project with the initial aim to implement an MPEG encoding/decoding library. The resulting project is integrated as multimedia engine in MPlayer, which also hosts the project.
- **2003**: Fabrice Bellard leaves the project, **Michael Niedermayer** acts as project maintainer since then.
- **March 2009**: release version 0.5, first official release
- **January 2011**: a group of discontented developers takes control over the FFmpeg web server and creates an alternative Git repo, a few months later a proper fork is created (**Libav**) https://libav.org/
- Now: Stable release 4.0.0 (April 20th, 2018)

Who are using FFmpeg

Video/Audio Players and

Video/Audio/3D Editors

Cloud Media

Other Software

Browser 

Graphic Libraries 

VoIP 

Disk authoring 

FFmpeg components

Commands (console)

ffmpeg
Command tool to do transcoding

ffplay
Simple player with SDL using ffmpeg demux/decoder

ffprobe
Tool to extract the information of multi-media stream

ffserver
Real-time stream server to broadcast multi-media stream

Libraries

libavdevice

libavutil
Common tool library

libavresample

libavformat
mux / demux (mp4,mkv, RTSP,RTP,RTMP,…)

libswresample

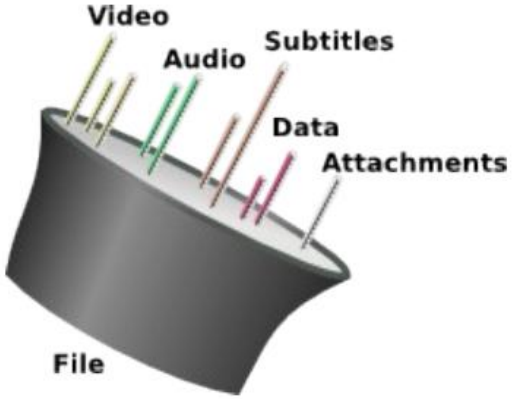
libavcodec
A Library to implement most of A/V codec, and used by most of popular codec tools(> 300 codecs)

libpostproc

libavfilter
Library for a/v filters which to implement all kind of effects, such as scale, crop, frc, etc

Basic

1. A file can be: Regular file, Pipe, Network stream, Device
2. A file has distinct multiplexed streams, each stream of 5 possible types
3. In FFmpeg, we talked with:
   Container/Streams/Codec/Frames/Packets
4. A sample to demo FFmpeg decode




```
10 OPEN video_stream FROM video.avi
20 READ packet FROM video_stream INTO frame
30 IF frame NOT COMPLETE GOTO 20
40 DO SOMETHING WITH frame
50 GOTO 20
```

http://dranger.com/ffmpeg/tutorial01.c

Libavformat

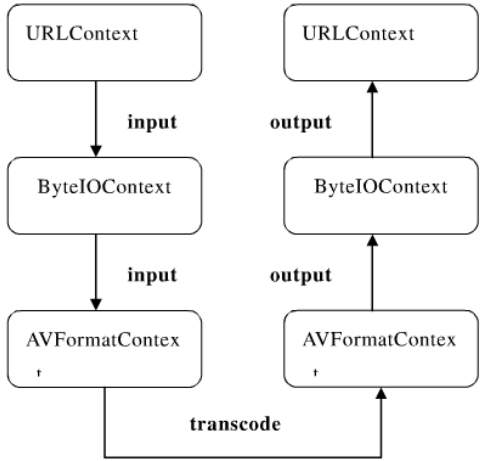
1. Provide the support to demuxer/muxer of all kinds of container, such as flv, ts, ps, ogg, mp4, asf, avi, etc.
2. Provide support to all kinds of protocols, such as local file, rtsp, http, ftp, rtp, hls, dash etc.
3. Structures in libavformat:

   AVFormatContext, AVOutputFormat, AVInputFormat, AVStream, URLProtocol
4. Easy to add support for new format: protocol and muxer
5. Flow chart relative to Data IO in FFmpeg

https://wiki.multimedia.cx/index.php?title=FFmpeg_demuxer_HOWTO

```
static const AVClass mov_class = {
    .class_name = "mov,mp4,m4a,3gp,3g2,mj2",
    .item_name  = av_default_item_name,
    .option     = mov_options,
    .version    = LIBAVUTIL_VERSION_INT,
};

AVInputFormat ff_mov_demuxer = {
    .name           = "mov,mp4,m4a,3gp,3g2,mj2",
    .long_name      = NULL_IF_CONFIG_SMALL("QuickTime / MOV"),
    .priv_class     = &mov_class,
    .priv_data_size = sizeof(MOVContext),
    .extensions     = "mov,mp4,m4a,3gp,3g2,mj2",
    .read_probe     = mov_probe,
    .read_header    = mov_read_header,
    .read_packet    = mov_read_packet,
    .read_close     = mov_read_close,
    .read_seek      = mov_read_seek,
    .flags          = AVFMT_NO_BYTE_SEEK,
};
```



Libavcodec

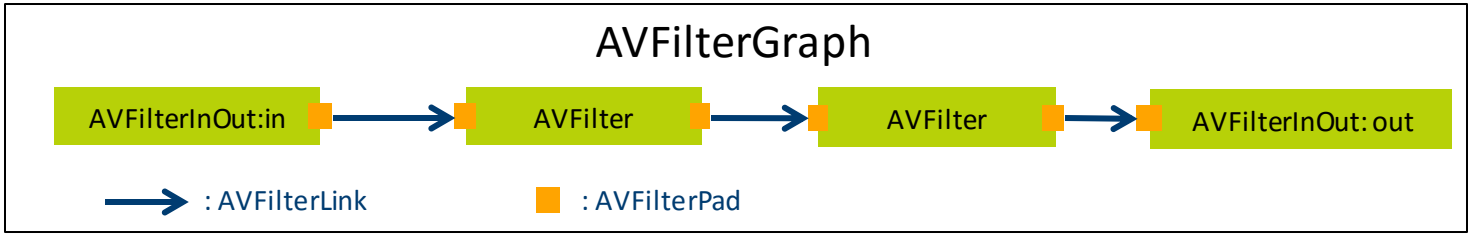
1. Provide the support to A/V decoder/encoder: >200 video codec, >150 audio codec, and most of subtitle codec.
2. Structures in libavcodec: AVCodecContext, AVCodec
3. FFmpeg provides codec support with 2 methods:
    1. native open-source (h264 decoder)
    2. third-party (libfdk_aac, libmp3lame, libx264)
4. Add new codec support for FFmpeg.

https://wiki.multimedia.cx/index.php?title=FFmpeg_codec_

```
AVCodec ff_mpeg1video_decoder = {
    .name                  = "mpeg1video",
    .long_name             = NULL_IF_CONFIG_SMALL("MPEG-1 video"),
    .type                  = AVMEDIA_TYPE_VIDEO,
    .id                    = AV_CODEC_ID_MPEG1VIDEO,
    .priv_data_size        = sizeof(Mpeg1Context),
    .init                  = mpeg_decode_init,
    .close                 = mpeg_decode_end,
    .decode                = mpeg_decode_frame,
    .capabilities          = AV_CODEC_CAP_DRAW_HORIZ_BAND | AV_CODEC_CAP_DR1 |
                             AV_CODEC_CAP_TRUNCATED | AV_CODEC_CAP_DELAY |
                             AV_CODEC_CAP_SLICE_THREADS,
    .flush                 = flush,
    .max_lowres            = 3,
    .update_thread_context = ONLY_IF_THREADS_ENABLED(mpeg_decode_update_thread_con
};
```

Libavfilter

1. Libavfilter provides support for a/v post-process, such as scaling, de-interlace, FRC, denoise, audio resample, etc
2. Structures: AVFilterGraph, AVFilterInOut, AVFilter, AVFilterContext, AVFilterLink, AVFilterPad, AVFilterFormats

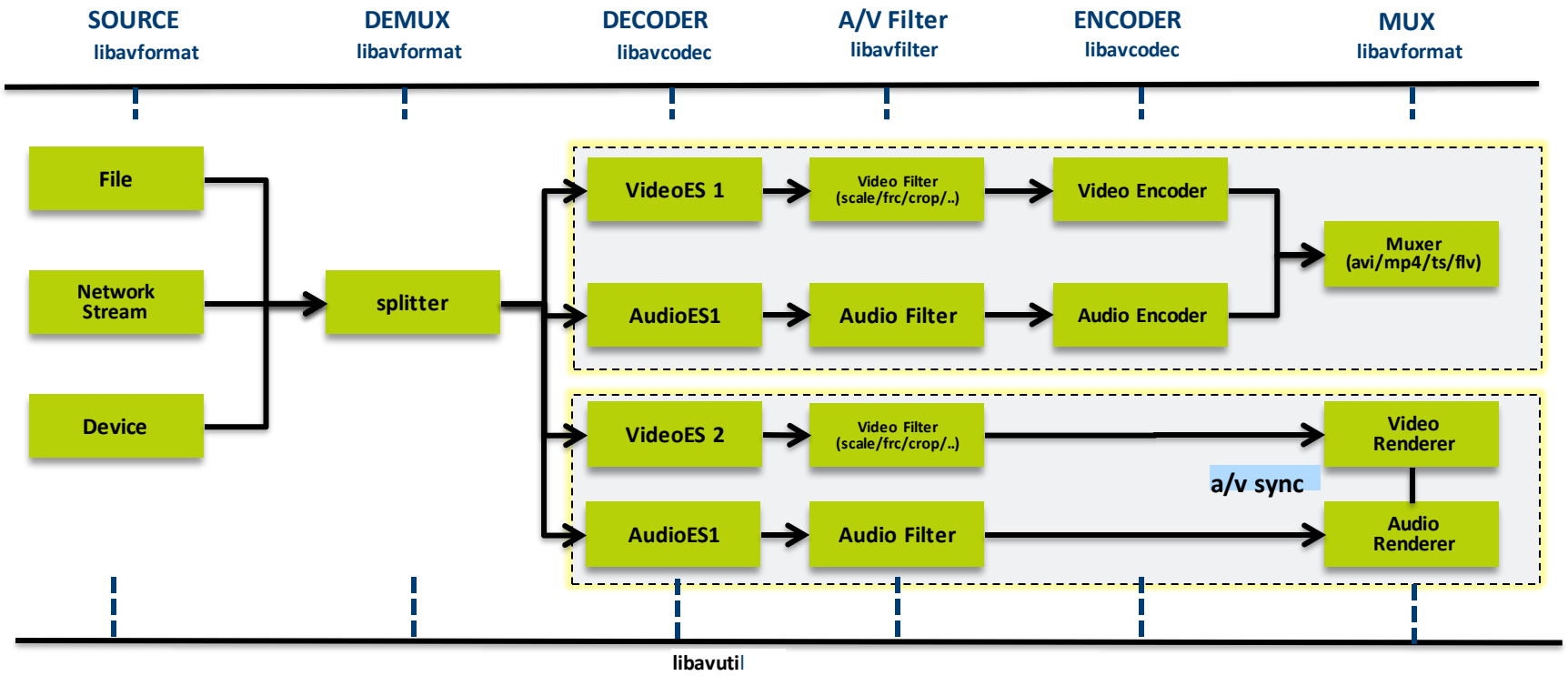


3. Add new filter into FFmpeg: define AVClass, input/output AVFilterPad, AVFilter
4. How to use filter in FFmpeg:

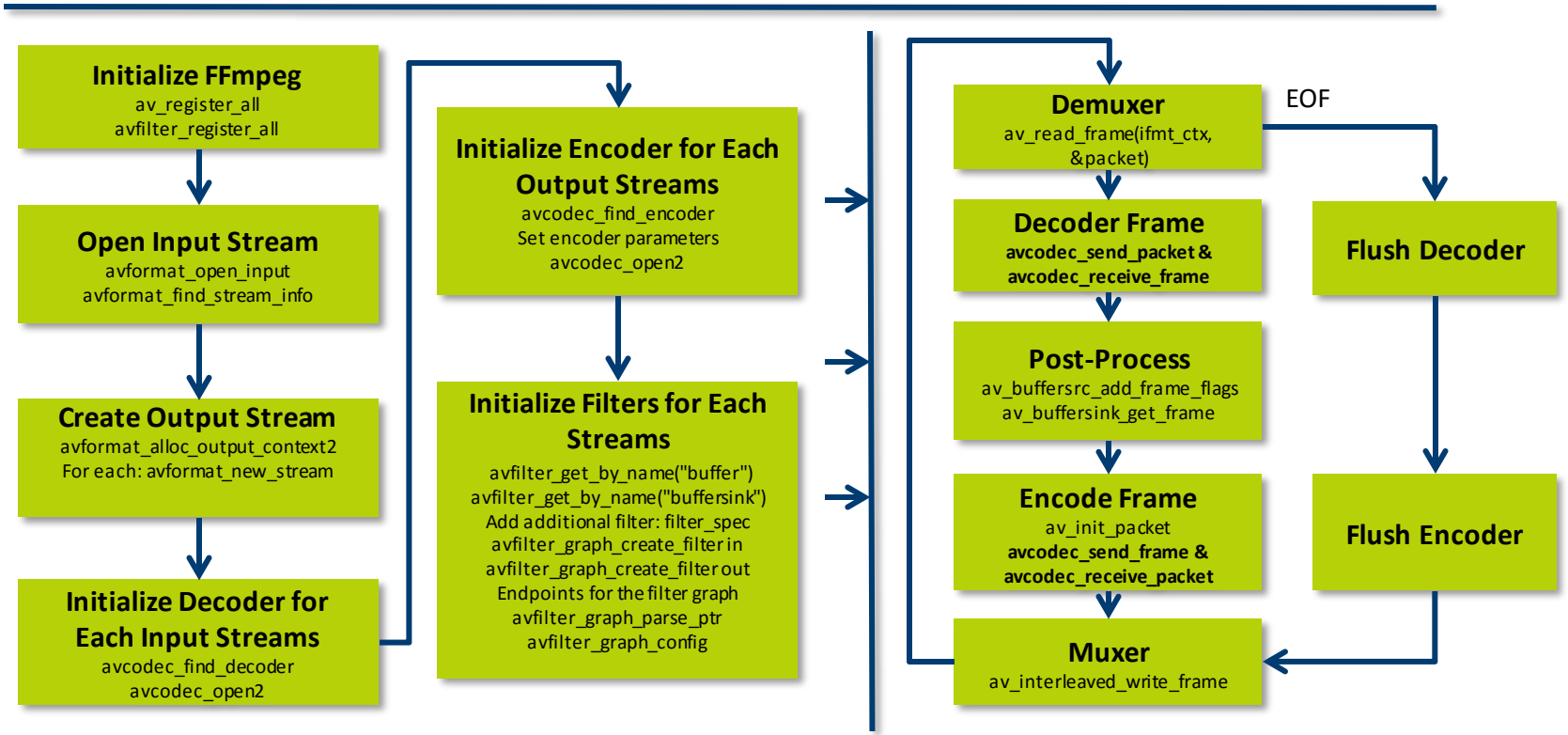
   ffmpeg -i input -vf yadif=0:0:0,scale=iw/2:-1 output

   https://github.com/FFmpeg/FFmpeg/blob/master/doc/writing_filters.txt

FFmpeg Transcoding



Developing with FFmpeg

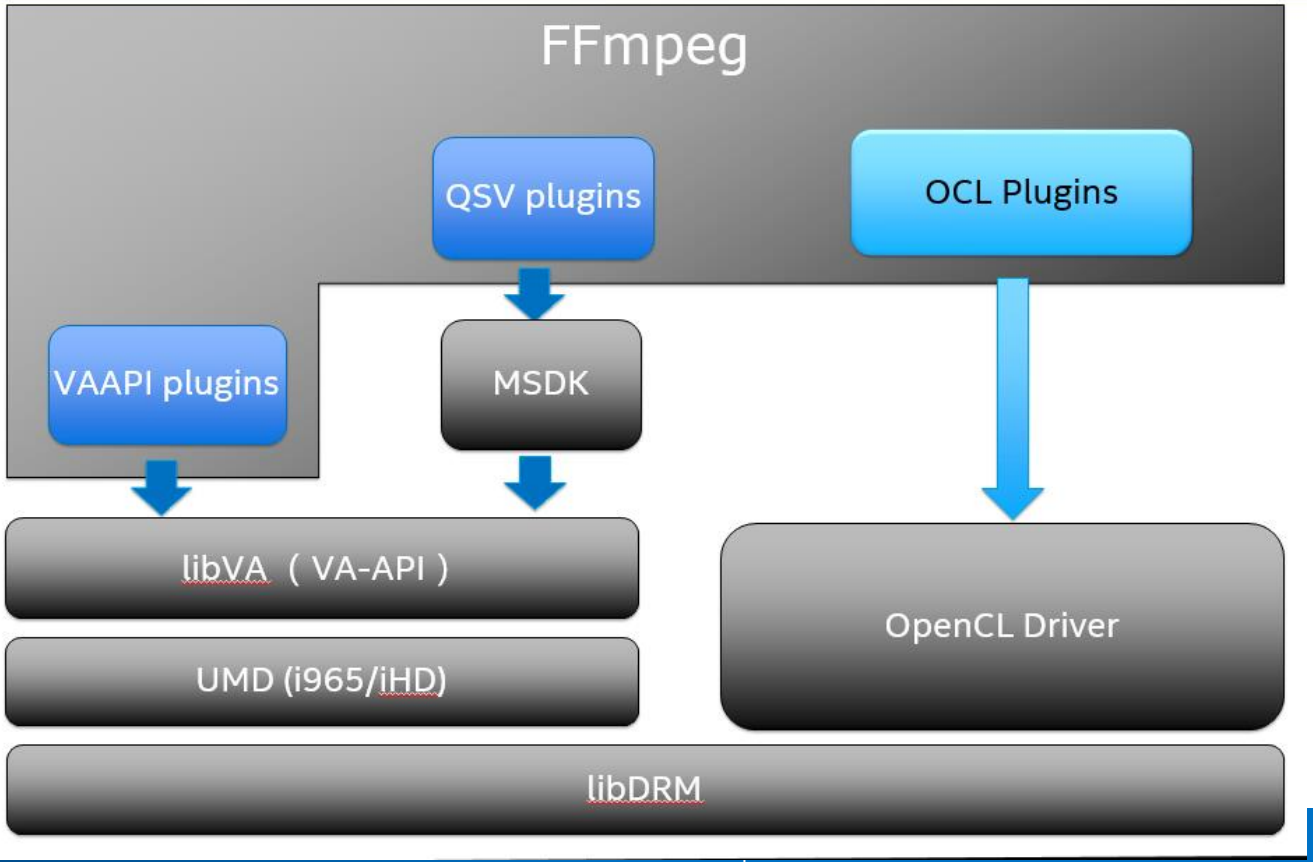


FFmpeg HWAccel with intel GPU

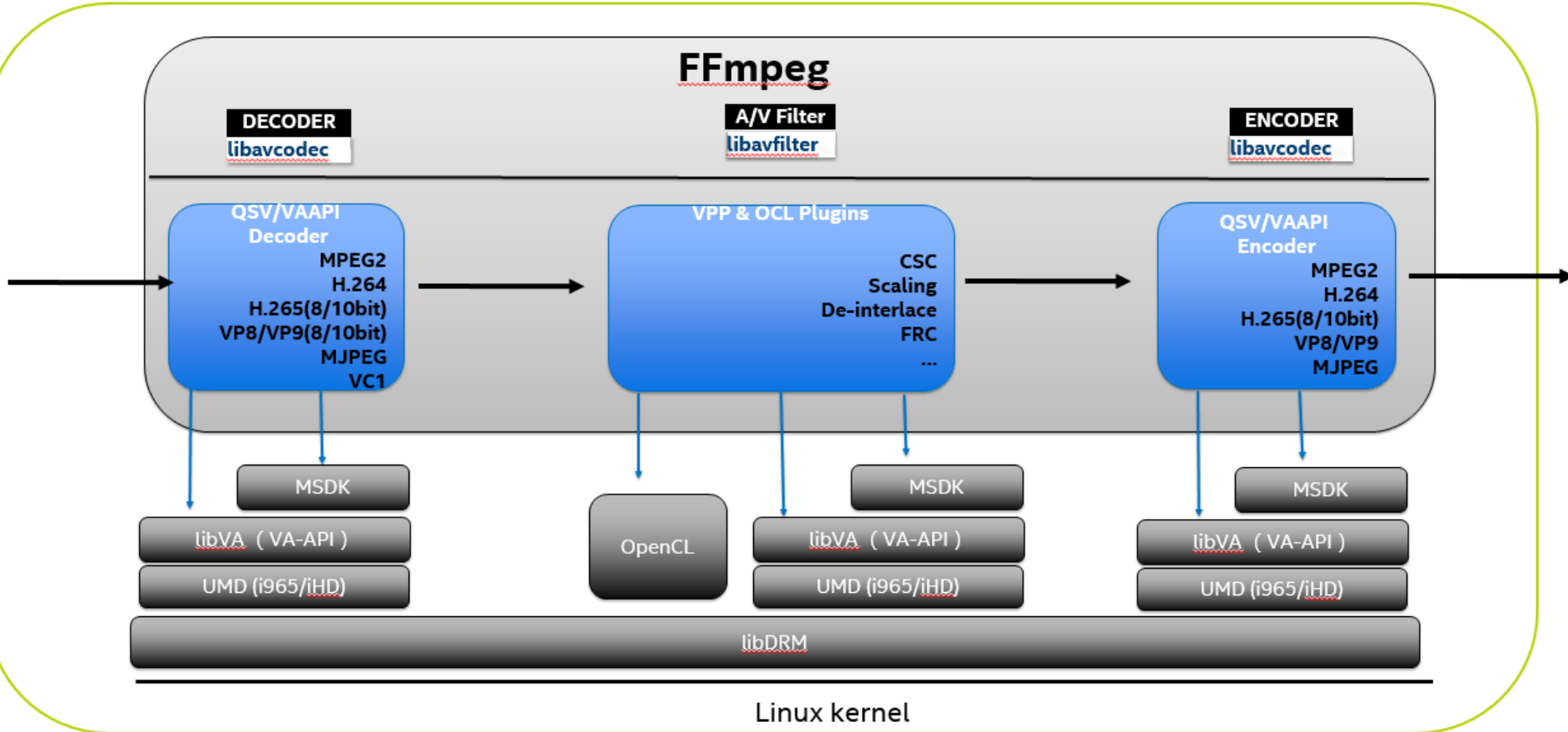
Solution overview:

- FFmpeg is most popular open source framework; it can help deploy Intel media solution in a quickest way.
- FFmpeg QSV plugins are based on MediaSDK; it has widely accepted by customers. vpp added as a filter.
- VAAPI is lower level API; FFmpeg VAAPI plugins provides more flexible solution for customers.
- Integrate 3rd-party OCL/OpenCV/Vulkan video processing Library to enrich the solution.
- FFmpeg Plugins with HW acceleration will speed up development for different usage

Big pictures



More Details



How to enable intel GPU to FFmpeg?

- Solution1 (FFmpeg QSV)
  - ✓ “Extra” FFmpeg library, now we will used the MSDK/libyami for Intel HW Acceled Decoder/Encoder/VPP/Transcode

You can think it’s like FFmpeg + libx264

- Solution2 (FFmpeg VAAPI)
  - ✓ Use general hardware accelerators interface (VAAPI, TI-DSP, CUDA…)
  - ✓ Light-weight library(less dependent on other libraries), flexible to support our customer
  - ✓ Native Intel GPU hardware accelerated video encoder and decoder through the integration of VAAPI

VA-API(Video Acceleration API) and driver

WHAT IS VA-API?

- An API specification
- A library implementation
- Open Source MIT license
- It is a front-end
- Opens and registers a backend

WHICH BACKEND?

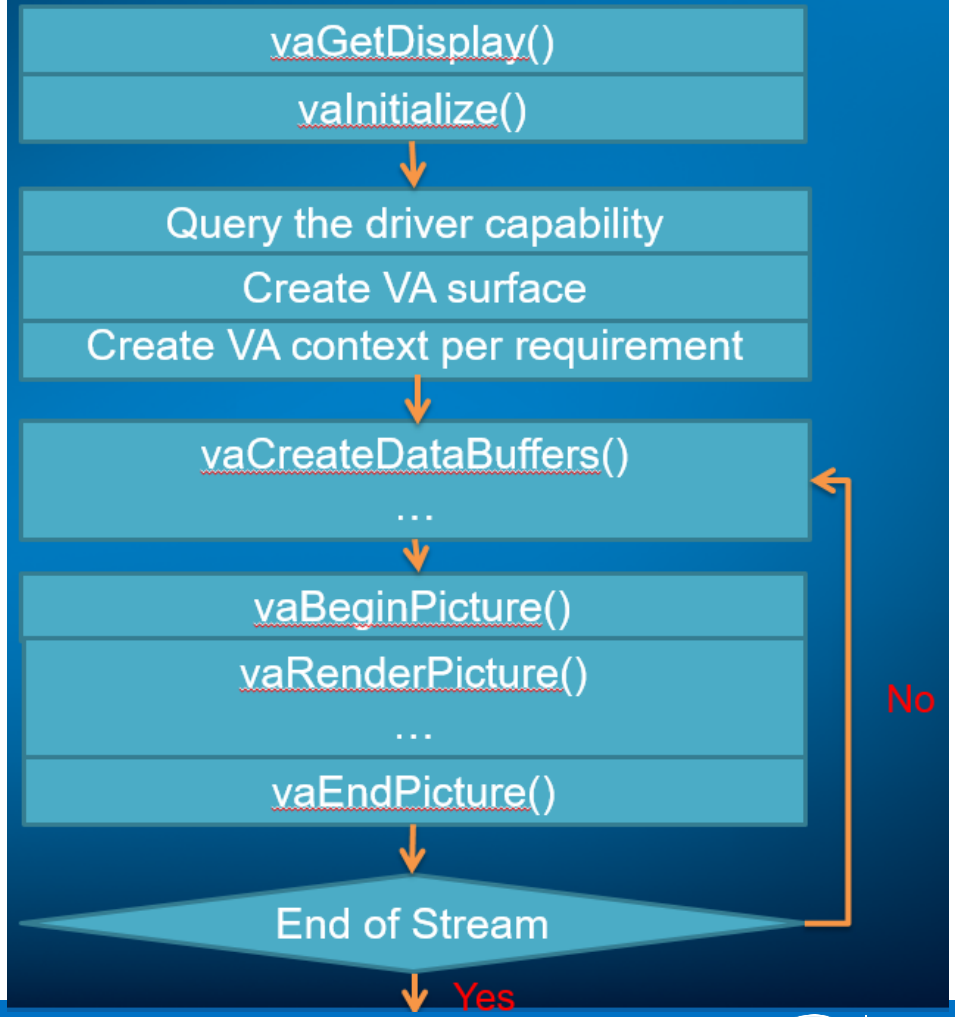
- Intel VA(i965) driver for Intel chip-sets
- Intel hybrid driver
- Intel HD driver/Media Driver
- Mesa's state-trackers for gallium drivers:
  - radeon, nouveau (?), freedreno, …
- obsoleted API bridges
  vdpau—va bridge
  powervr—va bridge

Basic concept in VA-API

- VADisplay
  - X11, DRM, Wayland, Android, etc.
- VAConfigID
  - VLD for requested codec.
- VAContexID
  - "Virtual" video processing pipeline. Identified by a unique context id
- VASurfaceID
  - Render targets. The major object in VA-API, used to hold pixel data for each frame
  - Not accessible to the client.
- VABufferID
  - Buffers are mainly used to pass data to the VA drivers ， data, parameters, quantization matrix, slice info, etc.

Program flow

- The same program flow control for decoding / encoding / video



Open questions

- FFmpeg and Media Studio/Media SDK
- FFmpeg and gstreamer
- FFmpeg and openMAX
- Supply a solution with FFmpeg based on Intel GPU
- The Road Ahead

  opencl/vulkan/opencv/DL/…

Q & A

More Questions ?

Use the Source, Luke!

Stop worrying and did it anyway!

Backup

reference

https://en.wikipedia.org/wiki/FFmpeg

git clone git://source.ffmpeg.org/ffmpeg.git ffmpeg

http://ffmpeg.org/

http://ffmpeg.org/developer.html#Contributing

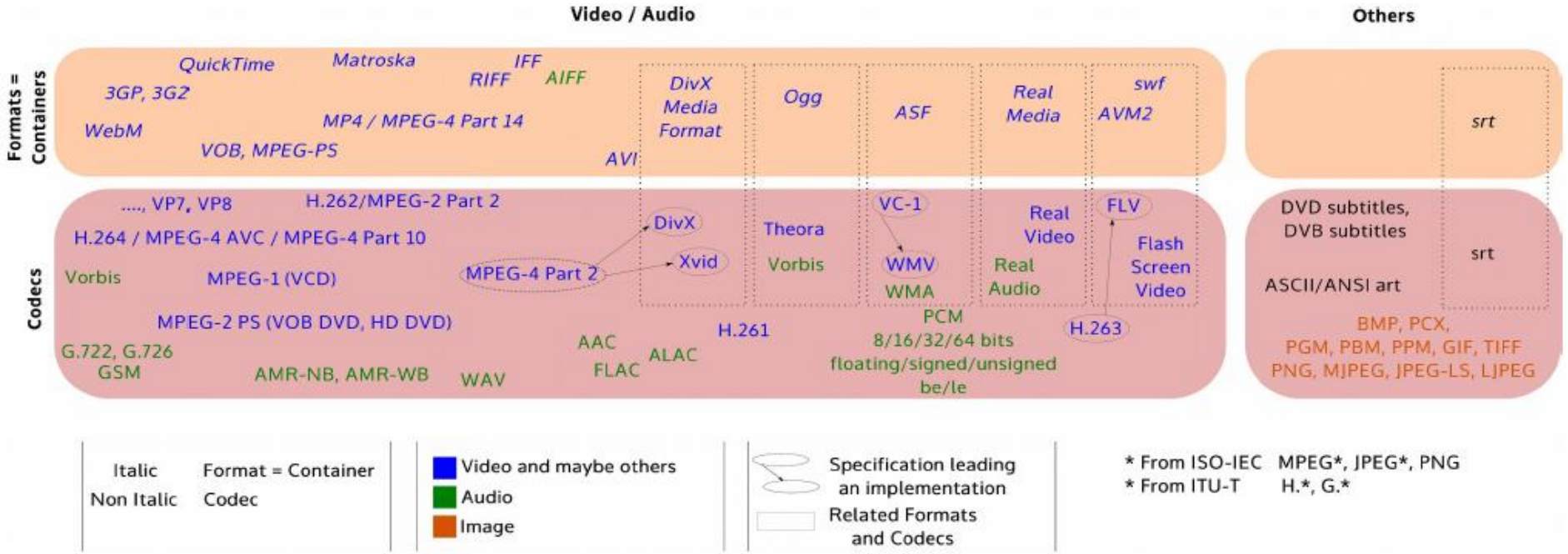
http://ffmpeg.org/legal.html

http://www.oschina.net/question/tag/ffmpeg

https://trac.ffmpeg.org/wiki/HWAccelIntro

https://trac.ffmpeg.org/wiki/Hardware/VAAPI

FFmpeg supported format and codec



FFmpeg HWaccel status

	Decoder		Encoder		Other support			
	Internal	Standalone	Hardware output	Standalone	Hardware input	Filtering	Hardware context	Usable from ffmpeg CLI
AMF	N	N	N	Y	Y	N	Y	Y
CUDA / CUVID / NVENC	N	Y	Y	Y	Y	Y	Y	Y
Direct3D 11	Y	-	Y	-	-	F	Y	Y
Direct3D 9 / DXVA2	Y	-	Y	-	-	N	Y	Y
libmfx	-	Y	Y	Y	Y	Y	Y	Y
MediaCodec	-	Y	Y	N	N	-	N	N
Media Foundation	-	N	N	N	N	N	N	N
MMAL	-	Y	Y	N	N	-	N	N
OpenCL	-	-	-	-	-	Y	Y	Y
OpenMAX	-	N	N	Y	N	N	N	Y
RockChip MPP	-	Y	Y	N	N	-	Y	Y
V4L2 M2M	-	Y	N	Y	N	N	N	Y
VAAPI	Y	-	Y	Y	Y	Y	Y	Y
VDPAU	Y	-	Y	-	-	N	Y	Y
VideoToolbox	Y	N	Y	Y	Y	-	Y	Y

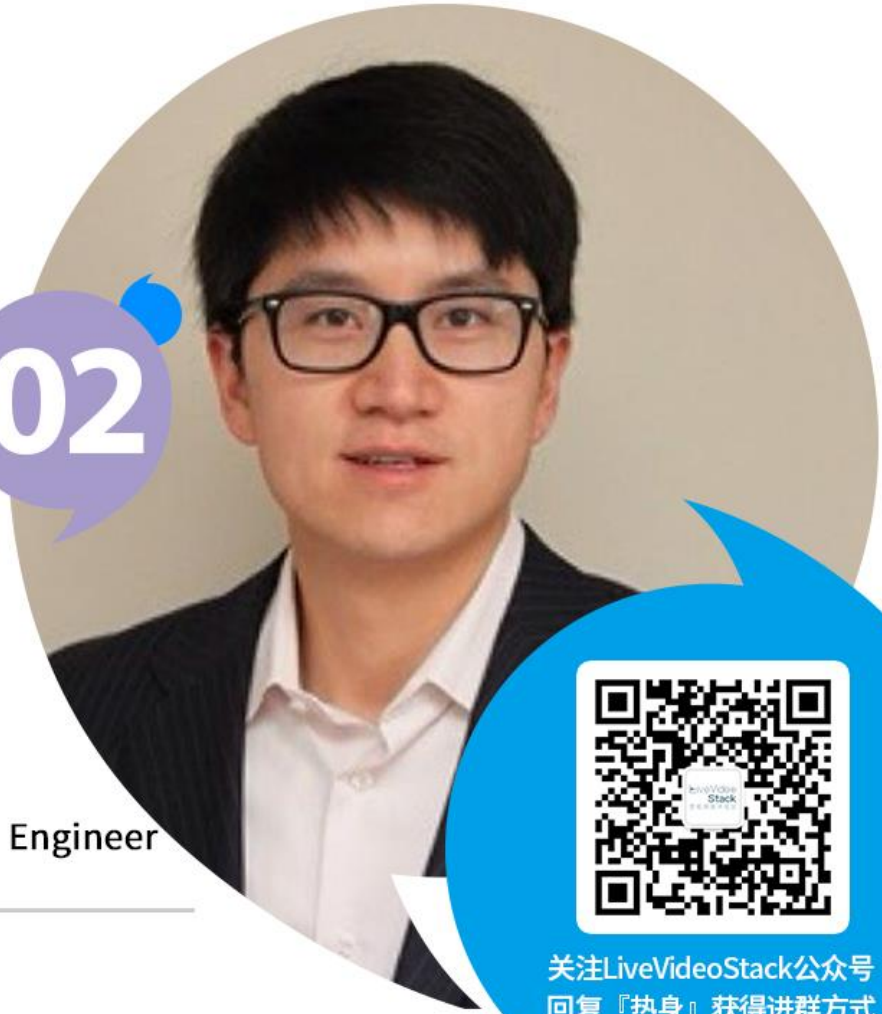
LiveVideo StackCon 音视频技术大会

技术开启新“视”界

Technology Bring New Vision

2018.10.19-20 北京丽亭华苑酒店

LiveVideoStackCon 2018 讲师热身分享会 02



7月5日21:30 准时开始

《互联网直播未来5年走势》

沈悦时 Twitch Principal Research Engineer

关注LiveVideoStack公众号
回复『热身』获得进群方式

LiveVideoStack 音视频技术社区

CSDN