

Interoperability between FFmpeg and GStreamer

Marek Olejnik
molejnik@fluendo.com

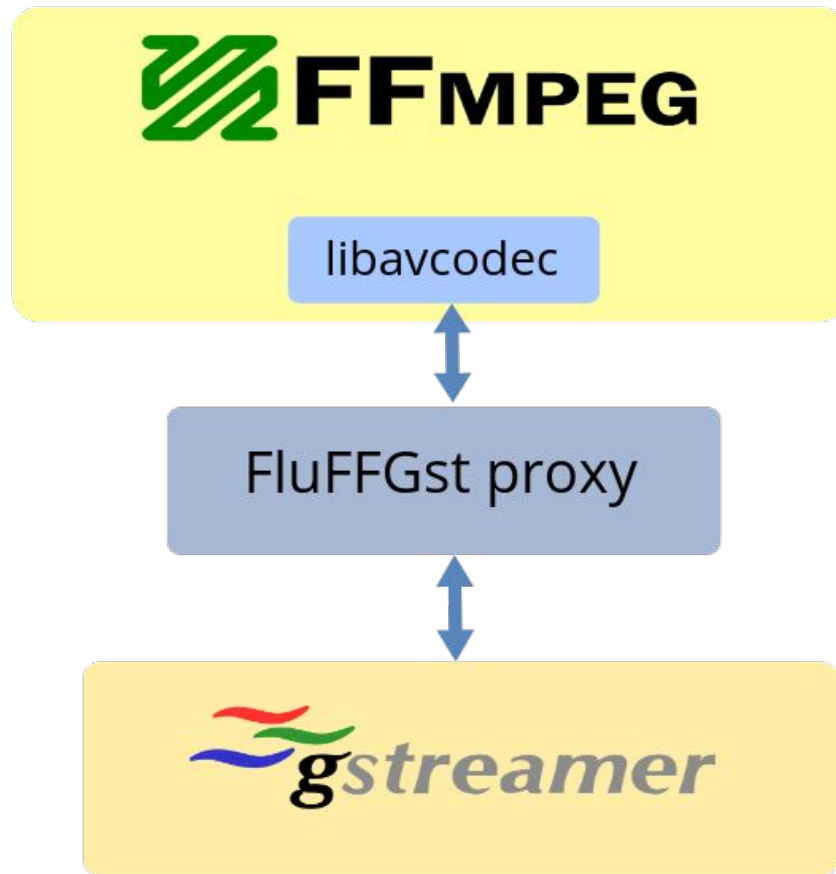
Motivation:

Decode audio/video with Fluendo plugins in applications using FFmpeg API:

- Codec patents
- Performance

Use cases:

- Firefox
- VLC



Registering the decoder

allcodecs.c :

```
/* audio codecs */  
REGISTER_DECODER(LIBFLUFFGST_AAC, libfluffgst_aac);  
REGISTER_ENCDEC (AAC, aac);  
REGISTER_DECODER(AAC_FIXED, aac_fixed);
```

Dependencies

configure:

```
EXTERNAL_LIBRARY_LIST="
```

```
    libfluffgst
```

```
    avisynth
```

```
    ...
```

External library support:

```
--enable-libfluffgst    enable Fluendo decoders [no]
```

```
--enable-avisynth      enable reading of AviSynth script files [no]
```

```
# these are off by default, so fail if requested and not available:
```

```
    enabled libfluffgst    && add_extralibs -lfluffgst
```

```
    enabled avfoundation_indev && { check_header_oc AVFoundation/AVFoundation.h || disable avfoundation_indev; }
```

```
# decoders / encoders
```

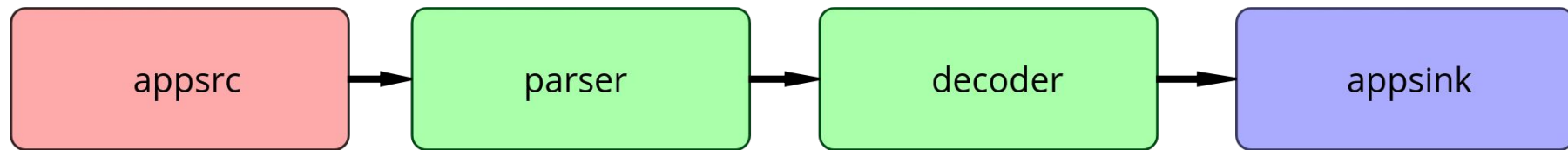
```
libfluffgst_aac_decoder_deps=libfluffgst
```

```
aac_decoder_select="imdct15 mdct sinewin"
```

Defining the decoder

```
AVCodec ff_libfluffgst_aac_decoder = {  
    .name          = "libfluffgst_aac_dec",  
    .long_name     = NULL_IF_CONFIG_SMALL("Fluendo AAC decoder"),  
    .type          = AVMEDIA_TYPE_AUDIO,  
    .id            = AV_CODEC_ID_AAC,  
    .priv_data_size = sizeof(FluFFGstAACDecContext),  
    .init          = libfluffgst_aac_dec_init,  
    .close         = libfluffgst_aac_dec_close,  
    .decode        = libfluffgst_aac_dec_decode,  
    .flush         = libfluffgst_aac_dec_flush,  
    .sample_fmts   = (const enum AVSampleFormat[]) {  
        AV_SAMPLE_FMT_FLTP, AV_SAMPLE_FMT_NONE  
    },  
    .capabilities  = AV_CODEC_CAP_DELAY,  
    .priv_class    = &libfluffgst_aac_dec_class,  
};
```

Pipeline



Difficulties

- Synchronous input-output - asynchronous pipeline
- Mapping `AvCodecContext` to `GstCaps`

<https://fluendo.com/en/developer/blog/>