LET'S CREATE IT

GStreamer as multimedia framework in Android: a new alternative.

CELF Embedded Linux Conference Europe 2010

Benjamin Gaignard

October 27, 2010

ST ERICSSON

Context

- ST-Ericsson U671X Android platform on ARM9 @ 416 MHz with video, audio and graphics dedicated hardware accelerators.
- Multimedia hardware capabilities:
  - Video encode and decode
  - Audio encode and decode
  - Graphics
  - Imaging
  - Camera
- Linux kernel 2.6.29
- GStreamer 0.10.26

GStreamer in Android: why ?

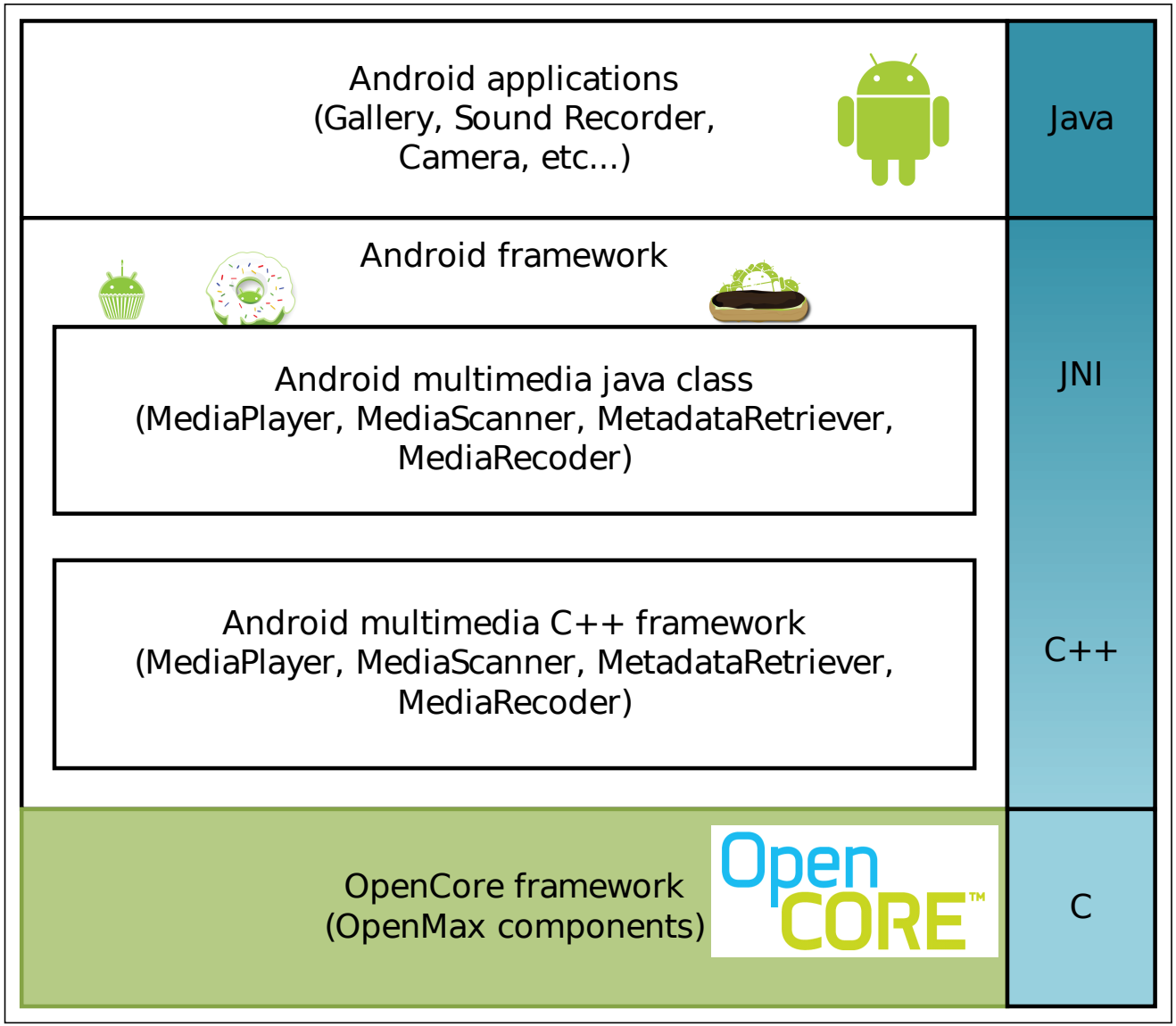
- GStreamer is a mature multimedia framework.
- GStreamer is a flexible framework.
- GStreamer is supported by an active community.
- ST-Ericsson U671X platform was developed and matured since years using GStreamer as multimedia framework with success.
- GStreamer is required by most of Linux framework: MEEGO, Ubuntu
- GStreamer graph based approach is naturally fitting U671X hardware split (1 GStreamer element for 1 HW block)

GStreamer to handle what ?

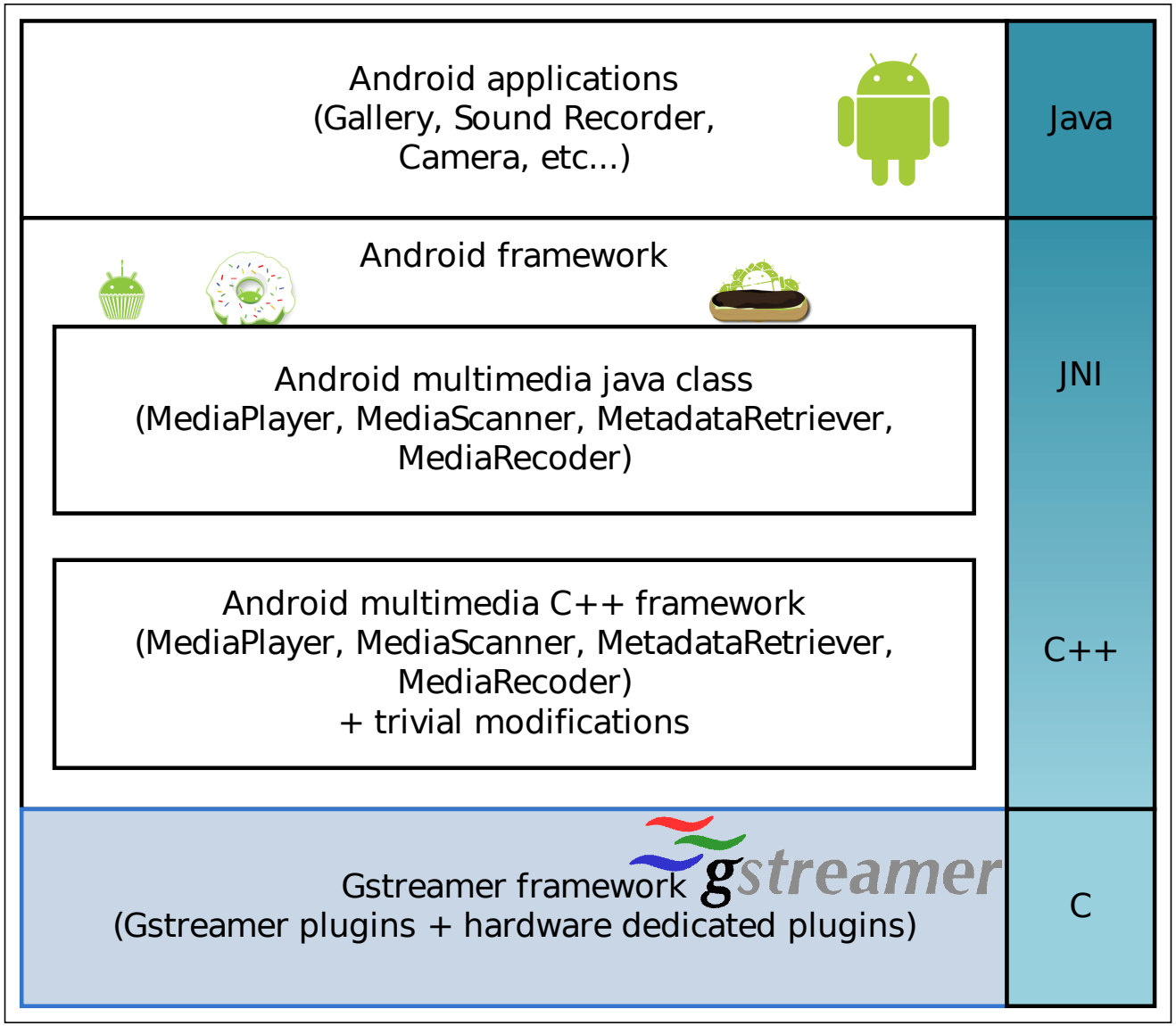
- Multimedia playback: audio/video streams from local storage, streaming or progressive download.
- Multimedia recording: Camera application, Sound Recorder.
- Metadata retrieving: audio/video tags, bitrates, codec information, thumbnails generation.

- Constraints:
  - Do not touch to Android java APIs
  - Follow Android versions (cupcake, donut, éclair ...)
  - Respect internal Android APIs

GStreamer replace OpenCore



GStreamer replace OpenCore



Porting GStreamer on Android, what are the difficulties ?

- Android isn’t built from classical Linux way (no pkg-config, no configure, no makefile…) it is required to adapt GStreamer build process to Android one.
- Glib isn’t provided natively by Android, need to add it.
- All of this have required to rewrite +70 Android makefiles (.mk)
- … but it was also an opportunity to carefully select the embedded elements to optimize GStreamer memory foot print and speed.
- Today U671X android platform embeds 273 elements split in 39 GStreamer plugins.
- Only use dynamic libraries to not break GStreamer (LGPL) and Android (Apache) licenses terms.

MediaPlayer services

- Playbin2 is used to handle playback, streaming and progressive download services
- Playbin2 has been customized to reduce memory consumption, with Collabora Multimedia partnership.
- Use 2 dedicated sinks for audio and video rendering:
  - Audio sink requesting adaptation in Android AudioTrack class.
- Make match GStreamer states (PAUSED, PLAYING …) and events (EOS, SEEK, ASYNC-DONE) to Android MediaPlayer expected states and messages.
- All the complexity to handle local playback, streaming or progressive download is hidden by playbin2.

MediaRecorder services

- Use dedicated GStreamer pipeline with hardware accelerated plugins.
- Only a limited number of codec supported in Android:
  - Video: MPEG4, H263, H264
  - Audio: AMR NB, AMR WB, AAC-LC
- Only one muxer required to handle all recording formats: gppmux
- v4l2src isn’t use as video source: Android Camera class provide the video frames.

MetadataRetriever services

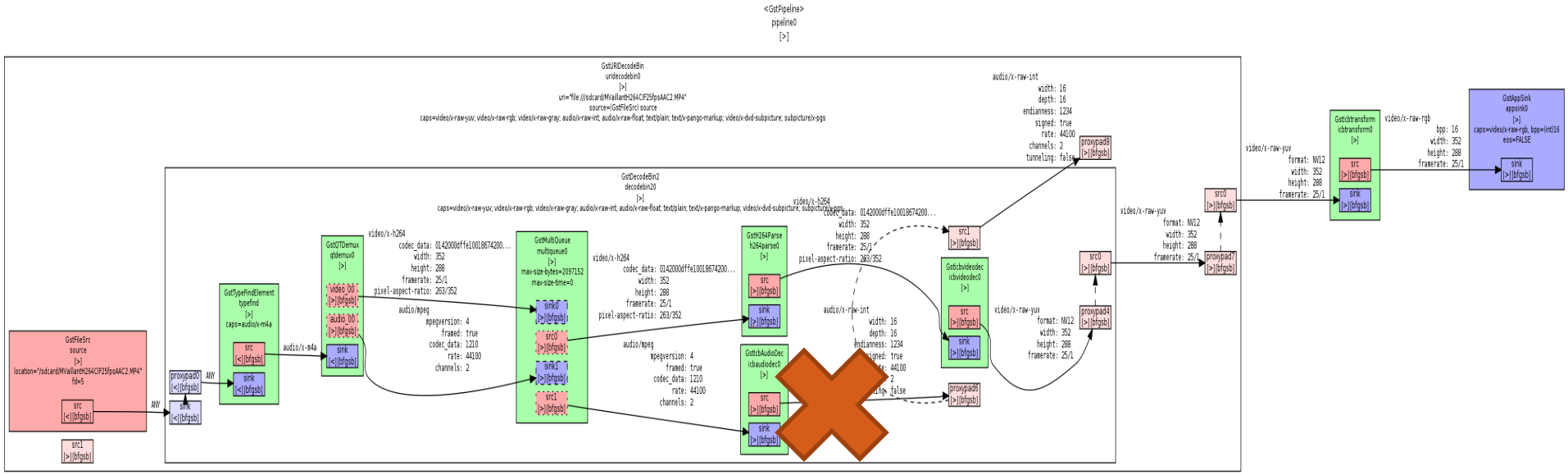
- Thumbnail generation for video files uses hardware accelerator for video decoding and color space conversion (YUV2RGB).
- Android metadata tags are mapped to GStreamer tags to extract: codec, bit rate, album art, ...
- Remove OpenCore implementation from Android MediaScanner class => everything is done by GStreamer.
- Use a simpler graph than playbin2 for better performance and without cpu/mem heavy cost.

  uridecodebin uri=%s ! icbtransform ! appsink caps="video/x-raw-rgb,bpp=16”

- Only video is decoded (not audio) for thumbnail generation.

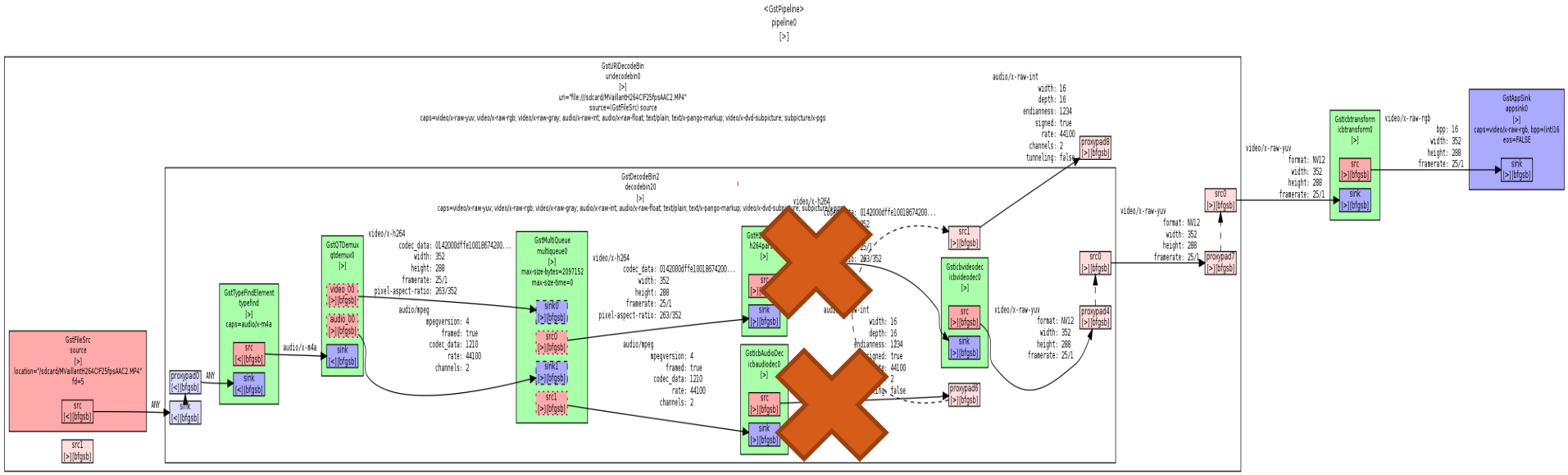
uridecobin optimizations

- For thumbnails generation we don't need to decode audio stream, we use uridecodebin's "autoplug-continue" callback to limit graph building.



uridecobin optimizations (2)

- For metadata retrieving use the same technique but don’t graph video too.
- It saves time at graph building, reduce memory usage, don’t use any hardware resource.



How to cope with Android licensing

- Android is under APACHE license.
- GStreamer is under LGPL v2.1 license.
- The wrapper between Android multimedia library and GStreamer is under ST-Ericsson copyright and is dedicated to U671X hardware platform.
- GStreamer libraries are dynamically linked in Android framework.
- Add NOTICE files in GStreamer to match with Android build process.

Improvements done in GStreamer with Collabora Multimedia support

- avidemux and qtdemux parsing speed up and memory consumption reduction.
- Add “push mode” to avidemux, qtdemux and flvdemux.
- New RTSP buffering mode to handle data burst on mobile networks.
- queue2 ring buffer mode improvement.
- QoS message to detect framedrop.

- All those improvements (+ trivial bugs correction) have being released under LGPLv2.0 and are now available on GStreamer main stream.

U671X-based phones in mass production

ACER betouch E110

ACER betouch E120

ACER betouch E130

HTC Tianyi

ST ERICSSON

Conclusion

- With minimum effort, GStreamer brings to Android:
  - A full open source multimedia framework.
  - Maturity.
  - Additional codecs (VC1, DIVX…) and demuxers (AVI, FLV…)
  - Community support and reactivity.
  - Performance.
  - Evolution (future codec, new streaming protocols, Video Telephony).

Questions and Answers

ST ERICSSON


THANK YOU
ST
ERICSSON