Lessons Learned Transforming Our Video Management System with GStreamer

Bumin Kaan AYDIN
Hüseyin Furkan KIYIKÇI

Gstreamer Conference 2025
London

HAVELSAN

The content of this document is the intellectual property of HAVELSAN. It is forbidden to copy or give all or part of this document to someone else without HAVELSAN's permission

ABOUT US

Personnel
2600+

Subsidiaries
14

Ecosystem Companies
900+

Countries
28

Command Control and Defence Technologies

Simulation and Training Technologies

Information and Communication Technologies

Robotics and Autonomous System

Cyber Security

Change default core library path

Application Development windows

Topics
My posts
My messages
Invite
More

CATEGORIES
General Discussion
Site Feedback
All categories

TAGS
windows
rust
android
python
macos
All tags

JeffZZ Apr 30

Well, I thought it was, as gstreamer has the `GST_PLUGIN_PATH` for these same exact purpose but for plugin libraries, so I assumed there would be something similar for core libraries within gstreamer. Thus, I've decided to ask here.

Reply

bradh May 1

Those core libraries are effectively "gstreamer". They need to be loaded before any environmental variable(s) that they specify could have effect.

Reply

bka May 2

We use NSIS for creating windows aplication setup and with that we can add gstreamer to the system path. In the uninstaller we remove it from the path. We kept the same folder structure as gstreamer installation so in the nsis script this works:
EnVar::SetHKLM
EnVar::AddValue "Path" "path-to-our-app\gstreamer\bin"

Apr 30

9 / 12
May 2

21d

e variables needed to locate the plugins and the plugin
T_PLUGIN_PATH

Quote Copy Quote Share

s in a subdirectory, I would not recommend the
ATH in the installer as recommended by @bka since
ed on that machine. There are 2 possible solutions for

llDirectory with the GStreamer libraries subdir before

or the GStreamer libraries in your subfolder and make
comment(linker, "/manifestDependency:
chitecture='*' version='1.0.0.0' type='win32'

wbtcpip Jul 3

i use SetDllDirectory and i put gstreamer dlls wherever i want. it works fine

Our Video Management System

- Integration with many cameras and NVRs
- Flexible user role and hierarchy management
- User actions are logged in detail
- Camera and NVR health statuses are monitored
- Export videos with custom encryption and transcoding

Live RTSP Streaming

More than 500 different camera models

(some models with different fw versions that behave different)

Network can be lossy – buffering is a must

But PTZ control should be responsive

Video enhancement options including stabilization

Several overlays including camera time and actionable GUI items

uridecodebin --> rtspsrc

- Live555 & ffmpeg based previous solution has matured over the years, there is no camera it cannot stream…
- In the initial tests we couldn’t stream many cameras even though uridecodebins was ok in the test environment
- Rtspsrc and its call back functions was the solution
  - on-sdp
  - before-send
- «force-non-compliant-url» since 1.24.7
- [rtsp @ 00000266936d0b00] CSeq 1 expected, 0 received.

Stream Buffering

Network may not be ideal, packet loss and jitter may happen-> Buffering is needed

When the operator uses PTZ controls visual response time is critical -> Minimum buffering

Trials:

- queue/queue2: max-size-time, min-threshold-time
- Latency
- GstBaseSink: max-lateness, processing-deadline, ts-offset

Solution:

- For H.264 decoder set compliance property to flexible
- Adding pad probes to sink/src of every element in the pipeline and correcting PTS when not valid
- Adjust Decodebin src pad-offset when operator starts using PTZ controls

Image Enhancements and Overlays

d3d11videosink

- draw-on-shared-texture deprecated -> emit-present
- Image enhancements including stabilization -> shaders
- Overlays including camera time: d2d drawing functions

GUI overlays

- Independent Qt5 widgets

Shaders and Overlay Render

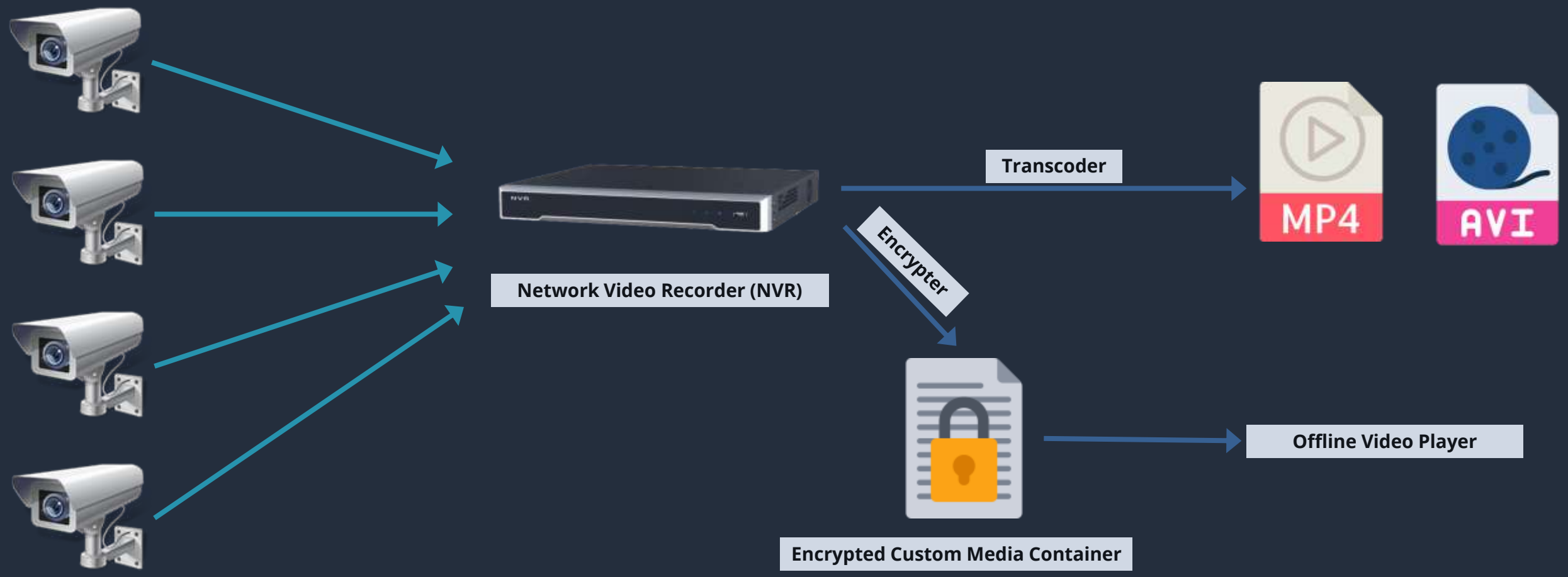
Parameters:

- Brightness
- Contrast
- Saturation
- HueShift
- Gamma

Scene translational correction for stabilization

D2D1 interface & Dwrite: 2D text drawing for overlays - Causes extra drawcall

Export Pipeline



HEVC Slicing Issue

Issues:

- Some HEVC videos have missing slices

Attempts:

1. Accumulating buffers on queue before parsing
2. Changing caps properties
3. Merging 1 secs worth of data in one giant buffer

HEVC Slicing Issue

- Merging data into one buffer
  - Stuttering issues
  - Playback rate issues
  - Trickplay issues
  - Seeking issues

How packets are normally arriving in stream (NAL type):

32 33 34 19 1 1 1

Our packets: 32 34 32 33 34 34 19 1 1 19 33 1 32 19

Reordered: 32 32 32 33 33 33 33 34 34 34 19 19 19 1 1 1

Shaders and Overlay Render

Issues:

- Cannot render on Paused state
- Mimicking present call on paused state by gst_video_overlay_expose() method with timer
- In some video formats video converter can interfer currently loaded shaders.

THANK YOU

www.havelsan.com

The content of this document is the intellectual property of HAVELSAN. It is forbidden to copy or give all or part of this document to someone else without HAVELSAN's permission