STATE OF THE UNION

GStreamer Conference

31 October 2019, Lyon

Tim-Philipp Müller < tim@centricular.com >

INTRO

WHO AM I ?

WHAT IS GSTREAMER ?

Most of you know this, so key points only.

Framework for multimedia processing.

Cross-platform, toolkit agnostic.

Any and all use cases.

Set of libraries and plugins.

Abstract API, very extensible.

We often wrap other libraries.

Low-level API and high-level API:

playbin, encodebin, RTSP server, non-linear editing, WebRTC, VoIP, transcoding, etc.

Integration with other frameworks and projects

e.g. WebKit/Blink, OpenGL, Vulkan
Windows, OS X, Android, iOS

Goal is to adapt to and integrate with
other platforms and frameworks
(inputs, outputs, decoders, DSPs/GPUs..)

SO, WHAT HAVE WE BEEN UP TO ?

RELEASES!

- 1.16: April 2019
- 1.16.1: September 2019 ☹
- goal: 6-monthly(ish) release schedule (still)
- but
- 1.18: Before xmas? Q1 2020?

WE'VE MOVED TO GITLAB!

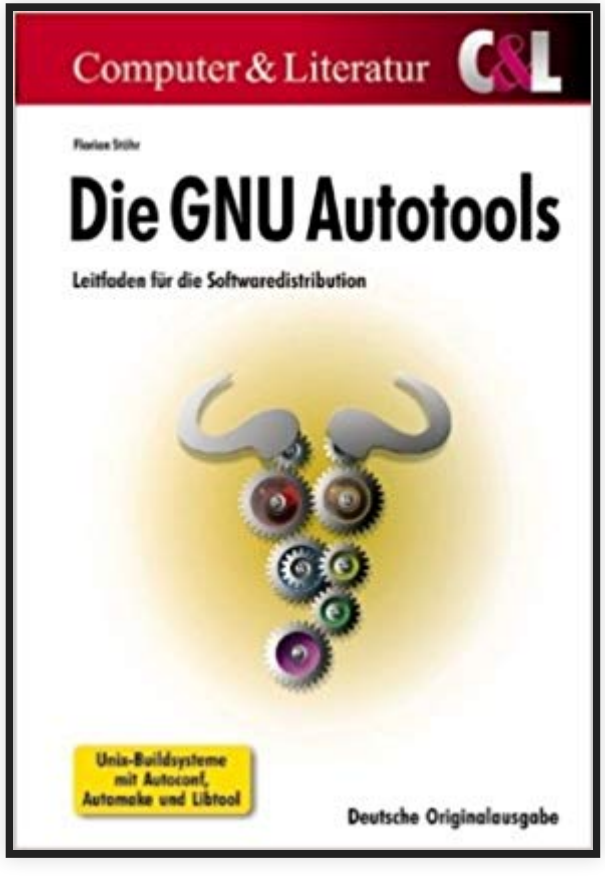
bugzilla.gnome.org -> gitlab.freedesktop.org

WHY GITLAB?

- modern workflows
- better patch review tools
- pre-merge continuous integration!

BUILD SYSTEMS I

Autotools Status: R.I.P. and thanks for all the fish!



BUILD SYSTEMS II

We have switched to Meson!

- what a journey!
- 1.16 ships with both Meson and Autotools support
- 1.18 will be Meson only

DOCUMENTATION

We have switched to hotdoc!

- both API docs and plugin docs
- no more gtk-doc
- unified docs module
- gst-docs release tarballs

SOME THINGS THAT LANDED IN 1.16 AND AFTER ...

WEBRTC

- "How do I stream to my web browser?"
- Low latency, works pretty much everywhere.
- webrtcbin + gstwebrtc library
- Different codecs
- Leverage all of GStreamer:
  transmuxing, hw-acceleration etc.

WEBRTC IMPROVEMENTS

- data channels for peer-to-peer communication based on SCTP
- BUNDLE support
- support for multiple TURN servers
- initial support for on-the-fly stream addition/removal (with some limitations)

AV1 VIDEO CODEC SUPPORT

Royalty-free next-generation video codec!

- Matroska and QuickTime/MP4 container support
- more options and formats for the AOMedia encoder
- superfast libdav1d decoder in gst-plugins-rs
- rav1e encoder in gst-plugins-rs
- MPEG-TS mappings coming soon? SVT-AV1?

HARDWARE-ACCELERATED VIDEO ENCODERS AND DECODERS

- video4linux
- nvidia: nvdec, nvenc
- intel msdk
- gstreamer-vaapi
- gst-omx
- Future: DXVA2? AMD? Vulkan?

NVCODEC: HARDWARE-ACCELERATED VIDEO ENCODING/DECODING FOR NVIDIA

- nvdec + nvenc now merged into one single plugin
- per-GPU elements with capability check
- dynamic library loading (for easier distribution)
- new codecs, more pixel formats, performance improvements

CLOSED CAPTIONS AND OTHER ANCILLARY DATA

- closed caption meta + combiner/extractor elements
- AFD + BAR metas

VIDEO AGGREGATOR NOW IN -BASE

- compositor + OpenGL mixer also in -base now
- compositor performance improvements:
  - don't draw background that's obscured
  - avoid unnecessary work for single input

BROWSER SOURCE

- Capture a web page as a video stream!
- wpesrc: a WebKit WPE-based web browser source element
- OpenGL acceleration
- Audio coming soon?

TRANSCODING

New GstTranscoder library in -bad!

- abstraction for transcoding tasks
- give it a spin and let us know how the API works for you!

LATENCY IMPROVEMENTS

- payloaders, depayloaders, parsers, decoders, encoders
- latency tracer improvements
- tsdemux: latency now configurable so can force lower latency than 700ms
- coming up: subframe support in encoders + decoders

STREAMING

- new AVTP elements for Audio/Video Bridging
- new plugin for Reliable Internet Stream Transport (RIST)
  - RIST TR-06-1 Simple Profile
  - built on RTP
  - bonding support (higher reliability or higher bandwidth)

RTP

- rtpitterbuffer / rtp stack performance improvements
- rtpsrc/rtpsink RTP streaming elements that can be used and configured via URIs

MPEG-TS MUXING IMPROVEMENTS

Improvements for broadcasting use cases:

- tsmux ported to aggregator
- new atscmux subclass for ATSC
- constant bitrate (CBR) support
- spec compliance fixes

RTSP IMPROVEMENTS

- ONVIF trickmode support in server + client
- TCP backpressure fixes in rtsp-server
- TCP interleaved performance improvements
- device provider to discover RTSP cameras

VULKAN

- loads of improvements, work continuing
- esp. useful or macOS/iOS where OpenGL has been deprecated (via MoltenVK)
- also have win32 support now

HDR VIDEO

High Dynamic Range Video:

- requires new signalling and plumbing
- muxers, demuxers, parsers
- new utility API
- lots of new high bit depth pixel formats
- still some work to do

WINDOWS IMPROVEMENTS

Too many to list!

- binary packages built with MSVC
- MSVC build checks on the CI
- d3d11videosink based on Direct3D11 incl. HDR10 support
- improved debug logging and printing
- gst-play-1.0 keyboard controls now work on win32

WINDOWS IMPROVEMENTS II

- UWP support (wip)
- leak tracer snapshotting / live object querying is now possible on Windows
- soon: d3dvideosink overlay composition support

VIDEO EDITING

- GStreamer Editing Services library:
  - marker lists
  - OpenTimeline support
  - gesdemux: play back edit list in playbin/decodebin
  - many other improvements
- accurate seeks now with correct audio lead-in for lossy formats

SPLITMUXSINK

- `muxer-pad-map` property, useful for tsmux
- can mux more than one video stream now

THE FUTURE ...

SCTE-35

Coming soon:
SCTE-35 support in MPEG-TS library and muxer/demuxer

DASH CREATION

- new dash sink element in the pipeline!

MORE WINDOWS IMPROVEMENTS

- direct3d gst lib?
- Windows Media Foundation elements?
- native cerbero builds on Windows (also for nightlies)

PERFORMANCE OPTIMISATIONS..

Everywhere, of course.

Lots of things in the pipeline to improve performance, latency and memory usage.

WHAT ELSE ?

RTMP

- new RTMP elements?

RUST

Let's talk about Rust! (again)

"Fast, safe and productive - pick three."

There will definitely be more Rust in our future.

RUST

Perfect language for us technically.

Excellent C compatibility.

Fantastic community.

Superb ecosystem.

Lots of positive experience gained in the last year(s).

GStreamer Rust bindings + plugin modules now officially upstream!

RUST (CLARIFICATION)

No plans to switch to it in the short run.

No plans to make it a hard dependency in core components.

We can do a lot without breaking GStreamer backwards compat.

But let's play with it, experiment, and gain more experience.

Need to make sure it works for all our users!

Something for the longer term.

GSTREAMER RUST BINDINGS

Should be in really good shape now!

Much more complete.

Many releases, many new users.

Subclassing/plugins possible for all important types.

Write more examples and plugins in Rust!

Let's ship Rust plugins as part of our binaries!

--> Sebastian's talk tomorrow

ROOM FOR IMPROVEMENT ?

MAKE IT EASIER TO WRITE "SIMPLE SERVERS" (STILL)

- souphttpserversink
- rtspserversink

STREAM API

- implement in all / more demuxers !
- support stream deactivation
- scalable streams
- implement in adaptive streaming demuxers

KLV

Need a good KLV story!

- metas
- streams
- splitter/combiner
- finish MPEG-TS support
- Matroska muxing

HARDWARE-ACCELERATED VIDEO DECODING/ENCODING

- make less painful, more automatic on request (esp. encoding)
- make more robust -> need CI

DO MORE WITH TRACERS!

- e.g. better latency tracing
- more powerful debugging

WHAT ELSE SHOULD BE ON OUR RADAR?

- SDI-over-IP standards (NDI source now on github)
- better OpenCV integration
- VR
- Video Analytics: one framework to rule them all?
- stabilise GstPlayer API

SHOULD WE HAVE A PROPER FORUM?

Discourse?

Not just mailing lists and IRC.

GNOME have set up an instance.

Ideally we'd have a freedesktop-wide one.

THAT'S ALL FOLKS

THANK YOU AND ENJOY THE CONFERENCE !

QUESTIONS? COMMENTS?