A New RTSP Source Element Written in Rust

rtspsrc2

GStreamer Conference 2024

Nirbheek Chauhan

https://nirbheek.in

Who Am I?

Platform Stuff, Rust, WebRTC, RTSP, Meson, Cerbero, CI

Whatever needs doing

Everywhere and also Nowhere

gstreamer

Centricular

What's wrong with rtspsrc?

- Created in 2005, simple design
  - At the time, neither live nor VOD

What's wrong with rtspsrc?

- Created in 2005, simple design
    - At the time, neither live nor VOD
    - DESCRIBE SETUP PLAY PAUSE TEARDOWN

gstreamer

Centricular

What's wrong with rtspsrc?

- Created in 2005, simple design
    - At the time, neither live nor VOD
    - DESCRIBE SETUP PLAY PAUSE TEARDOWN
    - READY → PAUSED ⇒ DESCRIBE+SETUP+PLAY aka open()

gstreamer

Centricular

What's wrong with rtspsrc?

- Created in 2005, simple design
    - At the time, neither live nor VOD
    - DESCRIBE SETUP PLAY PAUSE TEARDOWN
    - READY→PAUSED ⇒ DESCRIBE+SETUP+PLAY aka open()
    - PAUSED→READY ⇒ PAUSE

What's wrong with rtspsrc?

- Created in 2005, simple design
    - At the time, neither live nor VOD
    - DESCRIBE SETUP PLAY PAUSE TEARDOWN
    - READY→PAUSED ⇒ DESCRIBE+SETUP+PLAY aka `open()`
    - PAUSED→READY ⇒ PAUSE
    - pad deactivation ⇒ TEARDOWN aka `close()`

What's wrong with rtspsrc?

- Ties server state to client (pipeline!) state

What's wrong with rtspsrc?

- Ties server state to client (pipeline!) state
    - Multicast playback

What's wrong with rtspsrc?

- Ties server state to client (pipeline!) state
    - Multicast playback
    - PAUSE is sent on error, immediately followed by a TEARDOWN

What's wrong with rtspsrc?

- Ties server state to client (pipeline!) state
    - Multicast playback
    - PAUSE is sent on error, immediately followed by a TEARDOWN
    - Flushing seeks glitch & **deadlock**

What's wrong with rtspsrc?

- Ties server state to client (pipeline!) state
    - Multicast playback
    - PAUSE is sent on error, immediately followed by a TEARDOWN
    - Flushing seeks glitch & **deadlock**
    - Incorrect communication of intent

What's wrong with rtspsrc?

- Ties server state to client (pipeline!) state
    - Multicast playback
    - PAUSE is sent on error, immediately followed by a TEARDOWN
    - Flushing seeks glitch & **deadlock**
    - Incorrect communication of intent
- Some of these have workarounds that every app must implement, others are unfixable

gstreamer

Centricular

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command

gstreamer

Centricular

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added
  - Both Client→Server and Server→Client

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added
  - Both Client→Server and Server→Client
  - Collision ⇒

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added
  - Both Client→Server and Server→Client
  - Collision ⇒ **messages are dropped silently**

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
    - Fixed with a "cmd loop queue"
    - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added
    - Both Client→Server and Server→Client
    - Collision ⇒ **messages are dropped silently**
- 2021: Fix requires large code changes

gstreamer

Centricular

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added
  - Both Client→Server and Server→Client
  - Collision ⇒ **messages are dropped silently**
- 2021: Fix requires large code changes
  - Not tenable in rtspsrc

What's wrong with rtspsrc?

- 2011: open/pause/play/close are async but racy
  - Fixed with a "cmd loop queue"
  - Only one pending command
- 2018: New commands SET/GET_PARAMETER are added
  - Both Client→Server and Server→Client
  - Collision ⇒ **messages are dropped silently**
- 2021: Fix requires large code changes
  - Not tenable in rtspsrc
  - Workaround: add some sleeps in the application 🫠

What's wrong with rtspsrc?

What's wrong with rtspsrc?

- Some re-organising *was* done
    - `gstreamer-rtsp-1.0` & `gstreamer-sdp-1.0`

What's wrong with rtspsrc?

- Some re-organising *was* done
  - `gstreamer-rtsp-1.0` & `gstreamer-sdp-1.0`
- No major changes, only incremental

What's wrong with rtspsrc?

- Some re-organising *was* done
  - `gstreamer-rtsp-1.0` & `gstreamer-sdp-1.0`
- No major changes, only incremental
- RTSP commands & SDP from the network parsed in C

What's wrong with rtspsrc?

- Some re-organising *was* done
  - `gstreamer-rtsp-1.0` & `gstreamer-sdp-1.0`
- No major changes, only incremental
- RTSP commands & SDP from the network parsed in C
- RTSP 2.0 is impossible
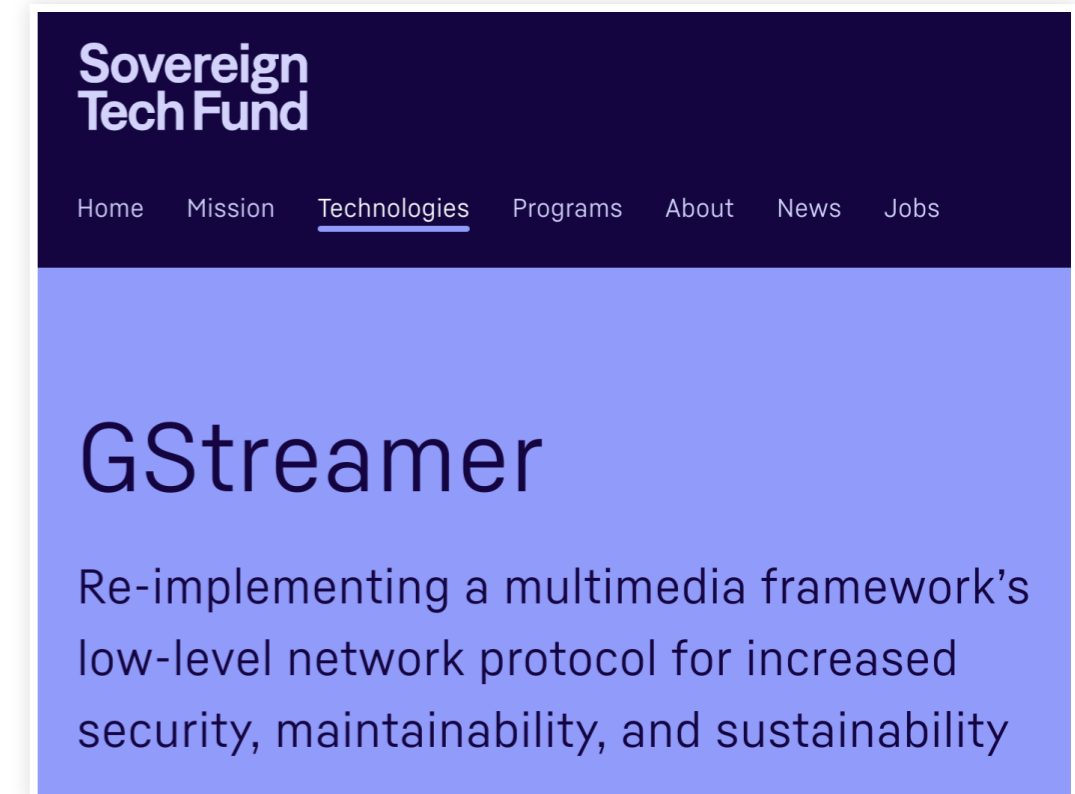
What's wrong with rtspsrc?

- Some re-organising *was* done
  - `gstreamer-rtsp-1.0` & `gstreamer-sdp-1.0`
- No major changes, only incremental
- RTSP commands & SDP from the network parsed in C
- RTSP 2.0 is impossible
- End-Of-Life

gstreamer

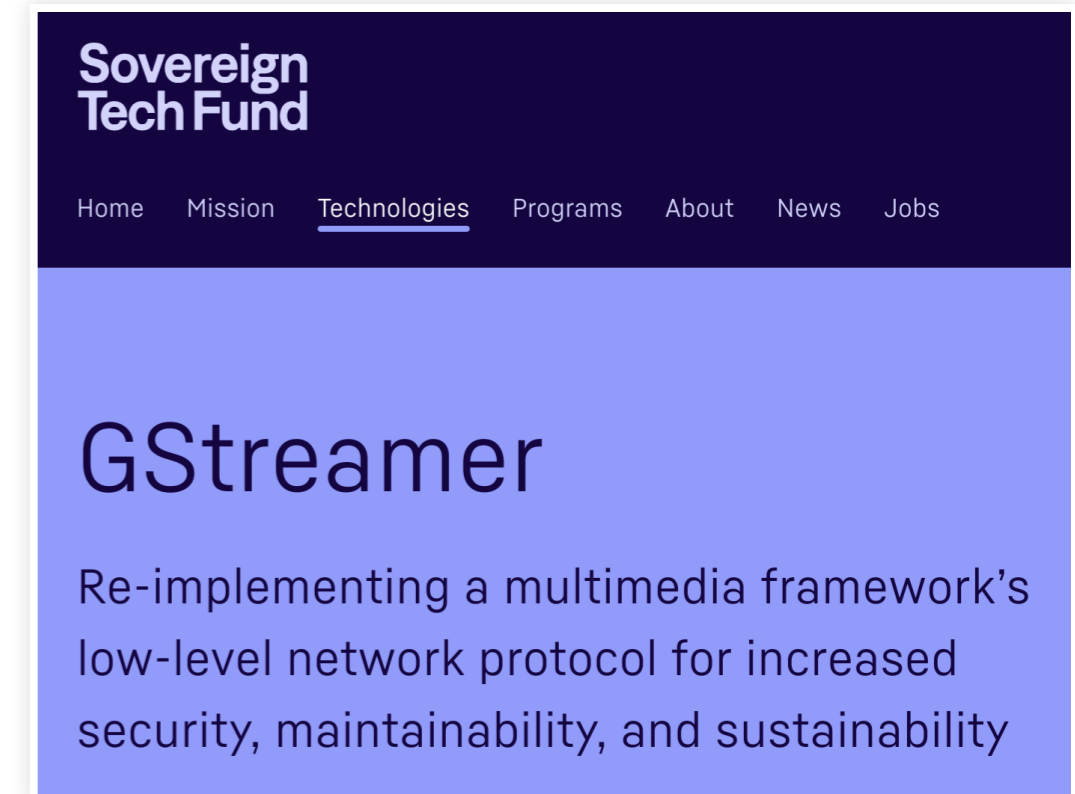Centricular

So we wrote something new

So we wrote something new

Sovereign Tech Fund

Sovereign
Tech Fund

Home   Mission   **Technologies**   Programs   About   News   Jobs

GStreamer

Re-implementing a multimedia framework's low-level network protocol for increased security, maintainability, and sustainability

gstreamer   Centricular

So we wrote something new

Sovereign Tech Fund

"GStreamer RTP sessions in Rust" by Matthew Waters

Sovereign Tech Fund

Home   Mission   Technologies   Programs   About   News   Jobs

GStreamer

Re-implementing a multimedia framework's low-level network protocol for increased security, maintainability, and sustainability
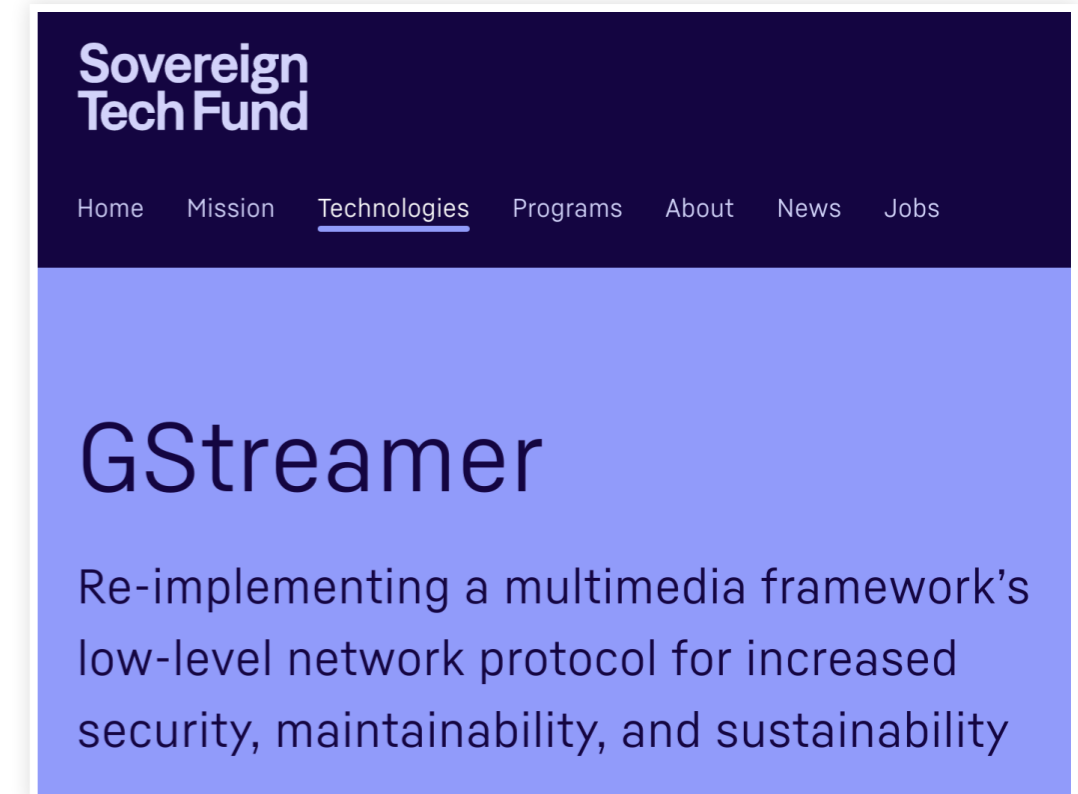
So we wrote something new

Sovereign Tech Fund

"GStreamer RTP sessions in Rust" by Matthew Waters

- RTP send and recv bins
- rtpjitterbuffer
- payloaders, depayloaders
- New RTSP source element

Sovereign Tech Fund

Home   Mission   Technologies   Programs   About   News   Jobs

GStreamer

Re-implementing a multimedia framework's low-level network protocol for increased security, maintainability, and sustainability

gstreamer

Centricular

rtspsrc2: Features

- Live streaming N audio and N video, with RTCP-based A/V sync
- Lower transports: TCP, UDP, UDP-Multicast
- RTP, RTCP SR, RTCP RR
- OPTIONS DESCRIBE SETUP PLAY TEARDOWN
- Supports both the old `rtpbin` and the new `rtpbin2` rust rewrite
  - Set USE_RTPBIN2=1 to use `rtpbin2`
- Lower transport selection **and** priority (NEW!)
  - Even supports different transport for each SETUP call
  - F.ex., KLV could be sent over TCP and the rest over UDP
- `port-start` instead of `port-range`

rtspsrc2: Characteristics

- `rtsp-types` and `sdp-types` crates
- `rtpbin2` ⇒ all reading, parsing, writing, etc is done in Rust
  - Except codec parsing and demuxing
- Async Rust, multi-threaded Tokio
  - Simpler and decoupled, easy to follow
- Network I/O
  - Does not use `udpsrc`, `udpsink`
  - Can be made pluggable or even I/O-less

rtspsrc2: Characteristics

- appsink + appsrc + ghostpads
    - Simplified event handling, startup/shutdown
    - Expose pads early, sparse streams
- Performance on-par with `rtspsrc`
- Tested against live555 and gst-rtsp-server
- Most features have not been implemented yet
- Plan is to replace rtspsrc

rtspsrc2: Low-Hanging Fruit

- Credentials support
- TCP over TLS support
- NAT hole punching
- SET_PARAMETER & GET_PARAMETER support
- Option to reconnect TCP control on disconnect
- Missing configuration properties:
  - `latency` & `do-rtx`
  - `do-rtcp`
  - `iface`
  - `user-agent`

gstreamer

Centricular

rtspsrc2: Giraffe Food

- Stream Collection API
    - Live streams first, VOD later
- SRTP support, incl. all key exchange mechanisms
    - Parsing more SDP attributes
- Control and media flow tunnelling over HTTP
- HTTP and SOCKS proxy support
- Make TCP connections optional when using UDP transports
- Explicit media clock sync support, such as RFC7273

gstreamer

Centricular

rtspsrc2: Re-Forestation

- RTSP 2.0 support
  - PLAY_NOTIFY, Media-Properties, End-Of-Stream, Scale-Change, etc
  - Needs an accompanying RTSP 2.0 server implementation
- Video-On-Demand support, both RTSP 1.0 and 2.0
  - PAUSE, Seeking, Looping
- ONVIF backchannel support
- ONVIF trick modes support
- ...

gstreamer

Centricular

rtspsrc2: Help Needed

rtspsrc2: Help Needed

The main selling-point of the old rtspsrc is the IP camera compatibility that has been built-up over the years. Matching that is absolutely a necessity.

rtspsrc2: Help Needed

The main selling-point of the old rtspsrc is the IP camera compatibility that has been built-up over the years. Matching that is absolutely a necessity.

Please give me access to your IP cameras either using a VPN (tailscale is a good option), or you can ship to me IP cameras that you want to get rid of.

rtspsrc2: Help Needed

The main selling-point of the old rtspsrc is the IP camera compatibility that has been built-up over the years. Matching that is absolutely a necessity.

Please give me access to your IP cameras either using a VPN (tailscale is a good option), or you can ship to me IP cameras that you want to get rid of.

You will thank me in 3-5 years when regulatory bodies force everyone to move to memory-safe languages for government contracting. 🙃

gstreamer

Centricular

rtspsrc2: Help Needed

The main selling-point of the old rtspsrc is the IP camera compatibility that has been built-up over the years. Matching that is absolutely a necessity.

Please give me access to your IP cameras either using a VPN (tailscale is a good option), or you can ship to me IP cameras that you want to get rid of.

You will thank me in 3-5 years when regulatory bodies force everyone to move to memory-safe languages for government contracting. 🙃

It's a fun project for me, and everyone benefits. Please get in touch!

nirbheek@centricular.com

Thanks!!