Liquidshop 3

IoT based Broadcast media capture

Lesson learned trying to build a remote managed IoT Broadcast capture and recording under bad internet.

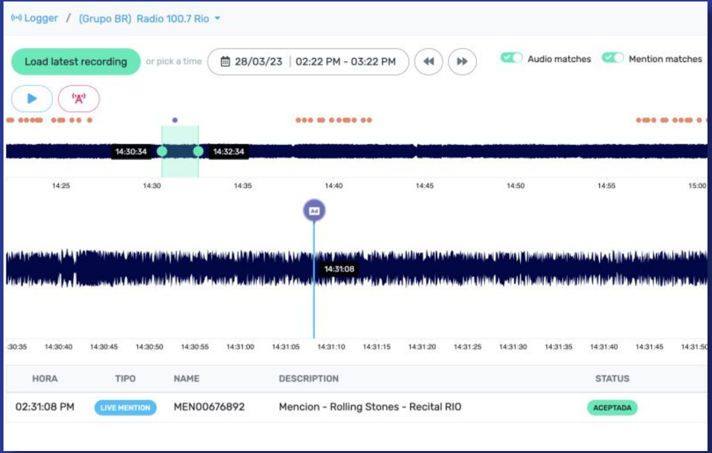
Alejandro Ferrari - 2023/05/30

MEDIAINBOX

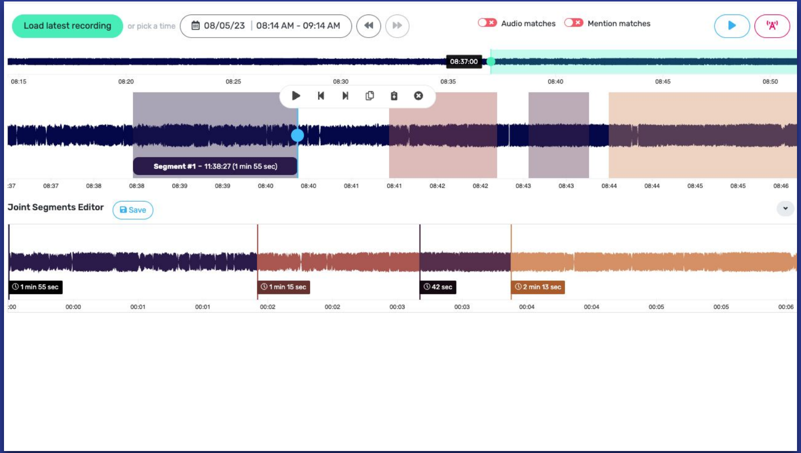
Liquidshop 3

About Us

We are an startup from Buenos Aires, Argentina, that provide services all over the world. With focus on Technology Innovation for Broadcast Media Groups, one of our main products is Cloud Logger, based on that, we provide a suite of tools from Auditing to Media clipping



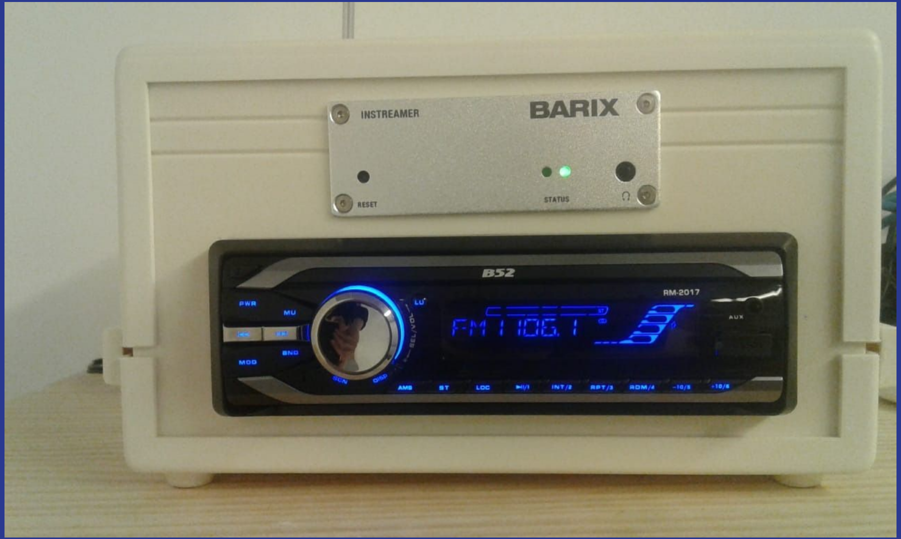
HORA	TIPO	NAME	DESCRIPTION	STATUS
02:31:08 PM	LIVE MENTION	MEN00676892	Mencion - Rolling Stones - Recital RIO	ACEPTADA



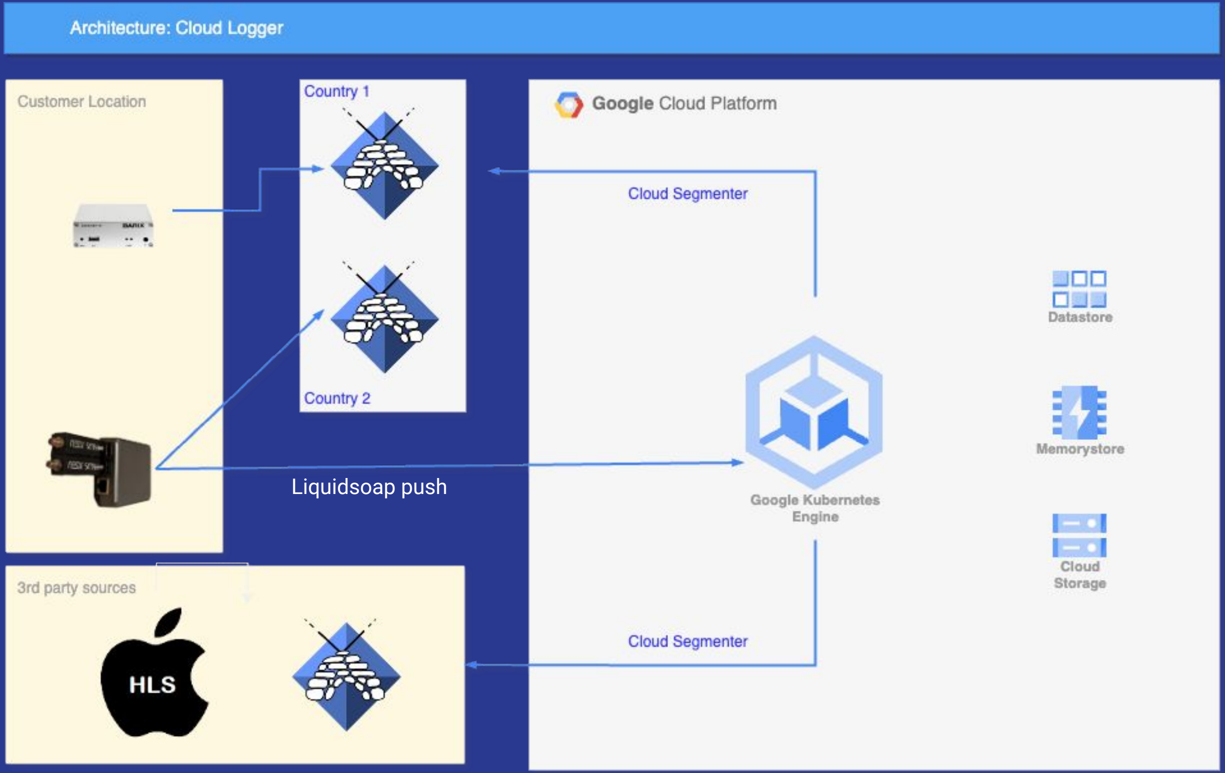
MEDIAINBOX

How the history beginning…

Our first version was capture the audio from the cloud, we still do that, but one day in 2017… one of our customers sent this picture to us, and when we ask for an explanation, they tell us an interesting history…



what is Cloud Logger?

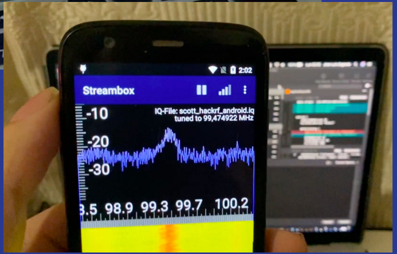


In response to that picture we build...

We started with 3 and later we upgrade to 4

In response to that picture we build...

many versions... inclusive an Android version...



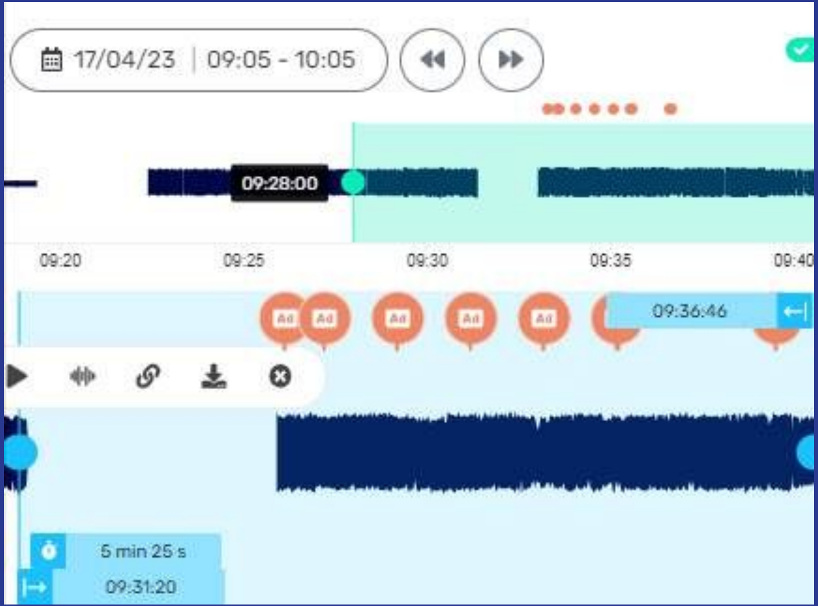
Challenges

Ingest is the most critical part of our product, and is the most hard to manage and implement.

- We need to provide a safe method, that allow to our customers send the live streaming signal to our SaaS infrastructure, and doesn't matter if internet is down, or overloaded, we can't lost a single second of audio
- We capture from Broadcast signals, AM / FM mostly, for that reason we need hybrid infrastructure, run capture around the world where the signal is distributed
- Remote management and support
- OTA Upgrades

Liquidshop 3

Challenges



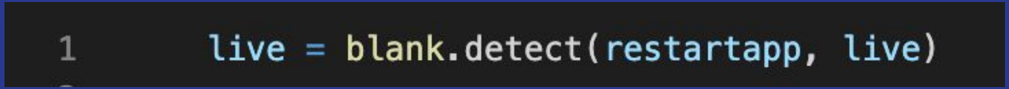
MEDIAINBOX

Lesson 1

We choose why there was, easy to find and cheaper, but after a few months running the MicroSD start to fail, the first version don’t do recording of audio, only capture and push to icecast servers on the cloud… that MicroSD take more time to fail, but fail soon or later…
With the second version, we start to recording internally and here start the nightmare!!! Slow write and SD Corruption made the capture fail a lot with large numbers of devices every day some was down.

Lesson 2

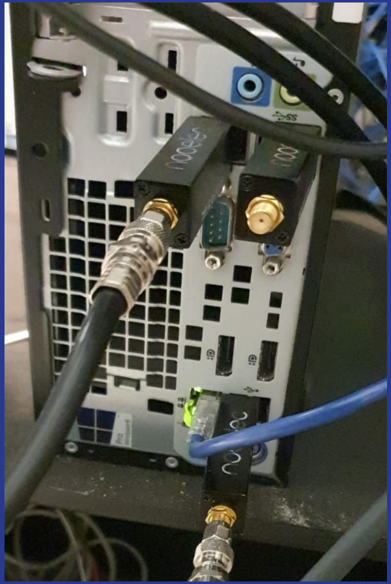
With the upgrade to a new issue arrive, USB power was not enough to maintain the RDS devices stable, and when USB bridge go down, the only way to fix them, is reboot or inclusive in some cases detach the device physically, that in a remote managed encoder is a big issue. One way to handle that was use Silence detector to automatically reboot the device. (no good options but worst was be sleeping and loose hours of audio capture)



```
        live = blank.detect(restartapp, live)
```

Hardware Upgraded - Current version

But problems with Storage (MicroSD) and Power made us to replace with refurbished servers / pcs.

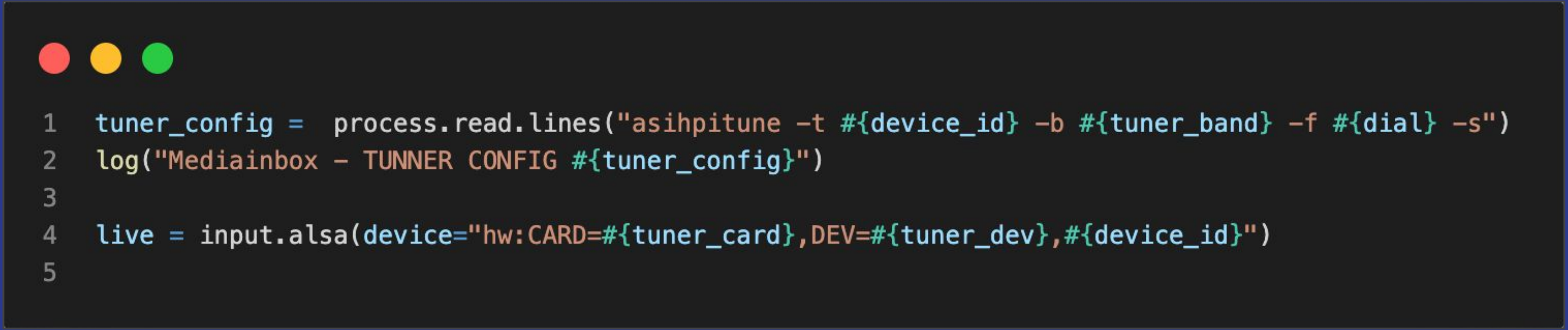


Professional hardware, all demodulation is internal by hardware

Entry Level hardware, all demodulation is by software

Liquidsoap snippets

AudioScience Profesional card using ALSA input

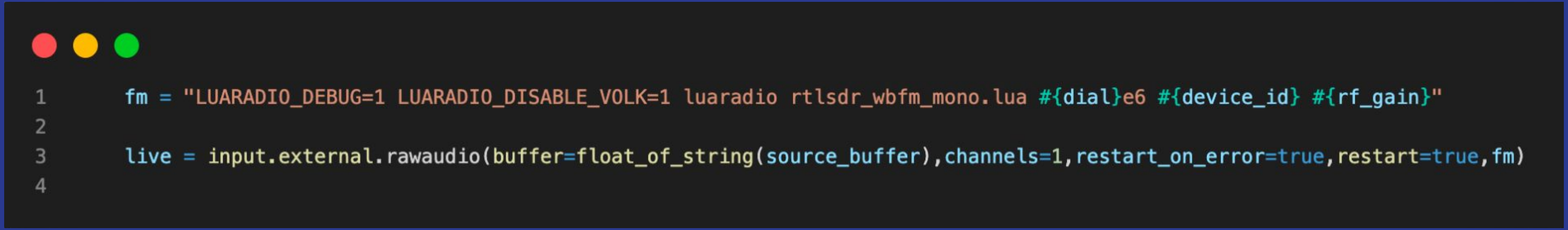


```
tuner_config =  process.read.lines("asihpitune -t #{device_id} -b #{tuner_band} -f #{dial} -s")
log("Mediainbox - TUNNER CONFIG #{tuner_config}")

live = input.alsa(device="hw:CARD=#{tuner_card},DEV=#{tuner_dev},#{device_id}")

```

SDR Input using LuaRadio over input.external.rawaudio



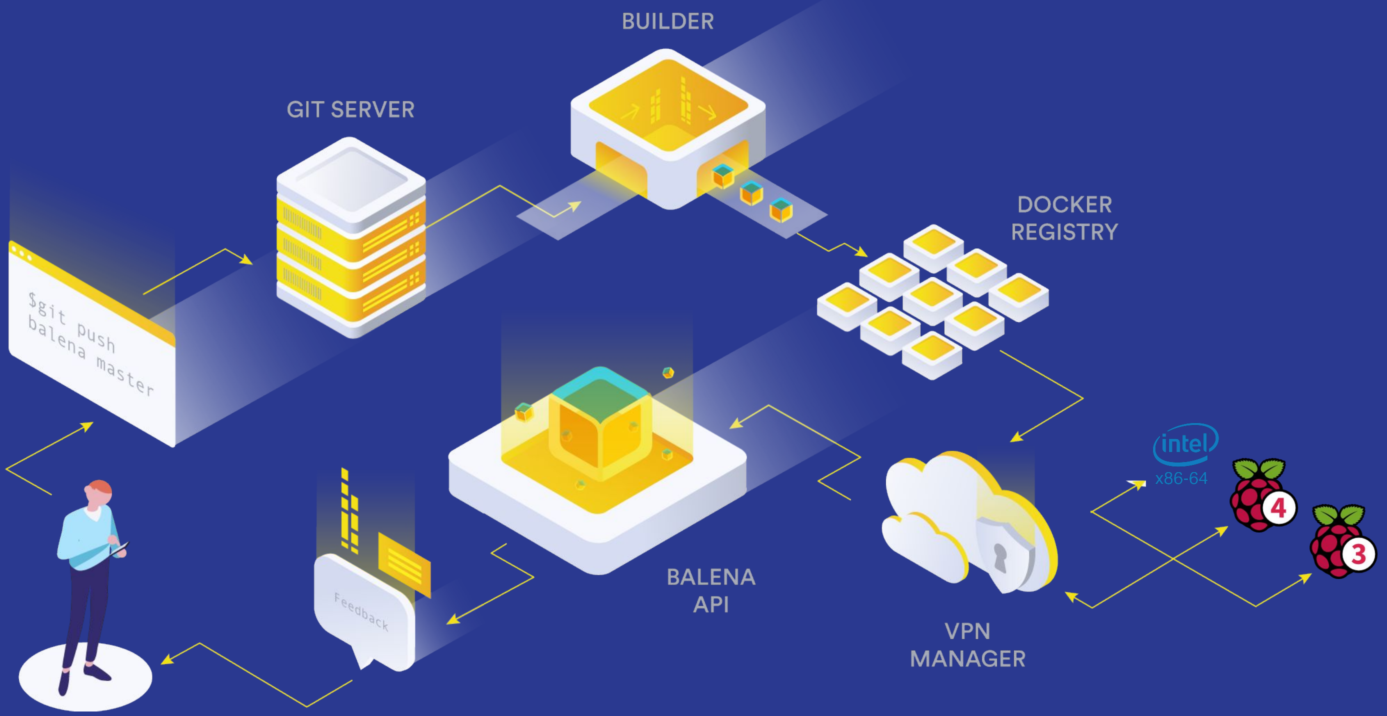
```
    fm = "LUARADIO_DEBUG=1 LUARADIO_DISABLE_VOLK=1 luaradio rtlsdr_wbfm_mono.lua #{dial}e6 #{device_id} #{rf_gain}"

    live = input.external.rawaudio(buffer=float_of_string(source_buffer),channels=1,restart_on_error=true,restart=true,fm)

```

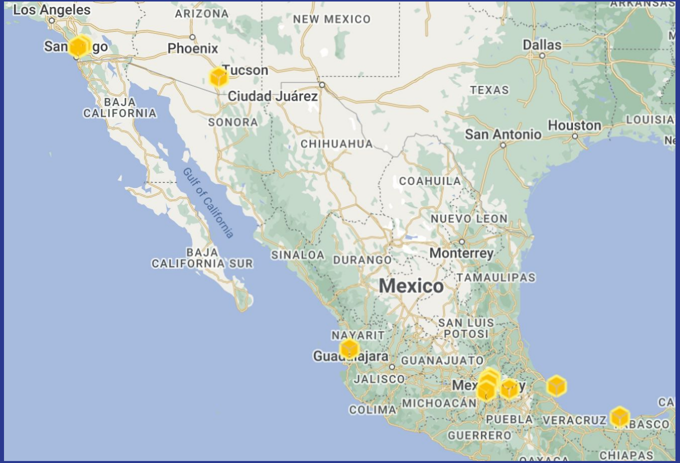
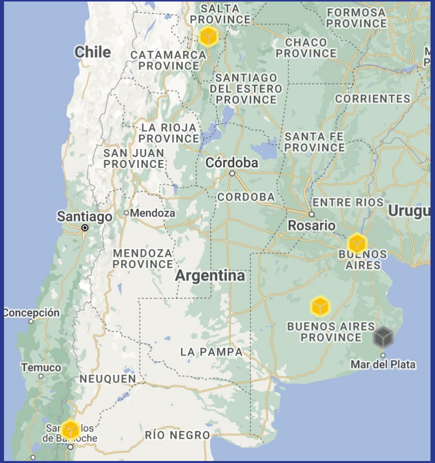
Liquidshop 3

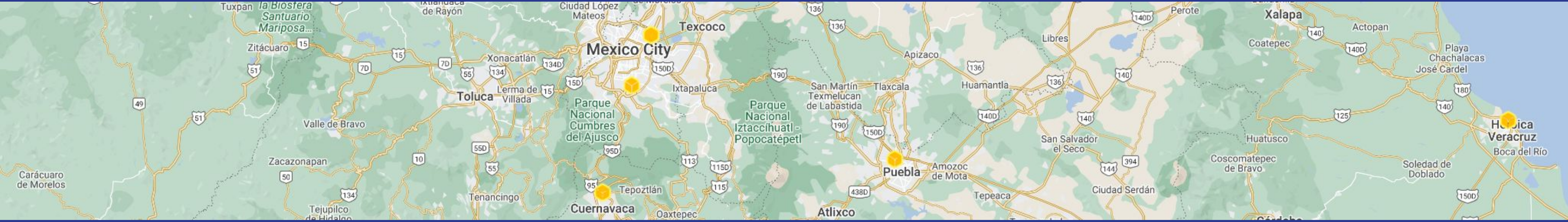
Build and Deploy



MEDIAINBOX

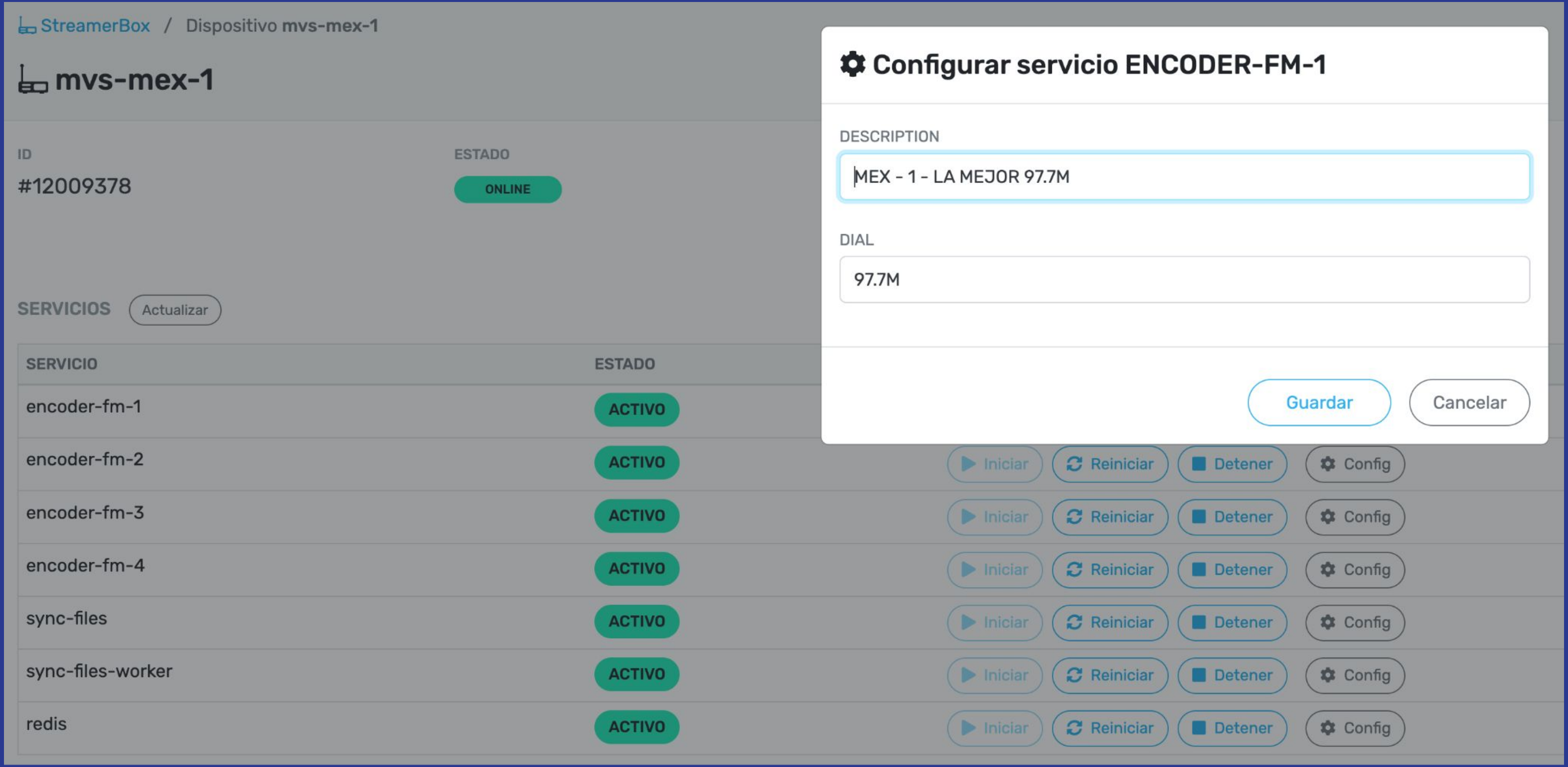
Distributed Deployments



Liquidshop 3

Customer View


StreamerBox / Dispositivo **mvs-mex-1**

mvs-mex-1

ID
#12009378

ESTADO
ONLINE

SERVICIOS Actualizar

SERVICIO	ESTADO	
encoder-fm-1	ACTIVO	
encoder-fm-2	ACTIVO	Iniciar · Reiniciar · Detener · Config
encoder-fm-3	ACTIVO	Iniciar · Reiniciar · Detener · Config
encoder-fm-4	ACTIVO	Iniciar · Reiniciar · Detener · Config
sync-files	ACTIVO	Iniciar · Reiniciar · Detener · Config
sync-files-worker	ACTIVO	Iniciar · Reiniciar · Detener · Config
redis	ACTIVO	Iniciar · Reiniciar · Detener · Config

Configurar servicio ENCODER-FM-1

DESCRIPTION
MEX - 1 - LA MEJOR 97.7M

DIAL
97.7M

Guardar Cancelar

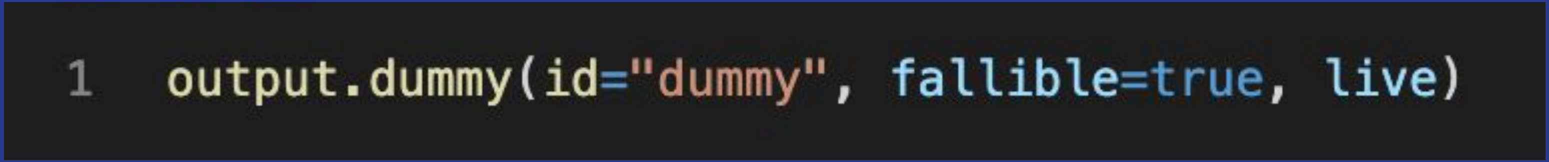
MEDIAINBOX

Lesson 3

If you need record in chunks for later join that segments, output.file is **NOT** your best option

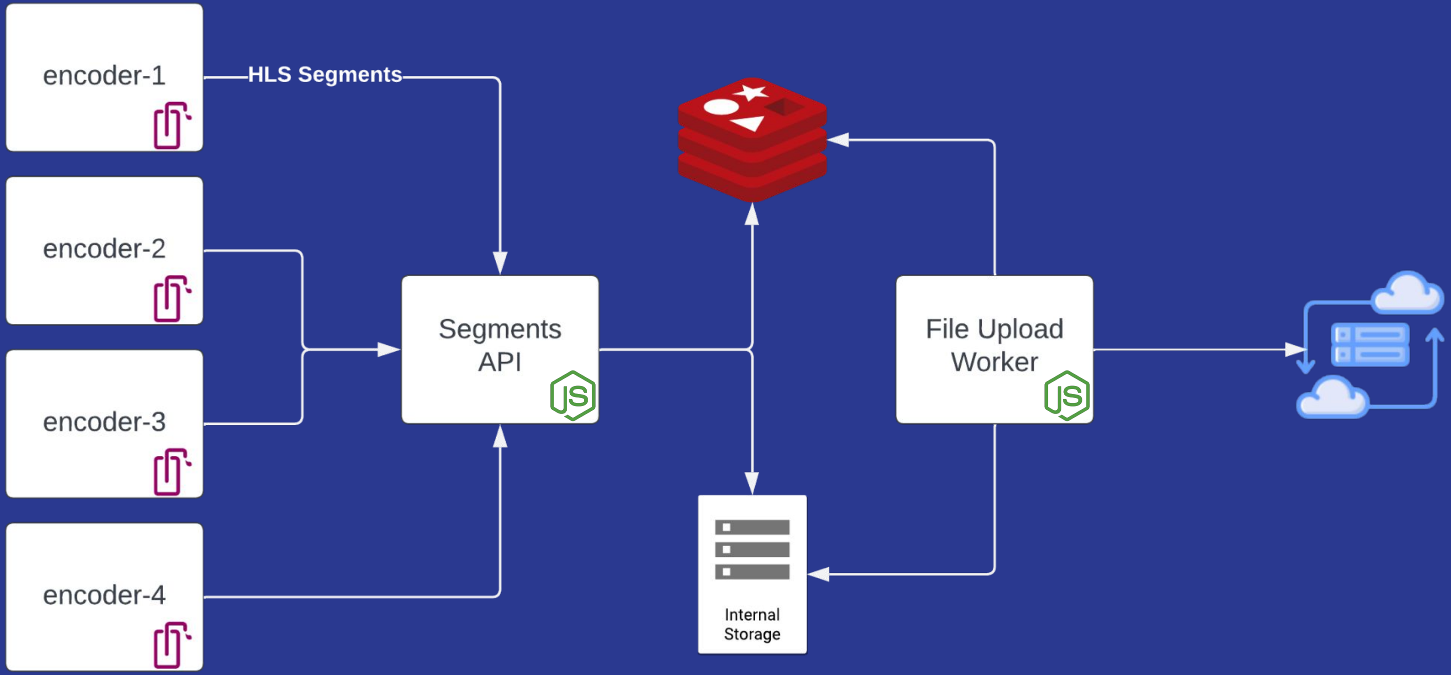
To accomplish that you need to use output.file.hls

Tip: If your source is some external script like our case, put an

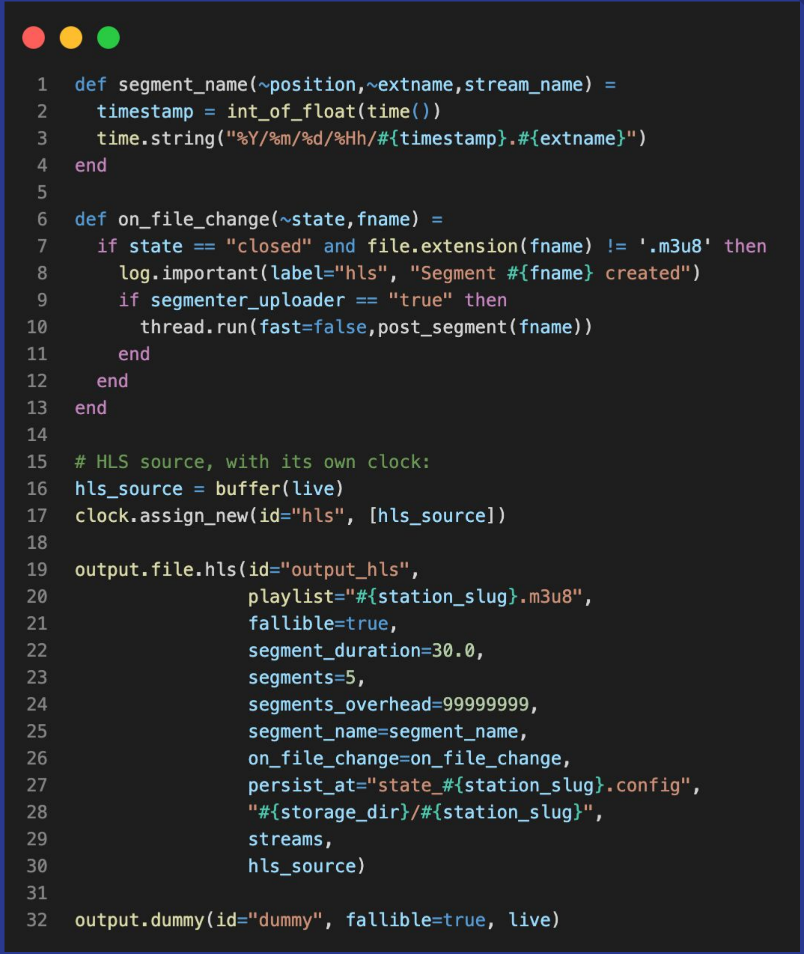


```
output.dummy(id="dummy", fallible=true, live)
```

Streambox inside!



Liquidshop 3



```
def segment_name(~position,~extname,stream_name) =
  timestamp = int_of_float(time())
  time.string("%Y/%m/%d/%Hh/#{timestamp}.#{extname}")
end

def on_file_change(~state,fname) =
  if state == "closed" and file.extension(fname) != '.m3u8' then
    log.important(label="hls", "Segment #{fname} created")
    if segmenter_uploader == "true" then
      thread.run(fast=false,post_segment(fname))
    end
  end
end

# HLS source, with its own clock:
hls_source = buffer(live)
clock.assign_new(id="hls", [hls_source])

output.file.hls(id="output_hls",
                playlist="#{station_slug}.m3u8",
                fallible=true,
                segment_duration=30.0,
                segments=5,
                segments_overhead=99999999,
                segment_name=segment_name,
                on_file_change=on_file_change,
                persist_at="state_#{station_slug}.config",
                "#{storage_dir}/#{station_slug}",
                streams,
                hls_source)

output.dummy(id="dummy", fallible=true, live)
```

MEDIAINBOX

Thanks!

mediainbox.net