

IP Streaming Performance

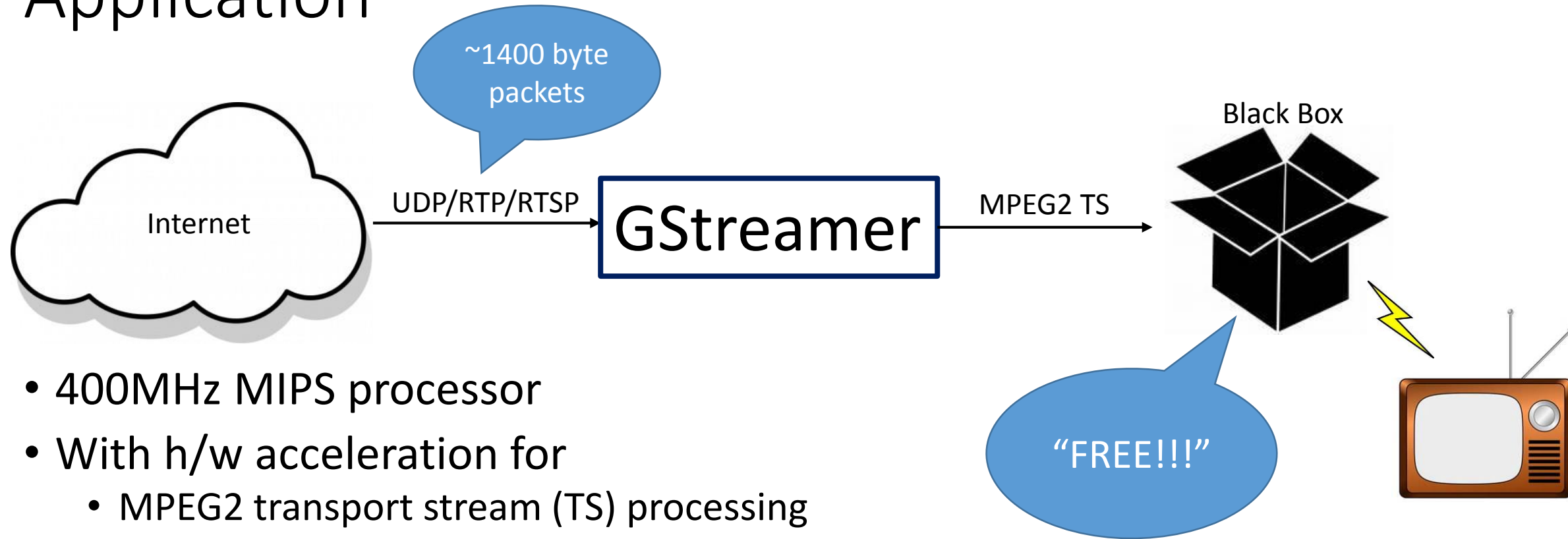
and some Solutions...

David Plowman

BrightSign Digital

October 2016

Application



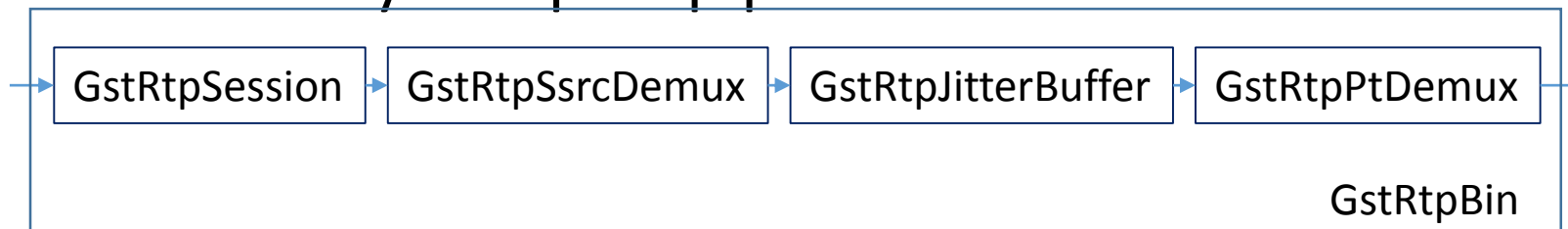
- 400MHz MIPS processor
- With h/w acceleration for
 - MPEG2 transport stream (TS) processing
 - 1080p60 decode
 - Colour conversion and display
- Performance dominated by GStreamer packet/buffer processing

Problem

- Poor performance compared to OEM code

Protocol	OEM Code	GStreamer
UDP	12Mbps	6Mbps
RTSP/RTP	12Mbps	3Mbps

- Relatively complex pipelines



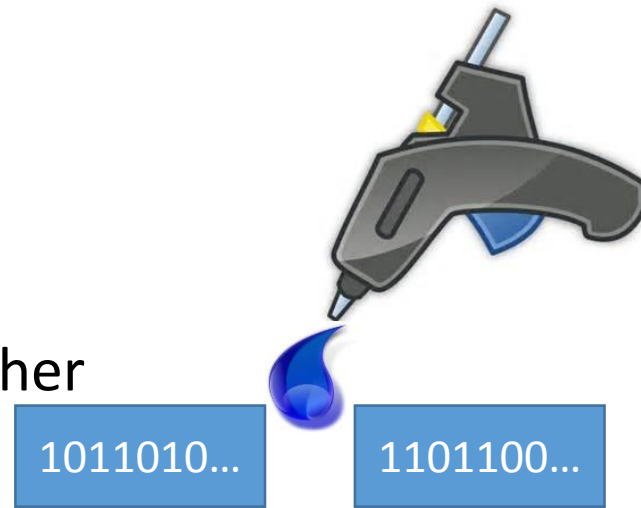
```
#2 gst_app_sink_render
#3 Gst::BaseSink_Class::render_vfunc_callback
#4 gst_base_sink_chain_unlocked
#5 gst_base_sink_chain_main
#6 gst_pad_chain_data_unchecked
#7 gst_pad_push_data
#8 gst_proxy_pad_chain_default
#9 gst_pad_chain_data_unchecked
#10 gst_pad_push_data
#11 gst_proxy_pad_chain_default
#12 gst_pad_chain_data_unchecked
#13 gst_pad_push_data
#14 gst_rtp_base_depayload_push
#15 gst_rtp_base_depayload_handle_buffer
#16 gst_rtp_base_depayload_chain
#17 gst_pad_chain_data_unchecked
#18 gst_pad_push_data
#19 gst_pad_chain_data_unchecked
#20 gst_pad_push_data
#21 gst_proxy_pad_chain_default
#22 gst_pad_chain_data_unchecked
#23 gst_pad_push_data
#24 gst_proxy_pad_chain_default
#25 gst_pad_chain_data_unchecked
#26 gst_pad_push_data
#27 gst_proxy_pad_chain_default
#28 gst_pad_chain_data_unchecked
#29 gst_pad_push_data
#30 gst_rtp_pt_demux_chain
#31 gst_pad_chain_data_unchecked
#32 gst_pad_push_data
#33 pop_and_push_next
#34 handle_next_buffer
#35 gst_rtp_jitter_buffer_loop
#36 gst_task_func
```



Solutions

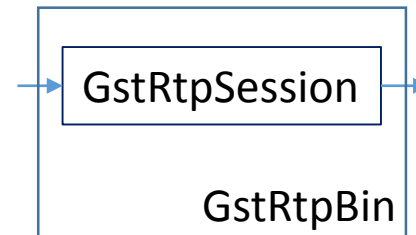
- UDP

- MPEG2 TS packets can simply be glued together



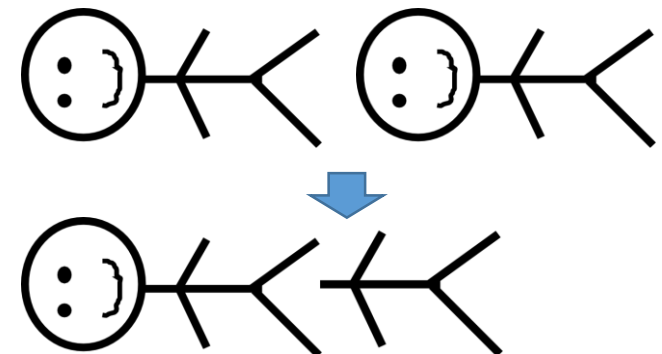
- RTSP/RTP

- Some elements are usually not required



- MPEG2 TS RTP packets can be glued together stripping the headers off all but the first packet

- But be careful about RTCP ⚠
- Many of the same benefits from using buffer lists, though support patchy



Results

Protocol	OEM Code	GStreamer	Optimised GStreamer
UDP	12Mbps	6Mbps	12Mbps
RTSP/RTP	12Mbps	3Mbps	11Mbps

- Suggestions

- Write a more optimised GstUdpSrc that can glue packets together and create buffer lists? Improve support for buffer lists generally.
- Produce a streamlined GstRtpBin for “regular” applications?
- For HTTP streams increase the default buffer size.



Questions? Please come and find me or email me at dplowman@brightsign.biz