

GCCG working group – March 1, 2021

Report out to the Vidtrans audience

Confidential Work of the VSF GCCG AHG

1

What are the “operating points” we see (G-C-C-G) (for TV)

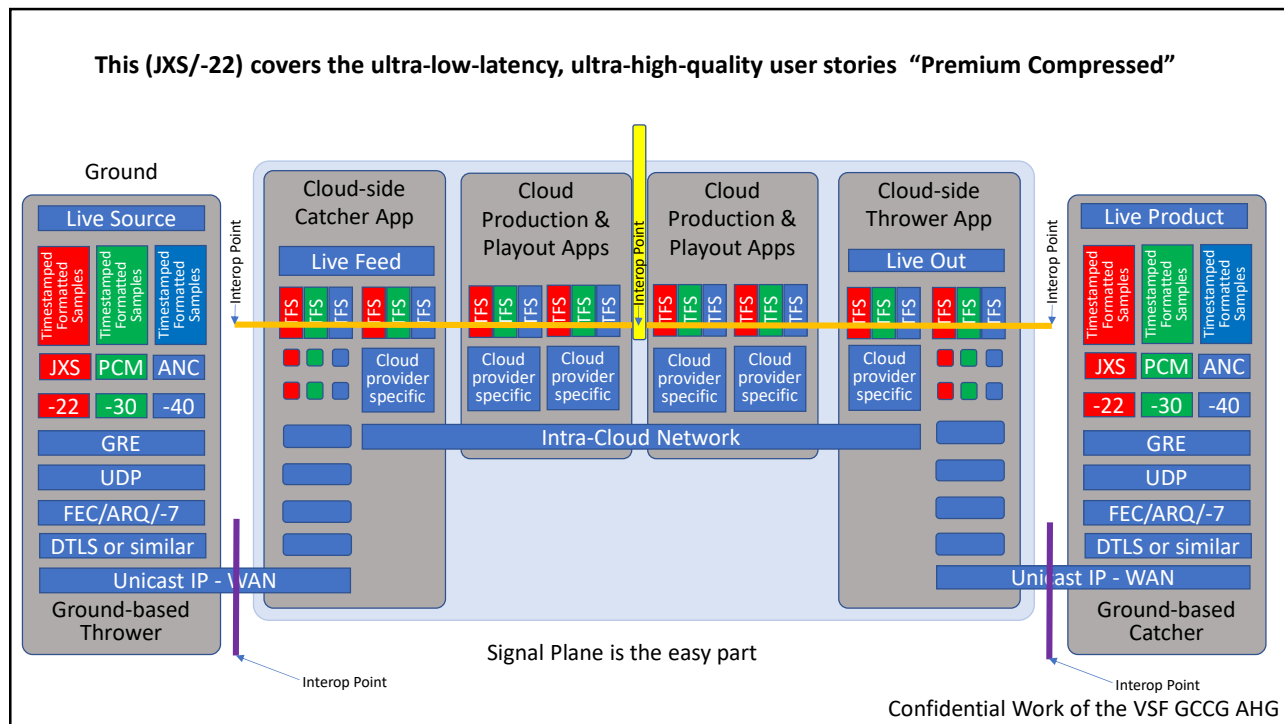
- Uncompressed (2022-6 or 2110-20) (already defined)
- **“PremiumCompressed” – super low latency, super high quality**
 - JXS for (1080i @ 200m?) (UHD @ 1000-1500M)
 - Dual-path reliability model for super-good reliability @ *minimum 100ms*
- J2K-ULL
- J2K (full frame)
- AVCI @ 100Mbps
- **Interactive Latency (small GOP or Prog Refresh)**
 - (HD) H264 (constrained VBV, 4:2:2, 10bit)
 - (UHD) H265 (constrained for latency)
 - 2022-2(TS-RTP)
 - RIST-FEC (+/- multi-path)
- **Interactive Latency ~20Mbit(HD) (ULL restricted VBV buffer 1.5s)**
 - ARQ/RIST or dual-path model or FEC? (Typical 4:2:2/10 profile)
- **Rate Optimized (longer latency @ lower rate) (HD@3-10 Mbits, more delay)**
 - ARQ/RIST or dual-path model or FEC? (Typical 4:2:0/8 profile)
- Zoom / GoToMeeting / Webex – whatever gets a picture on the screen
 - Internet best-effort reliability work

Premium Compressed (JPEG XS)
 VSF TR-08: codec & LAN 2110-22 base
 VSF TR-09: WAN 2110-x extension
 * reference TR-08
 * opt: with 2022-7
 * opt: with FEC (small intl, 1D)
 * opt: GRE Tunnel (ref RIST)
 * opt: encryption (ref RIST)

Bandwidth Optimized
 • (HD) H264 (420/8 standard vbv)
 • (UHD) H265 ()
 • 2022-2(TS-RTP)
 • ARQ
 • Opt: encryption (ref RIST)

Confidential Work of the VSF GCCG AHG

2



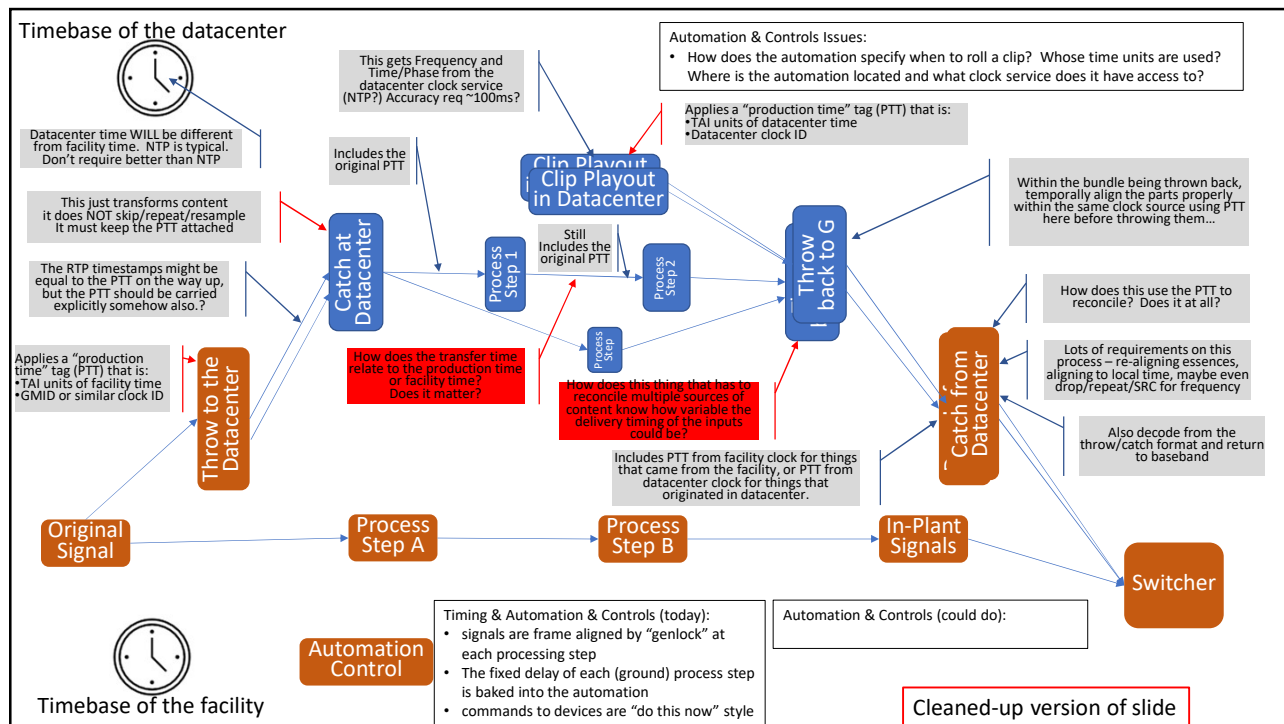
3

Potential work for this group

- Define Requirements for (Premium Compressed) TR-08 & TR-09 for G-C and C-G use cases
 - TR-07, TR-08, TR-09 work in progress in VSF
- Is there something more to document about the TS/IP/H.264 case? Or is it good enough already?
 - Possibly a common spec on how to organize connections (spin out of TR-09?)
- Time-Flow / Time-Transport / Time-Tagging model
 - How do we treat time in the concatenated virtualized systems
 - How do we integrate "real-time" on the ground with "floating time"
 - This is the current WIP in the GCCG group
- The Media Containers / Object Format structures for hand-off from application to application
 - The github part of AWS CDI is a step in this direction
 - And also how this handoff interacts with latency and latency accumulation
 - This is the next topic of the GCCG group

Confidential Work of the VSF GCCG AHG

4



5

How to pass data between SW applications?

- At what point in the stack should we ***stop*** defining?
 - We should define the media blob – the data structure that hands off between the application and the transporter
 - The pixels/samples/etc
 - The package of related information / metadata / description
 - The time-tag or time-information about them (see other discussion)
 - Document "canonical use models" (stripes –vs- frames, similar for audio)
 - AWS CDI spec on github has an approach for this.
 - Should we look at the offer/connect model
 - How does the application know what it could get
 - How does an application advertise what it can provide
 - How does entitlement/security/etc map into this?
- We should NOT define the ***how it gets there*** transporter

6

Original Activity Proposal -- Objectives

1) Determine if a common set of requirements for Ground-to-Cloud-to-Cloud-to-Ground media essence communications exists.

- Yes, common requirements exist

2) If common requirements exist, establish a common technical approach (perhaps to be documented in a VSF Technical Recommendation) for communication of media essences from G-C-C-G.

- WIP, good discussion
- VSF TR08 & TR09
- VSF RIST

Confidential Work of the VSF GCCG AHG